

Document downloaded from:

<http://hdl.handle.net/10251/119573>

This paper must be cited as:

Hernández-Orallo, E.; Cano, J.; Tavares De Araujo Cesariny Calafate, CM.; Manzoni, P. (2018). FALCON: A New Approach for the Evaluation of Opportunistic Networks. *Ad Hoc Networks*. 81:109-121. <https://doi.org/10.1016/j.adhoc.2018.07.004>



The final publication is available at

<http://doi.org/10.1016/j.adhoc.2018.07.004>

Copyright Elsevier

Additional Information

# FALCON: A New Approach for the Evaluation of Opportunistic Networks

Enrique Hernández-Orallo\*, Juan Carlos Cano, Carlos T. Calafate, Pietro Manzoni

*Departamento de Informática de Sistemas y Computadores. Universitat Politècnica de València. Spain.*

---

## Abstract

Evaluating the performance of opportunistic networks with a high number of nodes is a challenging problem. Analytical models cannot provide a realistic evaluation of these networks, and simulations can be very time-consuming, sometimes requiring even weeks only to provide the results of a single scenario.

In this paper, we present a fast evaluation model called FALCON (*Fast Analysis, using a Lattice Cell model, of Opportunistic Networks*) that is computationally very efficient and precise. The model is based on discretising space and time in order to reduce the computation complexity, and we formalised it as a discrete dynamic system that can be quickly solved.

We describe some validation experiments showing that the precision of the obtained results is equivalent to the ones obtained with standard simulation approaches. The experiments also show that computation time is reduced by two orders of magnitude (from hours to seconds), allowing for a faster evaluation of opportunistic networks. Finally, we show that the FALCON model is easy to adapt and expand to consider different scenarios and protocols.

*Keywords:* Opportunistic networks, Performance Evaluation, Simulation

---

## 1. Introduction

Communications in mobile opportunistic networks take place upon the establishment of ephemeral contacts among mobile nodes using direct communication (e.g., Bluetooth or WiFi Direct). Since these mobile devices can communicate only when the people carrying them come into contact, such  
5 networks are tightly coupled with the behaviour of human social networks.

Performance evaluation of these opportunistic networks is a challenging problem, as exposed in [1, 2]. The most common approach used is based on combining a network simulation tool with realistic mobility traces. Nevertheless, simulation can be very time consuming, and the results obtained are restricted to the limited scenarios of the mobility traces used. This complexity  
10 depends mainly on the number of nodes; experiments have shown that, for example, simulations of networks with about 500 nodes can take months to finish [3]. Analytical models can avoid some of these drawbacks by providing a fast and broader performance evaluation, but they are only applicable to very simple (sometimes unrealistic) scenarios. Finally, most realistic evaluations are based on measurements performed using real scenarios and equipment, that can be very expensive

---

\*Corresponding author.

*Email addresses:* [ehernandez@disca.upv.es](mailto:ehernandez@disca.upv.es) (Enrique Hