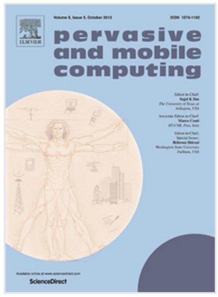
(Article begins on next page)

Accepted Manuscript



On the effect of human mobility to the design of metropolitan mobile opportunistic networks of sensors

Marcello Tomasini, Basim Mahmood, Franco Zambonelli, Angelo Brayner, Ronaldo Menezes

PII: S1574-1192(16)30119-5
DOI: http://dx.doi.org/10.1016/j.pmcj.2016.12.007
Reference: PMCJ 785

To appear in: *Pervasive and Mobile Computing*

Received date: 10 August 2016
Revised date: 11 November 2016
Accepted date: 23 December 2016

On the Effect of Human Mobility to the Design of Metropolitan Mobile Opportunistic Networks of Sensors

Marcello Tomasini[a,1,*], Basim Mahmood[b,1], Franco Zambonelli[c], Angelo Brayner[d], Ronaldo Menezes[a,1]

[a] *Florida Institute of Technology, USA*
[b] *University of Mosul, Iraq*
[c] *Università di Modena e Reggio Emilia, Italy*
[d] *Universidade Federal do Ceará, Brazil*

Abstract

We live in a world where demand for monitoring natural and artificial phenomena is growing. The practical importance of Sensor Networks is continuously increasing in our society due to their broad applicability to tasks such as traffic and air-pollution monitoring, forest-fire detection, agriculture, and battlefield communication. Furthermore, we have seen the emergence of sensor technology being integrated in everyday objects such as cars, traffic lights, bicycles, phones, and even being attached to living beings such as dolphins, trees, and humans. The consequence of this widespread use of sensors is that new sensor network infrastructures may be built out of static (e.g., traffic lights) and mobile nodes (e.g., mobile phones, cars). The use of smart devices carried by people in sensor network infrastructures creates a new paradigm we refer to as *Social Networks of Sensors* (SNoS). This kind of opportunistic network may be fruitful and economically advantageous where the connectivity, the performance, of the scalability provided by cellular networks fail to provide an adequate quality of service. This paper delves into the issue of understanding the impact of human mobility patterns to the performance of sensor network infrastructures with respect to four different metrics, namely: detection time, report time, data delivery rate, and network coverage area ratio. Moreover, we evaluate the impact of several other mobility patterns (in addition to human mobility) to the performance of these sensor networks on the four metrics above. Finally, we propose possible improvements to the design of sensor network infrastructures.

Keywords: Wireless Sensor Networks (WSNs), Human Mobility, Opportunistic Networks, Social Networks of Sensors (SNoS), Mobile Ad-Hoc Networks (MANETs)

1. Introduction

Since their introduction in the 1950s as a US military application to track Soviet submarines (known as the Sound Surveillance System, SOSUS) [1], sensor networks have taken on an increasing practical relevance [2] and the number of deployed sensor network infrastructures is now difficult to quantify [3]. Further, the recently-introduced paradigm of the Internet of Things (IoT) shows that we are moving toward a world where "smart" physical objects augmented with computing and sensing capabilities will be seamlessly integrated to many aspects of our lives [4]. When connected and mobile, these sensors form a mobile opportunistic network of sensors; while static sensors are attached to the physical infrastructure, mobile sensors are carried by *social entities* (such as humans). In this work, we refer to such a network as a "Social Network of Sensors" (SNoS). The "Social" aspect arises due to the patterns of interactions originated by the movements of the actors carrying the "objects" endowed with sensing capability (sensors) [5]. In fact, people's movements are far from random and inherently embed the social aspects related to human interactions [6, 7].

*Corresponding author
Email addresses: mtomasini@my.fit.edu (Marcello Tomasini), bmahmood@uomosul.edu.iq (Basim Mahmood), franco.zambonelli@unimore.it (Franco Zambonelli), brayner@dc.ufc.br (Angelo Brayner), rmenezes@cs.fit.edu (Ronaldo Menezes)
[1] BioComplex Laboratory, School of Computing, Florida Institute of Technology, 150 W. University Blvd., Melbourne, FL 32901, USA

One aspect of SNoS that makes it different from traditional sensor network infrastructures is that some of these sensors can be mobile as a consequence of the social movement of individuals (e.g. humans, vehicles). This means that movement is achieved *without requiring any energy from the sensor to achieve mobility*. Individuals carry devices such mobile phones, smart watches, and tablets which are endowed with sensing capabilities [8, 9].[2] As a consequence of such mobility, these nodes can be very effective in a SNoS infrastructure given that they can be opportunistically exploited to dynamically patrol and monitor the environment (e.g., traffic monitoring, monitoring of crime, prevention of forest-fires, tornado-warning systems, detection of chemical and biological traces as well as radioactivity). Mobile nodes in SNoS can be extremely valuable in locations where—in the future—enough smart objects can opportunistically exploit the environment or where the deployment of dense-enough wireless static sensor nodes would not be possible or economically feasible. In fact, the existence of mobile nodes can be used to improve network coverage efficiency or to eliminate "blind spots of coverage" caused by low availability of sensors [10, 11].

In recent years, a considerable amount of research work has been devoted to understanding how some degree of mobility, typically via mobility mechanisms embedded in sensor nodes [12], may improve the effectiveness of sensing coverage and data dissemination [13]. From a different perspective, many researchers faced the issue of exploiting humans carrying smart phones as sensor nodes, and analyze how and to which extent they can effectively achieve sensing goals [8]. However, *the issue of how a network of mobile sensors integrated with fixed sensor-network infrastructures can produce an overall impact on sensing performance and data delivery has not been deeply investigated*. Moreover, neither the scaling of such network with respect to the number of mobile nodes nor the optimization at the level of network design has been proposed. Indeed, in the design of sensor infrastructures, the mobility is surely coupled with fixed sensors, because different mobility patterns may lead to different interaction patterns between mobile sensors and static sensors. These interaction patterns reflect are the consequence of the interaction between social actors in the SNoS, thus the understanding of different mobility patterns in the context of sensor networks should help us design more efficient infrastructures.

The understanding of human trajectories are an important component in a number of major subjects, such as urban planning, traffic forecasting, epidemics modeling, and of course, our proposed SNoS. Specifically, a class of mobility models, generally known as random walks, has been quite successful at reproducing the statistical properties of movements at a long-term limit. The main mechanism consists of a series of successive random events, alternating displacements (i.e. movements) and changes of direction. Therefore, we compare the performance of four of the most studied mobility models in literature. Brownian Motion [14] is used as a reference null model, because it is widely used in formal proofs due to its analytic tractability. Lévy Flight [15, 16] has been used to model both animal and people spatial patterns. CTRW (Continuous-Time Random Walk) [17] enabled the modeling of temporal patterns of trajectories. The Individual Mobility (IM) model as proposed by Song et al. [18] represents the state-of-the-art in human mobility modeling. The mobility models have been selected not only because of their relevance to the field, but also due to their accuracy in reproducing large-scale regularities in human trajectories.

The main idea of this work is to characterize and analyze the performance in a SNoS composed of both mobile and static sensors in order to understand the design choices that could be made in deploying SNoS infrastructures in metropolitan environments. In the model we propose we assume that:

- Mobile sensors are carried by people, and the human mobility patterns that drive their movements have been modeled around the typically social mobility patterns of metropolitan environments. Although sensors could also be carried by cars or other mobile entities, in urban environments (e.g. Manhattan), the mobility of people tend to dominate others kinds of mobility.

- Static sensors are assumed to be deployed uniformly throughout the environment. This is realistic for metropolitan environments, where one could assume sensors to be deployed (or recruited) via specific design choices.

Using a number of simulations, we evaluated the behavior of a SNoS regarding the number of mobile sensors and four different mobility patterns in metropolitan environments. For that, four metrics have been benchmarked: detection time, report time, sensing coverage area, and data delivery rate. *Detection time* measures the responsiveness of

[2] At the time of writing of this paper, some popular smartphone brands were already equipped with a myriad of sensors. E.g., The Samsung Galaxy S5 had geomagnetic, temperature, humidity, air pressure, and many other sensors.

the network in recognizing new events. *Report time* captures the delay in information flow between two locations (generally called as source or detection location, and sink, where the event needs to be reported to). *Coverage area* describes the efficiency in covering the environment without leaving blind spots. Last, *delivery rate* is the fraction of events reported within a time constraint. We observe that mobile sensors can notably help delivery of the data to specific sinks (nodes responsible for collecting events detected by the network), and that the decision on where to place sink nodes is crucial to the performance of the overall SNoS infrastructure. Furthermore, we show how to extract the minimum number of mobile sensors required to achieve a good trade-off between performance and cost by detecting the threshold density of mobile sensors after which there is nearly no benefit to performance. The main results of this work can be summarized as follows. First, the effect of mobility models to the performance of sensor networks is crucial in sparse network becoming less relevant as the sensor density increases. Second, the sensor radii (communication range) is more important than the density of sensors in the environment. Third, the Individual Mobility model [18, 19] (the most accurate model to date) performs poorly compared to other mobility models, which may pose a big challenge in the engineering of sensor network infrastructures.

The rest of the paper is organized as follows: Section 2 discusses related work in sensor networks, participatory sensing, and mobility dynamics; Section 3 describes our SNoS model , the design of the SNoS simulator, and how we compare the performance of different mobility models; Section 4 illustrates and discusses the simulation results and our main findings; Section 5 summarizes this paper and provides the basis for future works.

2. Related Work

This work relates to three different research areas: *(i)* sensor networks, which deals with spatially distributed sensors used to monitor physical and environmental conditions; *(ii)* participatory sensing, which refers to the ability to create sensing infrastructures out of smart phones already distributed with people, and *(iii)* mobility dynamics, which aims at modeling and understanding patterns of mobility and how they can be used to model mobility of objects in the real-world.

2.1. Sensor Networks

Most of the works on sensor networks are on coverage, protocols, and algorithms to optimally deploy sensors [20–24]. The main difference among these works is on how the desired positions of sensors are computed. Typically, mobility is only exploited to achieve a static optimal configuration in an enlarged sensing environment, rather than in an environment where the dynamics of sensors' movements are exploited as an added characteristic of the sensor infrastructure [25].

Liu et al. [10, 26] formally proved, under simplified assumptions, that sensor mobility can be exploited to effectively reduce the detection time of a stationary intruder and improve network coverage when the number of sensors is limited. The basic concept behind their work is that, given a fixed number of sensors, the coverage area is inherently bound by the density of sensors. However, if sensors are allowed to move, the area that can be covered increases because sensors are now able to reach locations in the environment that would otherwise never be covered. In any case, the mobility models investigated are based on simple assumptions about mobility, and nothing is said about the potential impact of more realistic mobility models. In [12], Chin et al. investigate the problem of target detection and they evaluate the detection latency using a mobile sensor network with coordinated sensing but uncoordinated mobility. They found out that mobility improves detection latency compared to a static network with the same number of nodes as the mobile one. The sensing range of the sensors is based on a probability function; the further the sensor is from a target, the less likely the sensor will detect the event. Our model uses the simpler Boolean approach, which assumes that if the event is within the radius of a sensor, the event will always be detected.

Understanding the impact of mobility patterns is also of relevance in the design of forwarding protocols. Bild et al. [27] exploited the predictability of mobility patterns of people to improve the reliability and scalability of routing on Mobile Ad-Hoc Networks (MANETs). Scalability is important because most routing protocols do not scale to very large networks without traditional fixed infrastructures (obviously hard to be made available in hostile environments). Since they used a mobility model very similar to ours, we believe that most of their results also apply to our work, the consequence is that it may be possible to build MANETs of wireless devices (phones) at a city scale. More recently, human mobility has been studied in the context of metropolitan environments [28].

Authors confirmed some of the results obtained in smaller test environments. In particular, wireless networks that exploit human mobility exhibit a power-law distribution for contact and inter-contact times. Moreover, they showed the characteristics of the connectivity properties of human mobility, such as the small world phenomenon and the non-homogeneous betweenness centrality of sensor nodes, that is, some devices are better to act as forwarders.

2.2. *Participatory Sensing*

Participatory sensing is concerned with the possibility of using users' personal devices to form a sensor network [8, 9]. This is another area that recently emerged in the scientific community in which the impact of user mobility is very important. Participatory sensing relies on users holding smart (e.g., with sensing capabilities) phones to act as sensor nodes, i.e. by dynamically involving them in acquiring information about specific phenomena in an environment. Our approach tries to incorporate the best of the traditional sensor network sensing and of participatory sensing, by defining a hybrid system in which mobile human-carried devices and fixed sensing devices cooperate to improve coverage and data distribution.

In pure participatory sensing systems, the location and movement of people have a dramatic impact on the coverage of the sensed phenomena. This bi-directional relationship aspect between human and opportunistic networks has been covered for the case of IoT by Guo et al. [29]. On one hand, the opportunistic behavior becomes the main media to sense and monitor the human dynamics (e.g., mobility patterns can be learned from the GPS traces) and conversely, the performance of opportunistic IoT systems is affected by such dynamics. For example, discovery of an event cannot be guaranteed in participatory sensing, because one cannot exert control on the position of users or on their willingness to participate. At most, based on the analysis of mobility patterns, one can reason about the sensing area and on the minimal percentage of users that must be involved to probabilistically ensure coverage of such an area. Having the opportunistic, distributed sensing and computing characteristics, our model falls in the broader category of Mobile Crowd Sensing and Computing (MCSC) systems. Guo et al. [30] thoroughly review unique features and applications areas of MCSC systems. It also introduces a framework to build MCSC systems based on 5 layers: 1–crowd sensing; 2–data transmission; 3–data collection infrastructure; 4–crowd data processing; 5–applications. Our work aims to provide support in solving the issues at the data transmission level by providing a model to test protocols scaling under different mobility models and spatial distributions, but without having to deal with low level details of wireless technologies. Furthermore, it helps to test the robustness of data delivery among highly mobile devices. It also contributes at the data collection infrastructure layer, because it allows to test the deployment of nodes in the network which is useful to predict the scalability of the system and to pinpoint issues in coverage of the area. Our approach by accounting for the existence of fixed sensor nodes, goes in the direction of a more realistic future scenario, in which the opportunistic synergy of infrastructural sensing devices and of users will increase sensing coverage and exploit the best of human mobility.

Some recent approaches to participatory sensing propose to increase the percentage of people participating in sensing activities or to affect their mobility patterns so as to reduce coverage problems. Many approaches rely on monetary incentives [31] to improve user participation, although some recent proposals also suggest affecting participation and mobility patterns by means of a gaming approach [32], the recent *Pokemon Go* game[3], is a clear example of people movements being drive buy incentive in the game. Even though our current work does not account for the possibility to influence the mobility patterns of users, this is definitely an interesting area of future development and useful to the design of sensor networks. Jaimes et al. [33] provides an extensive survey on participatory sensing that we highly recommend to interested readers.

2.3. *Mobility Dynamics*

Random walks are a powerful tool to model individual mobility; in our work, we show that these models can be useful to better analyze the impact of mobility in SNoS [34]. One of the first patterns of movements studied by scientists was the model that is generally referred to as Brownian motion, because it was observed in 1827 by the botanist Robert Brown, and later explained by Albert Einstein [14]. This process consists of an alternation of fixed

[3]New as of July 2016

size steps in arbitrary directions. A Brownian motion is, in some sense, the limit in distribution of a symmetric random walk M_n defined in terms of a series Z of n random variables $Z_1, Z_2, \ldots, Z_n$ with $M_0 = 0$:

$$M_n = \sum_{i=0}^{n} Z_i \quad Z_i = \begin{cases} 1 & \text{with } P(1) = \frac{1}{2}, \\ -1 & \text{with } P(-1) = \frac{1}{2}. \end{cases} \tag{1}$$

In other words, Brownian motion can be approximated by a random walk when the number of steps $n \to \infty$. It is often assumed that Brownian motion does not have a practical relevance as a mobility model but it has been shown to be useful in describing the flight pattern of some insects [35]. In spite of not being an appropriate model of human mobility, simple random walks as described here, engender a number of more complex mobility models and have also been used in research literature for the analytic tractability.

In 1947, the French mathematician Paul Pierre Lévy proposed a new type of random process based on a specific kind of probability distribution known as *heavy-tailed*. More precisely, in a Lévy flight the probability $P(\Delta r)$ of finding a step of length Δr decays with

$$P(\Delta r) \simeq |\Delta r|^{-1-\alpha}, \tag{2}$$

for $0 < \alpha < 2$. The proposed method has been shown to be extremely useful to model the dynamics of critical phenomena in physics [36]. Moreover, scientists have used Lévy walk to describe the flight pattern of wandering albatrosses [37], the foraging pattern of spider monkeys [38], and also human mobility [16].

The aforementioned random processes can only capture the spatial properties of the movements. An extension of such processes that is capable to capture both the spatial and the temporal patterns of motion is the family of Continuous Time Random Walks (CTRW). CTRW is a random walk that includes random waiting times $\{\Delta t_1, \Delta t_2, \ldots\}$ between movements (jumps) $\{\Delta r_1, \Delta r_1, \ldots\}$ represented by mutually independent identically distributed (iid) random variables. Thus, the number of jumps n taken by a particle in a time interval Δt is also a random variable in CTRW. Both the displacements Δr_i and waiting times Δt_i are drawn from two probability density functions, respectively $P(\Delta r)$ and $P(\Delta t)$. A case where the distribution of waiting times has infinite variance (e.g., power law) is treated in [39] and it has been used by Brockmann et al. [40] to describe the scaling laws for the flow of bank notes, and then to infer the dynamics of human travels. Jump lengths distribution and waiting time distribution in human mobility have been found to follow a power law

$$P(\Delta r) \sim (\Delta r)^{-(1+\alpha)} \quad P(\Delta t) \sim (\Delta t)^{-(1+\beta)}, \tag{3}$$

with $\alpha = 0.59 \pm 0.02$ and $\beta = 0.60 \pm 0.03$. Gonzalez et al. [41] further improved the result by showing that the mobility of humans is characterized by a time-independent travel distance and a significant probability to re-visit previously visited locations. Their findings suggest that human trajectories are even better approximated if the jump length distribution follows a truncated power law

$$P(\Delta r) \simeq (\Delta r + \Delta r_0)^{-1-\alpha} \exp(\frac{-\Delta r}{k}). \tag{4}$$

Enabled by the increased availability of location data [42], a number of recent studies have focused on proposing sound and realistic human mobility models. Today, it is generally understood that human mobility patterns are non-random. This property has been studied by Song et al. [18] who explored the limits of predictability in human dynamics; the authors found a 93% predictability in user mobility and showed that it is independent of the distance users cover on a regular basis even if travel patterns differ considerably. These results are especially relevant because they tell us that human movements are not random thus the predictability can be exploited to increase performance and efficiency of sensor network protocols (e.g., routing data to sinks). Song et al. [19] have proposed a model for human mobility based on preferred locations where the jump length Δr and the wait time Δt follow a power-law distribution with exponential cutoffs as given in the following equations:

$$P(\Delta r) = (\Delta r + \Delta r_0)^{-1-\alpha} \exp(\tfrac{-\Delta r}{k_1}), \quad P(\Delta t) = (\Delta t)^{-1-\beta} \exp(\tfrac{-\Delta t}{k_2}), \tag{5}$$

where α and β control the scaling of jump length and wait time respectively, k_1 is the cutoff value of the jump length, k_2 is the cutoff value of the wait time, and Δr_0 is the minimum jump length. Furthermore, the model incorporate two generic mechanisms, exploration and preferential return, both unique to social human mobility and missing from the traditional random-walk (Lévy-flight or CTRW) models. The two mechanisms are as follows:

Exploration: a scaling law is proposed to indicate that the tendency to explore additional locations decreases with time. Indeed, the longer we observe a person's trajectory, the higher is the probability that he visited all nearby locations. Each jump can be an *exploration jump* with probability

$$P_{new} = \rho S^{-\gamma}, \tag{6}$$

where S is the number of distinct visited locations by the person (maybe carrying a sensor), while ρ and γ are parameters that characterize human mobility.

Preferential Return: humans show significant propensity to return to previously visited locations, such as their home or workplace; this happens with the complementary probability

$$P_{ret} = 1 - P_{new}. \tag{7}$$

The above means that a person goes back to a visited location i choosing it with a probability $\Pi_i = f_i$, proportional to the number of visits f_i made by that person to that location. As a person visits new locations, the number of distinct visited location S increases, thus reducing the probability of a new exploration jump (Equation (6)). This model—even in its simplicity—is able to capture most of the characteristics of human mobility, thus we chose it as our reference human mobility model.

Some recent works in human mobility modeling are based on Network Science to represent social ties, by considering these as a primary drive of individuals' movements. Social relations are described as a network, where nodes represent individuals and weighted links represent the strength of the social connections. The main idea is that the next location chosen by the user depends on the position of people with whom the user shares social ties. For instance, the HCMM model [43] applies and extends this idea by adding a location preference and incorporating power-law distribution of the jumps. In order to capture the periodical pattern in movements, GeSoMo [44] introduced the concept of time-varying networks, that is, the social strength of relationships among users (i.e. the weight of the edges in the social network) changes with time. Although these models give a thorough representation of human movements in a very particular scenario, they lack generality and they are usually too complex for mathematical reasoning and formal analysis. Furthermore, they require a large number of parameters, which are unknown, and tuning to accurately represent the chosen scenario. As such, we neglect to model the microscopic details of individual movements and focus only on regular everyday human mobility patterns [45].

3. Simulating Social Network of Sensors (SNoS)

In this section we propose a model for simulating Social Networks of Sensors. We were motivated to introduce our model, because most of the theoretical results in sensor networks and mobility do not match to real setups, that is, they use random motion and uniform sensor deployment which tends to be unrealistic for a Metropolitan scenario. At the same time, we tried to avoid the complexity and lack of broad applicability of truly realistic simulations. In practice, the choice of a simulator depends on two factors: the environment in which implement the simulation, and the model of the simulation. There exist several environments that enable to simulate with different degree of fidelity the layers of the ISO/OSI network stack, such as ns2-3 [46, 47], OMNeT++ [48], dtnsim [49], dtnsim2 [50], and the ONE [51]. Simulations based on these environments, such as [52] and [53], often require real-world traces that have low spatial and temporal granularity, lack a high number of individuals, or the population in them is fixed and highly specialized. Furthermore, these studies tend to focus mostly on the routing protocols. Another issue is that in dealing with such a fine detail in the simulation, the models are prevented from scaling to large number of devices or over large time periods. The CCPAC simulator [54] for opportunistic mobile networks (OMNs) attempts to improve scalability, but then it ends up concentrating on only one thing, the routing process in OMNs. No other part of the network stack is simulated.

Therefore, to provide a trade-off between accuracy and generality in the representation of a sensor network, we introduce some *social aspects*, and we create a new sensor network paradigm we named Social Networks of Sensors

(SNoS). With this approach, the temporal and spatial granularity can be set as high as what is needed for good simulation. Also, the number of nodes and their behavior is easily varied for different scenarios and sensitivity analysis. Sensors are not deployed *a priori* nor move randomly in our approach. Instead, we attach sensors to people in order to get mobility for free (as a result of the natural movement of people), which leads the movement of sensors to follow that of the population; they follow a pattern characteristic of human mobility. This pattern is especially important to consider, because sensors that interact with each other also reflect the pattern of interaction of the entities to which they are attached, i.e. sensor's interaction patterns reflect human social behavior. By introducing the SNoS paradigm, we argue we can gain a better understanding of the performance and the design issues that arise in such sensor networks.

There are many scenarios where social dynamics may have an impact on Social Networks of Sensors. The kind of network we envision here can be found in smart cities. In these cities there are mobile and static sensors communicating with each other. This kind of environment is usually characterized by a relatively high sensor density and many types of mobile entities (e.g., people, cars, bikes). SNoS networks have a number of design challenges. Performance-costs trade-offs are quite stringent, indeed while critical events (e.g., fires, thefts, leakages) should be detected as fast as possible and area coverage should be maximized, the number of sensors that can be deployed is likely very limited, and power consumption poses a limit on the sensing range. Moreover, given the size of the network (as number of sensors and area to be covered) and the need to be both resilient to failures and reliable in terms of availability, it is recommended that control be decentralized and self-organized. Finally, another major challenge to deal with is the heterogeneity in terms of connectivity, the computational power, and the power requirements of the entities in the network. The solution we propose is to exploit human mobility along with the current infrastructure to reach a reasonable tradeoff between performance and costs. Indeed, sensors carried by vehicles or people increase the coverage by exploring new locations as time passes.

Our simulations focus on two aspects. First, we compare the performance of different mobility models in a SNoS with both fixed and mobile sensors, paying special attention to human-like mobility. In particular, we benchmark three issues in sensor networks: *(i)* the time t_D to detect an event (source) in the environment, *(ii)* the time t_R to report that event to a specific location (sink) in the environment, and *(iii)* the fraction $f_a(t)$ of sensing-coverage area. *Our ultimate goal is to find how much the mobility model affects performance*, and if there is a threshold in sensor density after which the mobility model is less relevant to the performance. Second, we show how to deploy a SNoS realistic environment (based on population density) and the scaling of performance that can be expected.

3.1. The Model

We start with a representation of the environment. We envision a mix of mobile and fixed sensors in a metropolitan environment where people carry sensors forming a SNoS, but fixed sensors are attached to the city infrastructure [55]; Figure 1 depicts a city with fixed sensors (light, temperature and sound) coupled with people distributed in the city who assumed to carry smart devices with sensor capabilities. The city model consists of a square lattice of side ℓ (representing the metropolitan area) divided in square patches of one unit area (representing blocks); we have used this approach because it avoids us worrying about side-effects to our results due to the geography of the city. In this environment, we deployed two types of sensors: *static sensors* are distributed along the regular lattice while *mobile sensors* are distributed based on a negative exponential probability from the center of the city as proposed by Clark in his population density model [56]:

$$P(\delta \leqslant d) = 1 - e^{-\lambda d}, \tag{8}$$

where $P(\delta \leqslant d)$ represents the probability that the distance of a sensor is at most d from the center of the city (simulation environment). Our intent is to cover the simulation area in a very similar way as cities are organized, that is, most people are in the urbanized center while some live in the surrounding areas near the city and frequently move into the center. Focusing on metropolitan environments, we set to work under conditions that resemble densities of typical metropolitan areas, therefore we computed the reference city density $RD = 2000$ ppl/km^2, as the average density of 690 cities of the developed world with a population greater than 500,000 people as indicated in [57].

We assume a Boolean sensing network (where the event is either detected or not) with a fixed sensor radius r both for mobile and static sensors, we can argue that an event can be detected if and only if the event is located at a distance, $d \leqslant r$. Once the area of the environment (ℓ^2) is fixed, we can calculate the number of static sensors n_s required to

Figure 1: A sketch of a city where fixed sensors are found as part of the infrastructure and people walk around possibly carrying sensing devices. The combination of both fixed and mobile devices forms a SNoS.

guarantee full coverage of the metropolitan area as:

$$n_s = k^2,\ k = (\ell + r)/r, \tag{9}$$

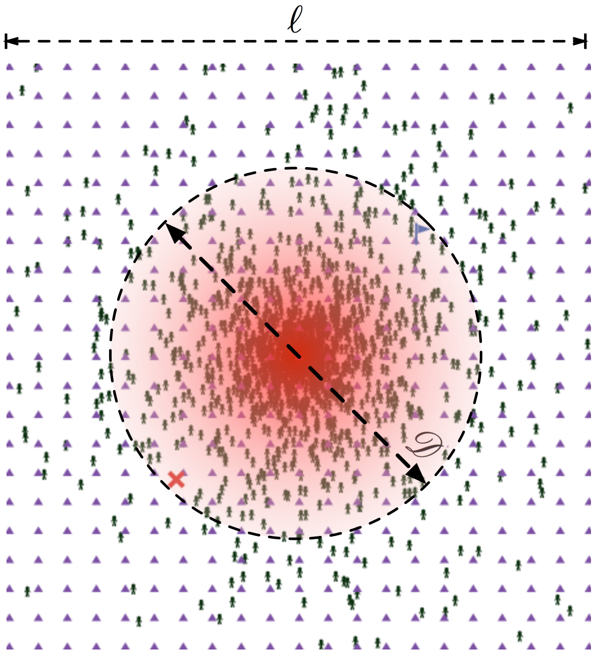
where r ($\ll \ell$) represents the radius of transmission in a square lattice of side ℓ. Let us define A as the area of the square environment, and a the area of a unit of area. Then, the number of mobile sensors n_m required to reach a desired sensor density $\lambda = (n_s + n_m)/A$ is given by:

$$n_m = \frac{x \cdot RD \cdot A}{a} - n_s, \tag{10}$$

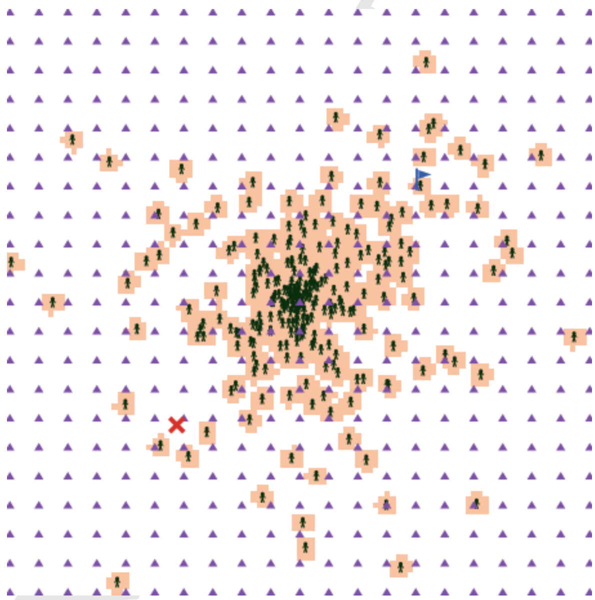
where $x \in X \subseteq [0, 1]$ represents the fraction of reference density we consider.

Now, to simulate the detection and reporting of an event, we include two special markers in the environment: the *event* and the *sink*. The event is what we want to detect (e.g., a fire, an explosion), whereas the sink is the place to where report the event (e.g., a police station). The event lasts for the entire length of the simulation. This assumption can be explained as such: in a system the effects of the event survive longer than the event itself, thus they can be detected to infer the event. We placed the sink and the event in the environment at a distance $\mathscr{D}$ from each other and $\mathscr{D}/2$ from the center of the environment. Thus, the value $\mathscr{D}$ corresponds to the diameter of a circle centered in the middle of the environment (city), such that approximately 80% of mobile sensors are included within this circle. By placing the event and the sink at distance $\mathscr{D}$, we argue that they are located in the periphery/suburbs of the city. Such setup represents an average scenario where a message has to be transmitted from one side of the environment to the other (Figure 2). We chose this approach of fixing event and sink to remove one variable from the simulations which would arise in case of a random deployment of these two markers.

Mobile sensors move at a constant speed, and they follow one of the mobility models we introduced before. We chose a constant speed because we assume mobile sensors are carried by pedestrians. People can walk at different speeds, but the speed is in the range of 3-5 km/h (7 km/h is the max speed for a walker, past that most people



(a) Center of the City



(b) Mobile Sensor Radius

Figure 2: (a) Setup of a simulation with 441 static nodes (violet) and 1,559 mobile nodes (dark green). The event is marked with a red cross and the sink with a blue flag. (b) Setup of a simulation environment with 441 static nodes and 265 mobile nodes (green), where we show mobile sensors with their radius of transmission.

start running) [58, 59]. Since we are focusing on the general performance scaling relationship, instead of accurately modeling every aspect, we can consider the impact on the results limited. Furthermore, while pedestrians can take other kind of vehicles that can actually speed up their movements, we can consider our assumption as a relatively safe worst case scenario assumption. As the simulation progresses, the mobile sensors move according to a specified model exploring the environment. Thus, the fraction of the covered area $f_a(t)$, $t \in \mathbb{N}$ increases; $f_a(t)$ is defined as the number of covered locations divided by the total number of possible locations ℓ^2, where ℓ is the side of the square lattice representing the environment (as in Figure 2). A location is considered visited if it has been reached by at least one sensor node during the execution of the simulation. At some point in time, a sensor should detect the event. From that point onward, the sensor starts spreading the information about the detected event to other sensors with a store-and-forward mechanism. The information of the event is stored by the sensors and forwarded to other sensors when they are within the communication range of each other with a store-and-forward mechanism. The forwarding protocol the sensors use to spread the message is the well known epidemic protocol [60].

3.2. The Simulation

The SNoS model has been implemented in NetLogo [61], a multi-agent programmable modeling environment; our implementation is freely available online [62]. The simulator is fully parameterized, and it supports several spatial distributions of sensors (e.g., lattice, uniform, exponential, normal) and different kinds of random walks (e.g., Wiener [14], Rayleigh flight, Cauchy flight, Lévy walk [15], Lévy with an exponential cutoff [63], CTRW [17], IM [18], recency model [64]). However, in this paper we do not present all the mobility models mentioned because the focus of the paper is mostly on human-like movements.

3.2.1. Parameters Used in the Simulation

The length ℓ of the side of the square simulated environment is set to $\ell = 100$. However, we had to modify the model of spatial distribution by truncating the exponential distribution so that all sensors are placed within the city urban area. In order to achieve this truncation, we first concentrated on the Cumulative Distribution Function

(CDF) of the exponential distribution (Equation (8)). In our simulations, we set $P(\delta \leq d = \ell/2) = 0.95$. That is, the probability that a sensor is found between the center of the city and one of the sides of the environment is 95% (since the environment representing the city is a square with side ℓ, as depicted in Figure 2). This choice means that we have an exponential distribution with mean $\mu = \frac{-\ell}{2\log 0.05}$. We have to implement a truncation because a little bit less than 5% of the sensors would be placed outside the simulation environment if we strictly follow Equation (8) (some of them will fall in the corners of the environment). When deploying the sensors in the simulation environment, if they happen to fall outside the environment, their position is recalculated (truncation), and then they are redeployed.

The simulator is executed using different kinds of random walks—Brownian motion, Lévy walk, and CTRW—and these are compared against the Individual Mobility model proposed by Song et al. [19]:

- Song's model has several parameters. The scaling parameters represent exponents of power-laws distributions for the jump length and wait time, while cutoff parameters control the point at which exponential cutoffs take place (Equation (5)). The preferential return is governed by the parameters ρ and γ (Equation (6)). For our model we used the parameter values as suggested by [19]: the jump-length scaling parameter $\alpha = 0.55$, the jump-length exponential cutoff $k_1 = \frac{\ell}{10}$, the wait-time scaling parameter $\beta = 0.8$, the wait-time exponential cutoff k_2 (it has been tested with two different values as explained later in the section), the scaling of preferential return $\gamma = 0.21$, and the preferential return probability weight $\rho = 0.6$. Finally, differently from [19], $\Delta r_0 = 1$ due to the process we used to generate the displacement [65].

  Song's model requires a transient period to build a list of previously visited locations. In our case we chose the transient to be 6000 ticks, which corresponds to the time for the mobile sensors to reach an average of $\langle N \rangle \sim 100$ distinct visited locations. After that threshold, $P(N)$ decrease exponentially (see the supplementary material of [18]).

- Brownian motion is the simplest model used. We assume a constant jump length of 1 unit, while direction is chosen randomly at every jump.

- Lévy walk uses the same scaling parameters of Song's model for the distribution of jump lengths. However, Lévy walk does not have a wait time between jumps.

- CTRW uses the same parameters as Song's model for scaling and exponential cutoff for jump length and wait time. This fact facilitates the quantification of the impact of cutoffs and preferential return, because there are less parameters to take into account in the comparison analysis.

In order to compare simulation units to real-world measurements, we started by matching the size of the simulation environment to a meaningful size of a city. We set the size of the square lattice $\ell = 100\,u = 10\,\text{km}$, which makes the simulation area to be $A = 100\,\text{km}^2$. Once the environment had an actual size we set the sensor radius to match real-world technologies, in our case we chose Bluetooth that has a range of about 10 meters, which is equivalent to $r = 0.1$ in the simulation environment. In order to compute the sensor density in terms of ppl/km^2, we defined the unit area a inside simulation environment as a square of 10×10 patches (1 km^2 in actual area). Given this setup, we now know that one discrete unit of space u in the environment is $u = 100\,\text{m}$. The concept of time in NetLogo simulations is discrete such that at every tick, sensors check if the event is in their radius. Those that have knowledge of the detected event check if there are other sensors in their radius, and if so they spread the information further in a ripple effect. The simulation stops when one of the sensors with the information about the event finds the sink node. We then tried to match the time unit (1 tick) with an equivalent unit in the real world. If we assume that mobile sensors move at a constant speed of $1\,u/\text{tick} = 100\,\text{m/tick}$, and that they are carried by pedestrians with an average constant speed of 5 km/h, then 1 tick equals 1.2 minutes in real time which gives us that 1 hour is equivalent to 50 ticks. We can now give a meaning to the wait-time cutoff k_2. We used 2 configurations: the first has $k_2 = 5$ ticks to show the impact of sensor radius on performance while the second case uses $k_2 = 17\,\text{h}$ as found in [19].

The next step is to compute the number of mobile sensors n_m. Note that from Equation (9), we know that achieving a perfect coverage with only static sensors is not feasible, because it would require more than a million static sensors to cover our simulation environment with a sensor radius $r = 0.1$. Not only that, but also static sensors will only have a minimal impact on performance because they are limited in number and therefore do not cover much area. Hence, mobile sensors represent the *only practical way* to implement a SNoS at a city level. The objective here is not to find the number of mobile sensors to achieve best possible performance, but to reach a certain event delivery ratio in a given amount of time, as a tradeoff between performance and the effort needed to involve a lot of people. We can obtain the number of mobile sensors as a function of the population density of the city from Equation (10). We choose different percentages, $X = \{0.01, 0.015, 0.02, 0.025, 0.035, 0.05, 0.07, 0.09, 0.11, 0.13, 0.15\}$, of reference density $RD = 2000\ ppl/km^2$ and observed the event delivery ratio to the sink. That is, how many runs out of 50 ended before reaching the time limit, given different time constraints (4h, 5h, 6h, 8h, 10h).

3.2.2. Test of the Implementation

The results of our implementation show an accurate behavior in following Song et al.'s model. Testing has been performed allowing a very long run time to confirm the asymptotic behavior (max $S(t) \sim 700$). The number of distinct visited locations over time should follow $S(t) \sim t^{\mu}$, with $\mu = \beta/(1+\gamma)$; with the configuration described here our implementation is rather accurate and gives $\mu \simeq 0.71$ versus the theoretical result $\mu \simeq 0.66$. Our results also exhibit the expected Zipf's law, $f_k \sim k^{-\xi}$ in the visitation frequency distribution of the k-the most visited locations with little differences (slight tendency to underestimate ξ exponent) compared to data showed in Song et al.'s work. The differences are possibly due to aggregation phenomena caused by Voronoi diagrams in Song's approach or due to different spatial resolution.

3.2.3. Limits of the Implementation of the Model

Every simulation has to reach a balance between accuracy and generality. They tend to be quite optimistic due to many factors (e.g., perfect scaling, no overhead in information transmission, simplified mobility models, Boolean sensing range). This is also the case here.

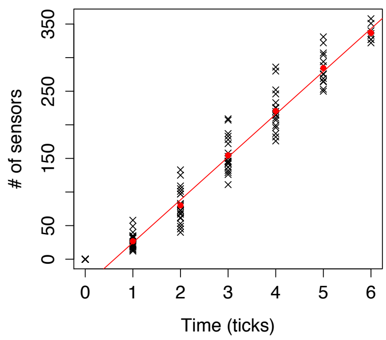
A first issue is the forwarding mechanism based on epidemic protocol, which in practice is inefficient and would waste both bandwidth and memory. However, there are protocols that have close to optimal performance while being much more efficient [66, 67]. Furthermore, the scope of this work is to study the performance from a general scaling perspective more than a quantitative approach, therefore we consider this assumption reasonable. Another issue is the homogeneous mobility (e.g., each sensor is statistically equivalent, and it moves at a constant speed), which does not reflect the real-world heterogeneity, because people can use public transportation as well as other mode of transportation. The impact of such assumption may be relevant in some scenarios because, as shown in Section 4, sensor speed can be the performance bottleneck in a sufficiently dense network. However, we can consider it a relatively safe assumption, because it represents a worst case scenario, and there is research showing that the actual impact is modest [68, 69]. Furthermore, the model allows to simulate only a single center city while many multi-center cities exist. In fact, the current implementation does not allow to distribute sensors according to multiple centers. Still, each one of the centers can be studied separately eventually. Finally, only one event and one sink can be deployed, even though the location is not limited to the deployment used in this work, and they are static thus the deployment is fixed a priori.

4. Simulation Results

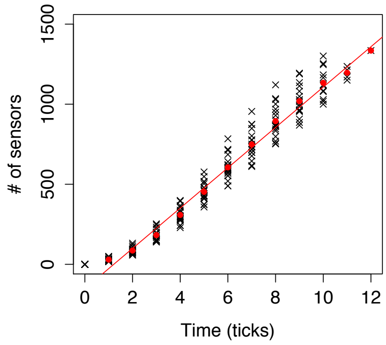
This section is organized as follows. First, we introduce a static network in order to show the performance of such setup. Although very expensive, its performance cannot be matched by a network of mobile sensors. However, the prohibitive cost makes its implementation impractical. Then, we analyze the performance that should be expected from mixed networks, with in-depth analysis of the human mobility model. Last, we propose some interesting solutions and approaches to SNoS design.

4.1. Static Networks

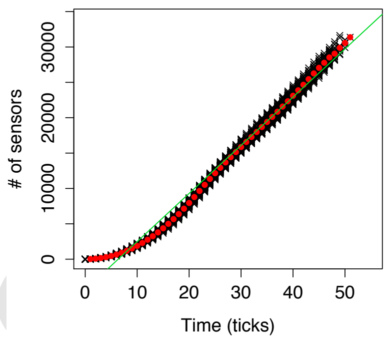
In order to benchmark the effect of mobile sensors, we decided to first look at the performance of a network made of only static sensors where any node has a (indirect) connection to any other node in the environment. Recall that we have a fixed size environment, we first calculated the minimum number of sensors necessary to cover the entire environment. This number is given by Equation (9), where r represents the radius of transmission in a square lattice of side ℓ. If we assume $r = 5$ and $\ell = 100$, then $n_s = 441$. Then, we run the simulator many times to get a pattern of spread of the information defined as the number of sensors knowing the information of the event as a function of



(a) $n_s = 441, r = 5$



(b) $n_s = 1681, r = 2.5$



(c) $n_s = 40401, r = 0.5$

Figure 3: Setups of a network with only static sensors connected in a mesh ($n_s = 441$, $n_s = 1681$, and $n_s = 40441$ respectively). Each column represents 20 runs and depict the number of nodes that have received the information about the event at that time. The red dots represent the average number of sensors knowing the event over the 20 runs. The lines are the linear regression lines of the averages. Adjusted Adj-R^2 and p-values are respectively Adj-$R^2_{441} = 0.997, p_{441} \sim 10^{-6}$, Adj-$R^2_{1681} = 0.9935, p_{1681} \sim 10^{-12}$, Adj-$R^2_{40441} = 0.9872, p_{40441} \sim 10^{-16}$.

time. The configuration here is similar to the one described for Figure 2 where $\mathscr{D} = 60$. Figure 3 shows the spread under this static-only assumption.

Most executions of the simulator stop when the simulator reaches tick number 5, $\bar{t}_{sim} \simeq 5$. It is very clear from Figure 3 that the performance of a static network is very good if we consider the time it takes for the event to spread—it indeed spreads rapidly. But one has to also look at other costs. The configuration depicted assumes $n_s = 441$. Given the nature of the square lattice, if we decrease the radius of the sensors so that $r = 2.5$ then $\bar{t}_{sim} \simeq 10$ ticks which is 2 times longer than the reference setup. While the increase in time is linear and inversely proportional to sensor radius, the number of sensors increase quadratically. Thus, the number of static sensors would have to quadruple to a minimum of $n_s = 1681$ when sensor radius is halved to 2.5. The same reasoning apply when we set $r = 0.5$ and $n_s = 40401$. The time $\bar{t}_{sim}$ is in fact about 10 and 5 times respect to the previous setups, but the number of sensors is respectively 100 and 25 times. Indeed, Equation (9) can be approximated by:

$$n_s \sim \frac{\ell^2}{r^2}. \tag{11}$$

In the real world, one is constantly faced with budget constraints which makes some of setups as depicted in Figure 3 unrealistic. In fact, deploying a large scale infrastructure is very expensive, and it requires a relevant management overhead. If we take the experiment to a scenario as the one explained in Section 3.1, the ability to generate such setups is even harder. An approach to overcome budget limitations is to exploit current infrastructures and attach sensors to mobile agents (e.g., people's smartphones, vehicles) to achieve acceptable performances while keeping costs under control.

4.2. Mobility Model Performance

The analysis in a mixed environment is more complex because we must distinguish between static nodes n_s and mobile nodes n_m. In general, the performance of a sensing network is directly affected by how good is the coverage of an area to discover an event and meet other sensors. In fact, the mobility of the nodes influences how many (unique contacts) and how often (inter-contact time) sensors meet each other. Therefore, mobility models that cover the environment better should also perform better.

4.2.1. Detection Time and Report Time

We can observe that detection time follows a law of the kind $t_D(n_m) = a n_m^{-b}$ where a and b are constants (Figure 4(a)). The scaling relationships happens to be independent of the model and thus it is the same for all four different mobility models except for the values of the constants a and b. Notice also that, if we add less than 5% of mobile sensors then the sensor density is too low, the space is not well covered, and it leads to a very high detection time. However, the detection time drops sharply, and the performance gain is not so prominent anymore when sensor density reaches 10% of RD. The diminishing returns is accentuated by the fact that the scaling relationship is a long tailed distribution, thus the asymptotic behavior is reached much more slowly than in the case of an exponential decay. Report time $t_R(n_m)$ is also defined by an equation similar to $t_D(n_m)$ and again followed by all mobility models (Figure 4(b)). The mobility model performance with respect to Detection Time and Report Time metrics can be summarized as follows:

- Song's human mobility model is the one performing the worst, because sensors are not free to move, but are constrained by the preferential return. In fact, the sensors tend to stay close to the original position of deployment, and only slowly diffuse in the environment. The waiting time between jumps further slow down the diffusion process. If we distribute the mobile sensors uniformly in the environment to counteract the preferential return, the performance is closer to the one of Lévy walk, but still worse because the preferential return limits the ability of the sensors to visit new locations and to meet new sensors (Figure 6). However, the non uniform distribution and the preferential return do not impact the scaling of performance when the distance between sink and event, which is linear (Figure 7(a)).
- There is no big difference between Lévy walk and CTRW, that is waiting time does not heavily impact detection and report time. This is the result of having many sensors, so at each point in time a large fraction of them is moving.
- We might be tempted to argue that the impact of waiting time to performance in a SNoS is relatively limited. In fact, the wait-time distribution is such that most pauses have a limited time length with few long pauses, therefore at any instant in time most sensors are able to move. However, increasing the wait-time cut off degrade performance in a subtle way, because it impacts the variance of performance, which is relevant for delivery under time constraints.
- The wait-time cutoff is not able to explain the performance difference between CTRW and Song's model. The key is the presence of preferential return mechanism in Song's model. This leads to hot spots of visitations, and if the sink is not in one of these hot spots then the chance of a sensor finding it decreases over time.
- The Brownian mobility model has a bottleneck in the length of the jump once sensors are spread across the entire environment as shown by the asymmetry between detection time and report time (Figure 4); in the former, Brownian motion is faster than Song's model, while in the latter it is slightly slower. Given these considerations, there are reasons to be skeptical on the practical relevance of theoretical results using random walks like Brownian motion or random way-point, because these models differs significantly from human mobility patterns.
- As the sensor density increases the difference in performance among different mobility models also decreases becoming essentially irrelevant. We must stress that this decrease is not a consequence of the interaction with static sensors, because static sensors are almost irrelevant given their small sensing radius and limited number (Figure 9). Instead, the phenomenon is due to the influence that higher density has on mobile sensor inter-contact time. This result is important, because it tells us that in a *sufficiently dense network*, the performance is not bounded by a specific mobility model but by sensor speed and protocol in charge of delivering information.

4.2.2. Network Coverage Area

If we look at "coverage" as the main factor, the mobility model used has a greater impact, because the covered area depends heavily from the mobility of sensors. As sensors move, they are able to "see" areas of the environment that would otherwise not be seen. This accounts for a fraction of the area visited/covered by the mobile sensors. Figure 5 shows the fraction of the area covered at the end of the simulation (i.e. $t_{sim} = t_D + t_R$) as we increase the number of mobile sensors. The performance measured with respect to Coverage Area metric can be summarized as:

- Lévy walk and CTRW have the best coverage since there is no preferential return or cut off in jump length thus sensors spread rapidly in the environment following a super diffusive process [36].

- Song's model behaves like a sub-diffusive process [41], thus should have a worse coverage than Brownian model. The fraction of covered area versus the number of mobile sensors for Song's mobility model follows a law of the kind $a + b\ln(n_m)$, where a and b are constants (Figure 5(a)). This is caused by a saturation process. That is, it is increasingly harder to achieve a greater coverage by simply adding more sensors, because sensors are not uniformly distributed in the environment and they are constrained by preferential return; hence the edges of the environment are less likely to be covered compared to locations in the center of the environment. This property combined with a spatial distribution of people that is not uniform hinder the performance of the whole network because not all areas will be covered with the same intensity. However, we must clarify that, if we allow the simulation to run indefinitely, there will be a time instant t_{sim} such that the fraction of covered area will be $f_a(t_{sim}) = 1$. If we remove such constraints by deploying the sensors uniformly, we can observe that individual mobility model has performance that are similar to the one of Lévy walk and CTRW (Figure 7(b)).

- If we assume for a moment that sensors are distributed randomly and uniformly in the space, then the fraction of the area $A_s(t)$ covered by sensors at each instant in time t is:

$$A_s(t) = 1 - e^{-\lambda\pi r^2}, \tag{12}$$

  where λ is the density of the Poisson process used to deploy sensors and r is sensor radius [10]. Notice also that sensors tend to spread in the environment and eventually reach an uniform distribution despite the initial exponential distribution. Then, as we can see in the equation for $A_s(t)$ the exponential decay depends linearly on the density but quadratically on the sensor radius, thus the fraction of covered area is influenced more by sensor radius than sensor density.

- Due to the large amount of people and high density of metropolitan cities, with just a relatively small percentage of the population it is possible to build a SNoS with a very good coverage. However, performance relative to t_D and t_R does not scale as well as we increase the number of mobile sensors hence it poses a problem, because it limits the efficacy of the network and its usefulness to cases where a high delay of information delivery can be tolerated (e.g., tracking of animals, street/place mapping).

4.2.3. Data Delivery Rate

In order to be functional a network must be able to deliver the information between nodes. Moreover to perform adequately, it must do it in a reasonable time. The delivery rate, which is defined as the fraction of events reported within a time constraint, is able to capture these two aspects. Thus, we studied the behavior of such metric as a function of the reference density RD for different time constraints (Figure 8). We observed the following:

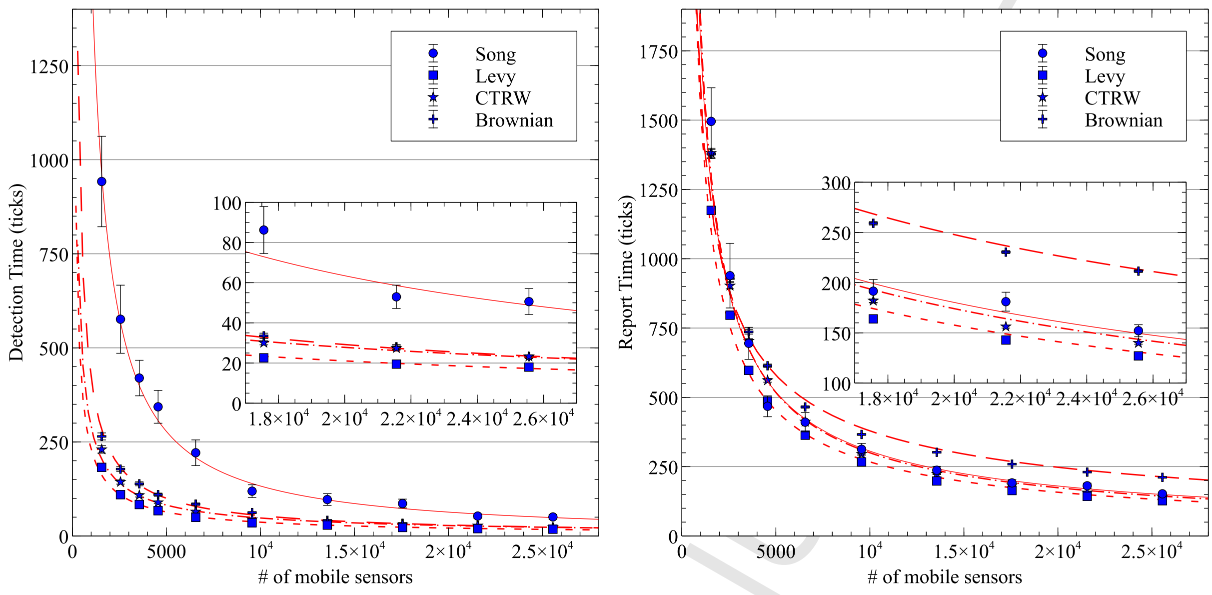
- Once a small density threshold is exceeded, the event delivery ratio increases sharply; this could depend on the nature of the epidemic spreading of the event. That is, the percolation threshold of the network is small, thus even if very few sensors find the event directly, it rapidly spreads over the network to reach the sink.

- Once we reach an upper threshold, the event delivery ratio tends to saturate. Therefore, it is not convenient to aim for a perfect delivery ratio, because the effort to obtain it (the number of mobile sensors required) grows faster than the percentage of reported events in time t_{sim}.

- The first two observations match well with the fact that data is fitted very well by a Gompertz function [70]. Gompertz function is a sigmoid function where the growth is slower at the start and end, but the upper asymptote of the function is approached much more gradually by the curve than the lower asymptote. In contrast, in the simple logistic function both asymptotes are approached by the curve symmetrically. The equation of Gompertz function re-parameterized according to [71] is:

$$A \cdot \exp\left\{ -\exp\left[\frac{\mu \cdot e}{A}(\lambda - \%RD) + 1\right]\right\}, \tag{13}$$

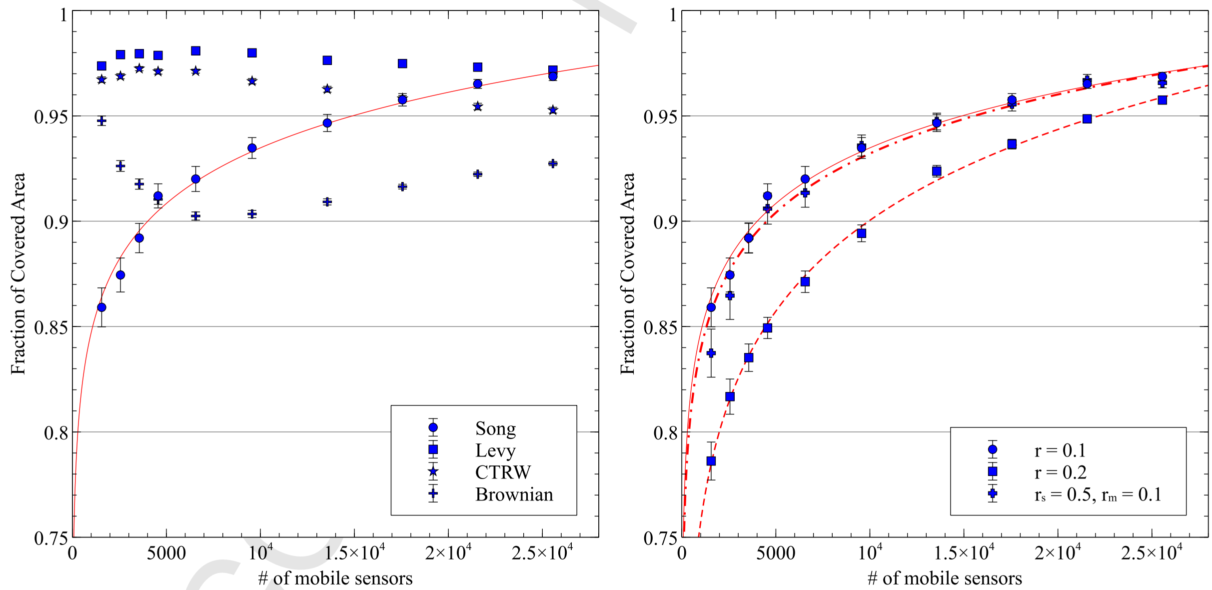
  where $e = \exp(1)$, A is the upper asymptote, λ is the length of the lag phase (the phase before the exponential growth), and μ is the max growth rate. In our case, $A = 100$ and λ is a density, instead of a time, that may represent the percolation threshold of the network. Both λ and μ depend on the time constraint on t_{sim}.



(a) Detection Time

(b) Report Time

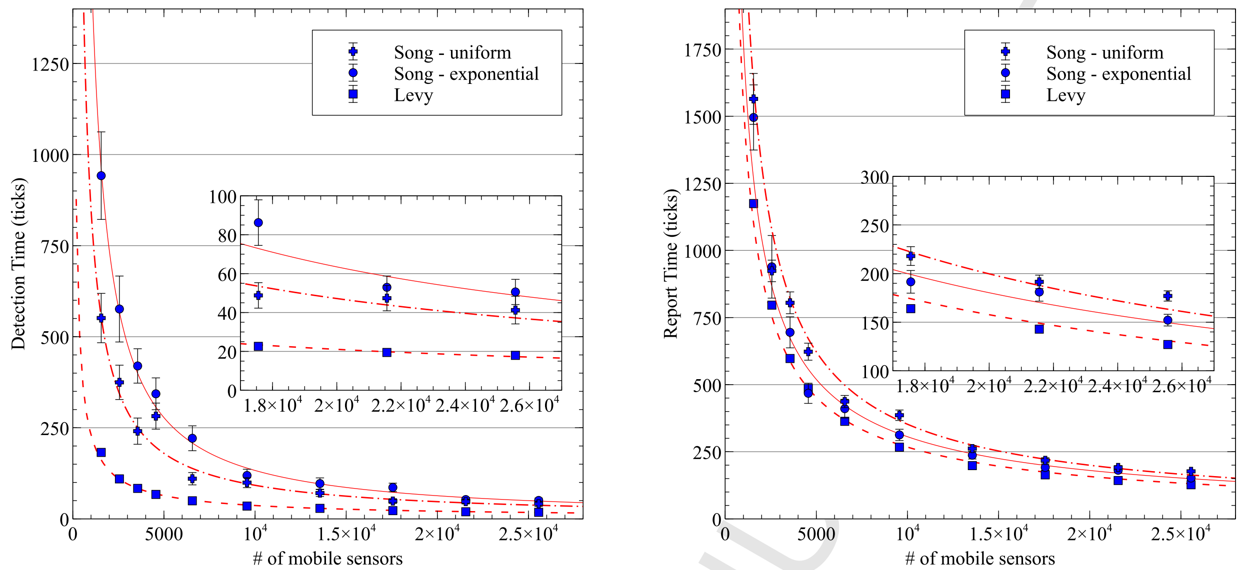
Figure 4: Detection Time t_D follows the law $t_D \sim a n_m^b$. Report Time t_R follows the law $t_R \sim c n_m^d$. The red lines represent the fit lines following such power law equations. All mobility models follow the same scaling relationship, which is a consequence of sensor spatial distribution instead.



(a) Fraction of Area when $r = 0.1$

(b) Fraction of Area IM model

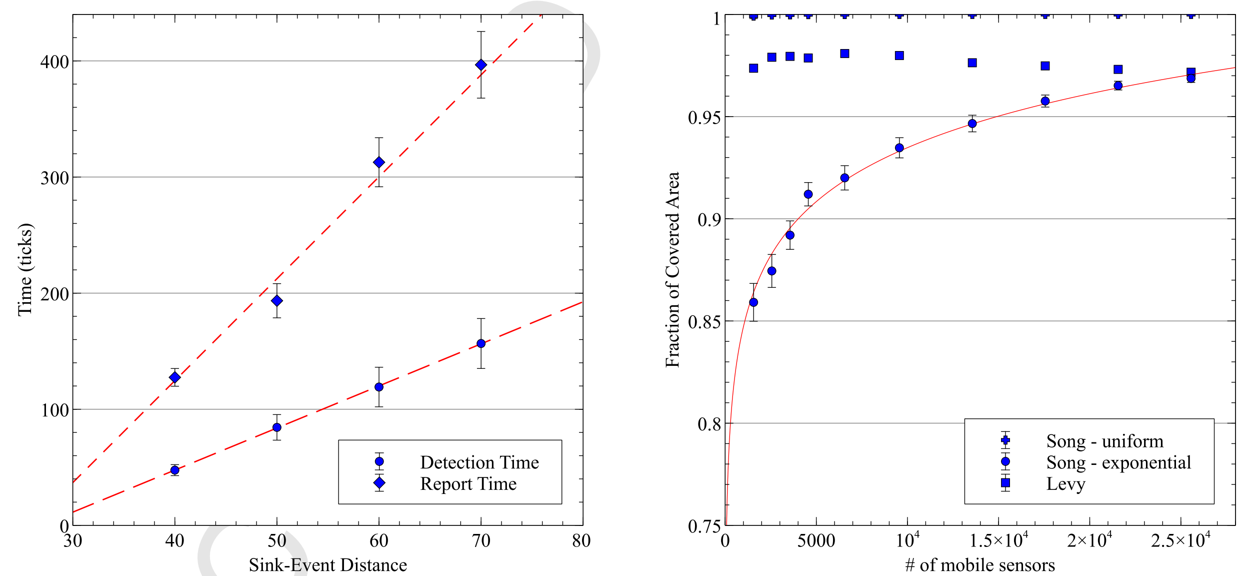
Figure 5: Maximum fraction of covered area versus n_m for different mobility models. In Song's model the Fraction of Covered Area $f_a(t_{sim})$ is proportional to $f_a(t_{sim}) \sim a + b ln(n_m)$. This means that a saturation occurs while we approach a total coverage. The area covered when $r = 0.2$ is smaller because the time of the simulation t_{sim} is shorter, however it retains the same scaling law.



(a) Detection Time

(b) Report Time

Figure 6: (a) The detection time both with non uniform and uniform mobile sensor distribution. As a comparison metric we also included the Lévi walk. The uniform distribution performs much better than exponential because the density of sensors at the periphery, where the event is located, is higher than in the exponential distribution case. However, the preferential return slow down the visitation of new locations, therefore IM model is still slower than Lv́y. (b) The report time actually gets worse because while there is an improvement on the outer periphery, there is also a decrease in performance when forwarding the event closer to the center of the environment. There, the density of sensors actually decreased compared to the case with exponential distribution.



(a) Detection Time and Report Time

(b) Fraction of Area

Figure 7: (a) The scaling of performance when the distance between sink and event changes is linear (b) The fraction of area covered when the sensors are distributed uniformly is close to 100%, as we would expect given the number of mobile sensors deployed.

- Unfortunately, the wait-time cutoff has a *big* impact on the delivery ratio under time constraints (Figure 8). It can be explained by the fact that it increases both the average value (although slightly) and the variance (almost doubles) of t_{sim}, thus the network does not perform consistently. The variance of t_{sim} is negatively impacted by the long tailed wait-time distribution. This behavior represents a major problem since it affects the max grow rate μ, which imply that the delivery ratio increases slower than wanted as we add sensors. The solution is either to increase sensor density (increase costs) or relax time constraints (accept lower performance) or both. However, performance are bound to saturate. Therefore, we introduce some design principles in Section 4.3 to improve performance.
- The density required to achieve the necessary performance tells us the fraction of the population of the city that should be involved to build the desired network. This is especially important since it imposes a lower limit on the size and density of the population of the city. We can expect that in small cities it is not possible to achieve some acceptable performance in a SNoS. Data for some of the biggest cities in the USA is shown in Table 1; it should be noted that USA cities have usually a lower density than other metropolis especially if compared to South America or Asia metropolis, thus they represent a worst case scenario.

Table 1: Densities D of some of the biggest cities in the USA with the density required to achieve the sensor density of the case $\%RD = 10$. Even for some big cities a relatively large percentage of the population need to be involved thus the deployment and implementation of the SNoS must be carefully designed.

City	$D(ppl/km^2)$	$10\frac{RD}{D}$
New York, NY-NJ-CT	1800	11.11
Los Angeles, CA	2400	8.33
Chicago, IL-IN-WI	1300	15.38
Philadelphia, PA-NJ-DE-MD	1100	18.18
Boston, MA-NH-RI	800	25.00
Miami, FL	1800	11.11

4.3. *Improving Performance*

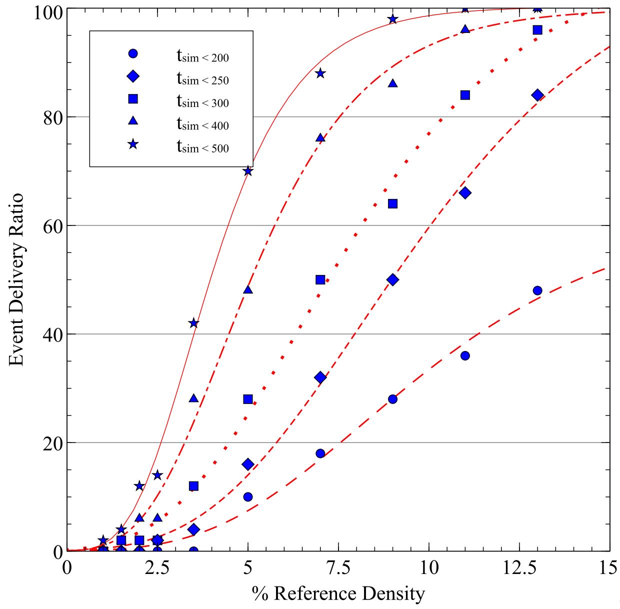
Network performance is of great interest and presently could be the biggest obstacle in the actual implementation of a SNoS because it seems that network latency remains high no matter how many mobile sensors we use. In fact, authors in [72] show that these kinds of networks expect transmission latency in the order of minutes to about half an hour. However, our simulations clearly show that the delay is between 2 and 4 hours even at the highest density, and independently of the mobility model. In order to overcome this limitation, we have worked on approaches that could vastly improve this aspect. There are two complementary approaches: optimize the deployment of the infrastructure and work on sensor properties.

4.3.1. *Improving the Infrastructure*

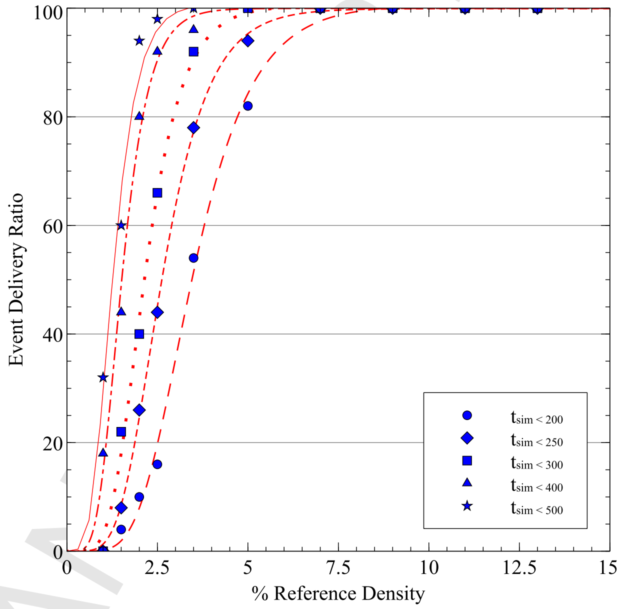
First, since most of mobile sensors are near the center of the city, it is unnecessary to have static sensors in that place given that it is already covered very well. Thus we should move some of those sensors in periphery area. The amount N_s of static sensors which can be relocated is

$$N_s = \frac{n_s * \pi * \mathscr{D}^2}{4A}, \tag{14}$$

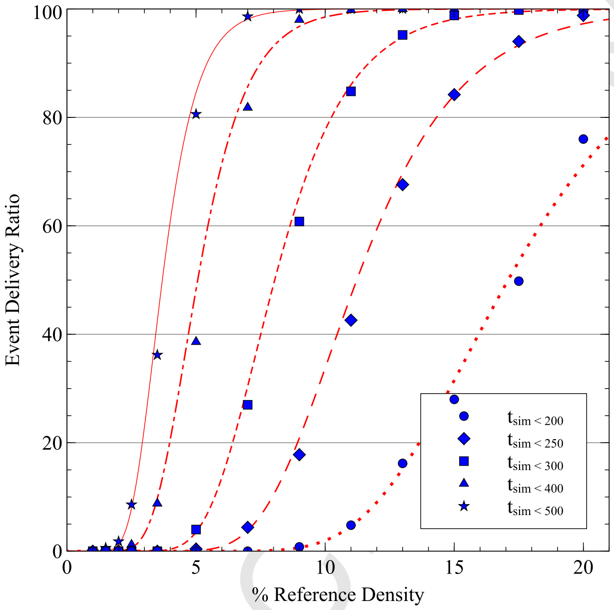
where symbols have the meaning described in Section 3. This approach leads to cost savings if we choose not to deploy those expendable static sensors, or it leads to improvements in performance and coverage if we move the static



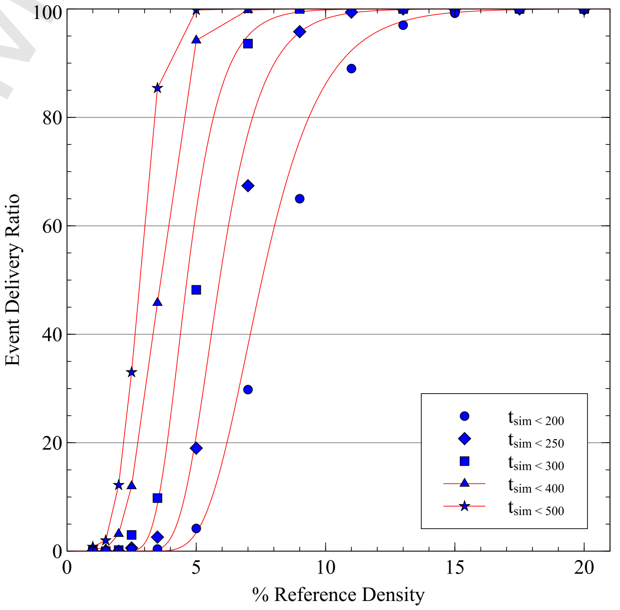
(a) Delivery Ratio of IM model with r = 0.1



(b) Delivery Ratio of IM model with r = 0.2



(c) Delivery Ratio of Wiener Mobility Model



(d) Delivery Ratio of Lévy Mobility Model

Figure 8: Event delivery ratio is fit very well by the Gompertz function. Song's model performance can be improved to match Lévy walk by doubling sensor radius. Brownian mobility model perform better than IM due to less variance of t_{sim}.

sensors in the periphery. Another technique could consist in building short paths from the periphery to the center so that if they are hit by a sensor that knows about the event, it immediately propagates where the density of mobile sensors is greater thus greatly reducing report time.

Second, due to the urban population density distribution and the greater relevance of report time over detection time, to maximize performance. The best sink deployment is in the center of the city. This also leads to more uniform distance from event to sink in case the event is placed randomly in the space thus reducing the variance of t_D and t_R. However, the model for population density we used does not take into account multi-centered cities [73]; in this case, we recommend to place a sink in each and every center.

Third, there are two thresholds. A lower bound threshold that ensures that mobile sensors achieve a high enough density and a higher bound threshold over which it is not worth the addition of more mobile sensors because they do not give a significant improvement of performance. Of course, these thresholds could vary as a function of the cost of mobile sensors, that is, there still exists a tradeoff between cost and performance. Moreover, our analysis does not take into account sensor faults, that is part of the sensors may be going offline with a given probability. Deployment must take in account some over-provisioning.

4.3.2. Improving Sensor Properties

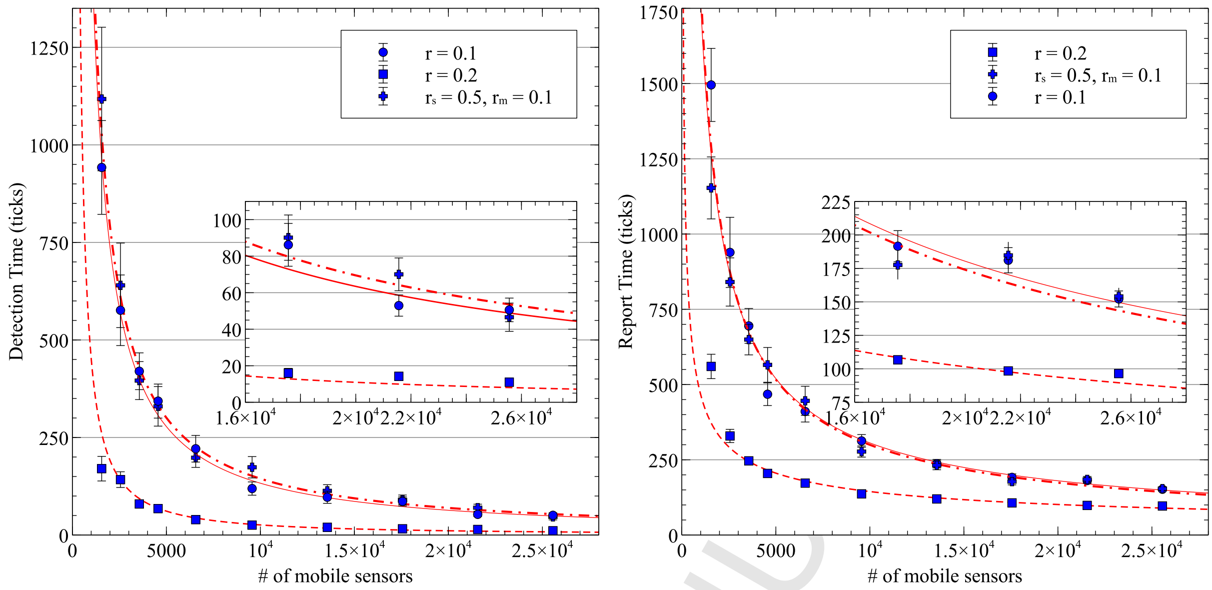
The first idea is derived from the fact that the presented setup uses mobile and static sensors that have the same connectivity range. In this configuration, static sensors have no more relevant meaning, because they do not significantly participate in detection and delivery, they simply do not cover any meaningful area. Instead, we could deploy expensive static sensors that exploit the current infrastructure, so that they do not have power constraints, and they can have a bigger sensing radius. This way, static sensors can function as the gateways for data spreading to mobile sensors. However, note how performance is actually unaffected (Figure 9 and Figure 5(b)). The lack of performance improvement is due to the fact that the communication between mobile sensors and static sensors is asymmetric. In fact, while static sensors can spread the event at a further distance to mobile sensors, the static sensors must be in range of a mobile sensor to receive the information. Therefore, the probability of a mobile sensor to spread the information to a static sensor is unaffected, and the impact of static sensors remain statistically insignificant. In order for such setup to work it is required that sensors have beam-forming capabilities [74].

The second idea, which is complementary to the first one, is to rely on the current network infrastructure for static sensors so that they constitute a connected network, working at "0 latency" and representing a short path to the sinks. This approach has also the benefit to vastly reduce the memory pressure on mobile sensors because they can hand off as much data as they can to the static sensors. Static sensors then become responsible for forwarding data to the sink using a wired infrastructure. In a city with many Wi-Fi hotspots, we believe this may be very effective.

Finally, we can mitigate the impact of sensor radius using a different technology for wireless communications. Notice that doubling sensor radius (from $r = 0.1$ to $r = 0.2$) leads to an almost 4 times performance improvement when sensor density is low, as predicted by Equation (12). However, such gain becomes linear when sensor density increases (Figure 9). Indeed, Bluetooth is infrequently used by smartphone users, while most of the time Wi-Fi is turned on and hence more accessible. Moreover, in these kinds of devices, battery consumption is not something that is really enforced since users charge their devices almost daily. Wi-Fi has a greater range and performance than Bluetooth and it retains backward compatibility while improving performance in each new version. It also allows the creation of ad-hoc networks and development of automatic configuration protocols (e.g., Zeroconf). Ubiquity and current availability make Wi-Fi the best candidate for an efficient SNoS.

5. Conclusions and Future Works

We found that the detection time t_D and the report time t_R performance follow a scaling law relationship in the number of mobile sensors and that in dense networks the differences in performance among mobility models fade out. Moreover, we showed that the sensor radius has a bigger impact on performance than sensor density, thus it should be maximized whenever possible. Furthermore, the delivery ratio to the sink is negatively influenced by wait time and preferential return, thus the Individual Mobility model, which is the closest to human mobility, has a performance that is significantly worse than the other mobility models tested. Finally, we propose some design patterns to improve performance.



(a) Detection Time (b) Report Time

Figure 9: Effect of sensor radius on SNoS performance, when sensors move accordingly to Song's model. We can increase radius of every sensor, or we can increase only sensor radius r_s of static sensors. Detection time performance is not influenced by an increase in sensor radius of static sensors, unless the event is deployed in their sensing range, which is not the case here.

Given the great flexibility of the model we implemented, future works will focus on three aspects: different mobility models and spatial distributions combinations that could lead to better performances; scenarios where these combinations can be applied to obtain real-world benefits (e.g., in a forest or in the ocean we could attach sensors to animals, in a battlefield sensors might be air-dropped or attached to soldiers); protocols that take into account the patterns of human mobility and data aggregation [75] need to be developed to improve information spreading efficiency while maintaining performance.

References

[1] F. Nack, An overview on wireless sensor networks, Institute of Computer Science (ICS), Freie Universität Berlin.

[2] I. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, A survey on sensor networks, Communications Magazine, IEEE 40 (8) (2002) 102–114. doi:10.1109/MCOM.2002.1024422.

[3] D. Bri, M. Garcia, J. Lloret, P. Dini, Real deployments of wireless sensor networks, in: Sensor Technologies and Applications, 2009. SENSORCOMM'09. Third International Conference on, IEEE, IEEE, Washington, DC, USA, 2009, pp. 415–423.

[4] L. Atzori, A. Iera, G. Morabito, The internet of things: A survey, Computer networks 54 (15) (2010) 2787–2805.

[5] B. Guo, Z. Yu, X. Zhou, D. Zhang, Opportunistic iot: Exploring the social side of the internet of things, in: Computer Supported Cooperative Work in Design (CSCWD), 2012 IEEE 16th International Conference on, IEEE, 2012, pp. 925–929.

[6] E. Cho, S. A. Myers, J. Leskovec, Friendship and mobility: user movement in location-based social networks, in: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2011, pp. 1082–1090.

[7] D. Wang, D. Pedreschi, C. Song, F. Giannotti, A.-L. Barabasi, Human mobility, social ties, and link prediction, in: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2011, pp. 1100–1108.

[8] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, R. A. Peterson, H. Lu, X. Zheng, M. Musolesi, K. Fodor, G.-S. Ahn, The rise of people-centric sensing, Internet Computing, IEEE 12 (4) (2008) 12–21.

[9] D. Estrin, Participatory sensing: applications and architecture, Internet Computing, IEEE 14 (1) (2010) 12–42.

[10] B. Liu, P. Brass, O. Dousse, P. Nain, D. Towsley, Mobility improves coverage of sensor networks, in: Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing, MobiHoc '05, ACM, New York, NY, USA, 2005, pp. 300–308. doi:10.1145/1062689.1062728.
URL `http://doi.acm.org/10.1145/1062689.1062728`

[11] E. Natalizio, V. Loscrí, Controlled mobility in mobile sensor networks: advantages, issues and challenges, Telecommunication Systems 52 (4) (2013) 2411–2418. doi:10.1007/s11235-011-9561-x.
URL `http://dx.doi.org/10.1007/s11235-011-9561-x`

[12] T.-L. Chin, P. Ramanathan, K. K. Saluja, Analytic modeling of detection latency in mobile sensor networks, in: Proceedings of the 5th International Conference on Information Processing in Sensor Networks, IPSN '06, ACM, New York, NY, USA, 2006, pp. 194–201. doi:10.1145/1127777.1127810.
URL `http://doi.acm.org/10.1145/1127777.1127810`

[13] D. Wang, Z. Wen, H. Tong, C.-Y. Lin, C. Song, A.-L. Barabási, Information spreading in context, in: Proceedings of the 20th International Conference on World Wide Web, WWW '11, ACM, New York, NY, USA, 2011, pp. 735–744. doi:10.1145/1963405.1963508.
URL `http://doi.acm.org/10.1145/1963405.1963508`

[14] A. Einstein, On the theory of brownian movement, Annalen der Physik 19 (4) (1906) 371–381.

[15] B. B. Mandelbrot, The fractal geometry of nature, Macmillan, San Francisco, CA, 1983.

[16] I. Rhee, M. Shin, S. Hong, K. Lee, S. J. Kim, S. Chong, On the levy-walk nature of human mobility, IEEE/ACM transactions on networking (TON) 19 (3) (2011) 630–643.

[17] E. W. Montroll, G. H. Weiss, Random walks on lattices. ii, Journal of Mathematical Physics 6 (1965) 167.

[18] C. Song, Z. Qu, N. Blumm, A.-L. Barabási, Limits of predictability in human mobility, Science 327 (5968) (2010) 1018–1021. arXiv:http://www.sciencemag.org/content/327/5968/1018.full.pdf, doi:10.1126/science.1177170.
URL `http://www.sciencemag.org/content/327/5968/1018.abstract`

[19] C. Song, T. Koren, P. Wang, A.-L. Barabasi, Modelling the scaling properties of human mobility, Nat Phys 6 (10) (2010) 818–823.
URL `http://dx.doi.org/10.1038/nphys1760`

[20] A. Howard, M. J. Matarić, G. S. Sukhatme, Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem, in: Distributed Autonomous Robotic Systems 5, Springer, Fukuoka, Japan, 2002, pp. 299–308.

[21] Y. Zou, K. Chakrabarty, Sensor deployment and target localization based on virtual forces, in: INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies, Vol. 2, IEEE, San Francisco, Ca, USA, 2003, pp. 1293–1303. doi:10.1109/INFCOM.2003.1208965.

[22] H. Zhang, J. C. Hou, Maintaining sensing coverage and connectivity in large sensor networks, Ad Hoc & Sensor Wireless Networks 1 (1-2) (2005) 89–124.

[23] N. Bicocchi, M. Mamei, F. Zambonelli, Self-organizing spatial regions for sensor network infrastructures, in: Advanced Information Networking and Applications Workshops, 2007, AINAW '07. 21st International Conference on, Vol. 2, IEEE, Washington, DC, USA, 2007, pp. 66–71. doi:10.1109/AINAW.2007.323.

[24] A. Aziz, Y. Sekercioglu, P. Fitzpatrick, M. Ivanovich, A survey on distributed topology control techniques for extending the lifetime of battery powered wireless sensor networks, Communications Surveys Tutorials, IEEE 15 (1) (2013) 121–144. doi:10.1109/SURV.2012.031612.00124.

[25] S. Chellappan, W. Gu, X. Bai, D. Xuan, B. Ma, K. Zhang, Deploying wireless sensor networks under limited mobility constraints, Mobile Computing, IEEE Transactions on 6 (10) (2007) 1142–1157. doi:10.1109/TMC.2007.1032.

[26] B. Liu, O. Dousse, P. Nain, D. Towsley, Dynamic coverage of mobile sensor networks, Parallel and Distributed Systems, IEEE Transactions on 24 (2) (2013) 301–311.

[27] D. R. Bild, Y. Liu, R. P. Dick, Z. M. Mao, D. S. Wallach, Using predictable mobility patterns to support scalable and secure manets of handheld devices, in: Proceedings of the Sixth International Workshop on MobiArch, MobiArch '11, ACM, New York, NY, USA, 2011, pp. 13–18. doi:10.1145/1999916.1999921.
URL `http://doi.acm.org/10.1145/1999916.1999921`

[28] A. S. Cacciapuoti, F. Calabrese, M. Caleffi, G. Di Lorenzo, L. Paura, Human-mobility enabled networks in urban environments: Is there any (mobile wireless) small world out there?, Ad Hoc Networks 10 (8) (2012) 1520–1531.

[29] B. Guo, D. Zhang, Z. Wang, Z. Yu, X. Zhou, Opportunistic iot: Exploring the harmonious interaction between human and the internet of things, Journal of Network and Computer Applications 36 (6) (2013) 1531–1539.

[30] B. Guo, Z. Wang, Z. Yu, Y. Wang, N. Y. Yen, R. Huang, X. Zhou, Mobile crowd sensing and computing: The review of an emerging human-powered sensing paradigm, ACM Computing Surveys (CSUR) 48 (1) (2015) 7.

[31] J.-S. Lee, B. Hoh, Dynamic pricing incentive for participatory sensing, Pervasive and Mobile Computing 6 (6) (2010) 693–708.

[32] J. Rula, F. E. Bustamante, Crowd (soft) control: moving beyond the opportunistic, in: Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications, HotMobile '12, ACM, New York, NY, USA, 2012, pp. 3:1–3:6. doi:10.1145/2162081.2162086.
URL `http://doi.acm.org/10.1145/2162081.2162086`

[33] L. Jaimes, I. Vergara-Laurens, A. Raij, A survey on incentive techniques for mobile crowd sensing, Internet of Things Journal, IEEE PP (99) (2015) 1–1. doi:10.1109/JIOT.2015.2409151.

[34] M. Tomasini, F. Zambonelli, A. Brayner, R. Menezes, Evaluating the performance of social networks of sensors under different mobility models, Social Computing / IEEE International Conference on Privacy, Security, Risk and Trust, 2010 IEEE International Conference on 0 (2013) 397–402. doi:http://doi.ieeecomputersociety.org/10.1109/SocialCom.2013.62.

[35] O. Dousse, C. Tavoularis, P. Thiran, Delay of intrusion detection in wireless sensor networks, in: Proceedings of the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc '06, ACM, New York, NY, USA, 2006, pp. 155–165. doi:10.1145/1132905.1132923.
URL `http://doi.acm.org/10.1145/1132905.1132923`

[36] A. A. Dubkov, B. Spagnolo, V. V. Uchaikin, Lévy flight superdiffusion: an introduction, International Journal of Bifurcation and Chaos 18 (09) (2008) 2649–2672.

[37] G. M. Viswanathan, V. Afanasyev, S. V. Buldyrev, E. J. Mur-phy, P. A. Prince, H. E. Stanley, Lévy flight search patterns of wandering albatrosses, Nature 381 (1996) 413–415.

[38] G. Ramos-Fernandez, J. L. Mateos, O. Miramontes, G. Cocho, H. Larralde, B. Ayala-Orozco, Lévy walk patterns in the foraging move- ments of spider monkeys (ateles geoffroyi), Behav. Ecol. Sociobiol. 55 (3) (2004) 223–230.

[39] M. F. Shlesinger, J. Klafter, Y. Wong, Random walks with infinite spatial and temporal moments, Journal of Statistical Physics 27 (3) (1982) 499–512.

[40] D. Brockmann, L. Hufnagel, T. Geisel, The scaling laws of human travel, Nature 439 (7075) (2006) 462–465.

[41] M. C. Gonzalez, C. A. Hidalgo, A.-L. Barabasi, Understanding individual human mobility patterns, Nature 453 (7196) (2008) 779–782. URL http://dx.doi.org/10.1038/nature06958
[42] C. Song, D. Wang, A. Barabasi, Connections between Human Dynamics and Network Science (2012). arXiv:arXiv:1209.1411v2. URL http://arxiv.org/abs/1209.1411
[43] C. Boldrini, A. Passarella, Hcmm: Modelling spatial and temporal properties of human mobility driven by users' social relationships, Comput. Commun. 33 (9) (2010) 1056–1074. doi:10.1016/j.comcom.2010.01.013. URL http://dx.doi.org/10.1016/j.comcom.2010.01.013
[44] D. Fischer, K. Herrmann, K. Rothermel, Gesomo: A general social mobility model for delay tolerant networks, in: Mobile Ad-hoc and Sensor Systems (MASS), 2010 IEEE 7th International Conference on, IEEE, San Francisco, CA, 2010, pp. 99–108. doi:10.1109/MASS.2010.5663959.
[45] D. Karamshuk, C. Boldrini, M. Conti, A. Passarella, Human mobility models for opportunistic networks, Communications Magazine, IEEE 49 (12) (2011) 157–165.
[46] S. McCanne, S. Floyd, K. Fall, K. Varadhan, et al., Network simulator ns-2 (1997).
[47] T. R. Henderson, M. Lacage, G. F. Riley, C. Dowell, J. Kopena, Network simulations with the ns-3 simulator, SIGCOMM demonstration 15 (2008) 17.
[48] A. Varga, R. Hornig, An overview of the omnet++ simulation environment, in: Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, p. 60.
[49] S. Jain, K. Fall, R. Patra, Routing in a delay tolerant network, in: Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, SIGCOMM '04, ACM, New York, NY, USA, 2004, pp. 145–158. doi:10.1145/1015467.1015484. URL http://doi.acm.org/10.1145/1015467.1015484
[50] E. P. Jones, L. Li, J. K. Schmidtke, P. A. Ward, Practical routing in delay-tolerant networks, IEEE Transactions on Mobile Computing 6 (8) (2007) 943–959.
[51] A. Keränen, J. Ott, T. Kärkkäinen, The one simulator for dtn protocol evaluation, in: Proceedings of the 2Nd International Conference on Simulation Tools and Techniques, Simutools '09, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, Belgium, 2009, pp. 55:1–55:10. doi:10.4108/ICST.SIMUTOOLS2009.5674. URL http://dx.doi.org/10.4108/ICST.SIMUTOOLS2009.5674
[52] A. Islam, M. Waldvogel, Reality-check for dtn routing algorithms, in: 2008 The 28th International Conference on Distributed Computing Systems Workshops, IEEE, 2008, pp. 204–209.
[53] A. Petz, J. Enderle, C. Julien, A framework for evaluating dtn mobility models, in: Proceedings of the 2nd International Conference on Simulation Tools and Techniques, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009, p. 94.
[54] C. Chilipirea, A.-C. Petre, C. Dobre, F. Pop, G. Suciu, A simulator for opportunistic networks, Concurrency and Computation: Practice and Experience.
[55] M. Tomasini, F. Zambonelli, R. Menezes, Using patterns of social dynamics in the design of social networks of sensors, in: Proceedings of the 2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, GREENCOM-ITHINGS-CPSCOM '13, IEEE Computer Society, Washington, DC, USA, 2013, pp. 685–692. doi:10.1109/GreenCom-iThings-CPSCom.2013.125. URL http://dx.doi.org/10.1109/GreenCom-iThings-CPSCom.2013.125
[56] C. Clark, Urban population densities, Journal of the Royal Statistical Society. Series A (General) 114 (4) (1951) 490–496.
[57] www.demographia.com, Demographia world urban areas, 9th annual edition, Tech. rep., Demographia (Mar. 2013). URL http://www.demographia.com/db-worldua.pdf
[58] R. Knoblauch, M. Pietrucha, M. Nitzburg, Field studies of pedestrian walking speed and start-up time, Transportation Research Record: Journal of the Transportation Research Board (1538) (1996) 27–38.
[59] M. Moussaïd, N. Perozo, S. Garnier, D. Helbing, G. Theraulaz, The walking behaviour of pedestrian social groups and its impact on crowd dynamics, PloS one 5 (4) (2010) e10047.
[60] A. Vahdat, D. Becker, et al., Epidemic routing for partially connected ad hoc networks (2000).
[61] U. Wilensky, Netlogo (1999). URL http://ccl.northwestern.edu/netlogo/
[62] M. Tomasini, B. Mohammed M., R. Menezes, Social network of sensors (snos). URL http://modelingcommons.org/browse/one_model/3991
[63] R. N. Mantegna, H. E. Stanley, Stochastic process with ultraslow convergence to a gaussian: the truncated lévy flight, Physical Review Letters 73 (22) (1994) 2946.
[64] H. Barbosa, F. B. de Lima-Neto, A. Evsukoff, R. Menezes, The effect of recency to human mobility, EPJ Data Science 4 (1) (2015) 1–14.
[65] A. Clauset, C. R. Shalizi, M. E. Newman, Power-law distributions in empirical data, SIAM review 51 (4) (2009) 661–703.
[66] T. Spyropoulos, K. Psounis, C. S. Raghavendra, Spray and wait: an efficient routing scheme for intermittently connected mobile networks, in: Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking, ACM, 2005, pp. 252–259.
[67] B. Mahmood, M. Tomasini, R. Menezes, Social-driven information dissemination for mobile wireless sensor networks, Sensors & Transducers 189 (6) (2015) 1.
[68] Ó. Helgason, S. T. Kouyoumdjieva, G. Karlsson, Does mobility matter?, in: Wireless On-demand Network Systems and Services (WONS), 2010 Seventh International Conference on, IEEE, 2010, pp. 9–16.
[69] Ó. Helgason, S. T. Kouyoumdjieva, G. Karlsson, Opportunistic communication and human mobility, Mobile Computing, IEEE Transactions on 13 (7) (2014) 1597–1610.
[70] B. Gompertz, On the nature of the function expressive of the law of human mortality, and on a new mode of determining the value of life

contingencies, Philosophical Transactions of the Royal Society of London 115 (1825) pp. 513–583.
URL http://www.jstor.org/stable/107756

[71] M. Zwietering, I. Jongenburger, F. Rombouts, K. Van't Riet, Modeling of the bacterial growth curve, Applied and environmental microbiology 56 (6) (1990) 1875–1881.

[72] A. Zanella, N. Bui, A. P. Castellani, L. Vangelista, M. Zorzi, Internet of things for smart cities, IEEE Internet of Things Journal.

[73] D. A. Griffith, Modelling urban population density in a multi-centered city, Journal of Urban Economics 9 (3) (1981) 298–310. doi:10.1016/0094-1190(81)90029-2.
URL http://www.sciencedirect.com/science/article/pii/0094119081900292

[74] J. Li, P. Stoica, Robust adaptive beamforming, Vol. 88, John Wiley & Sons, 2005.

[75] A. Brayner, A. Lopes, D. Meira, R. Vasconcelos, R. Menezes, An adaptive in-network aggregation operator for query processing in wireless sensor networks, J. Syst. Softw. 81 (2008) 328–342.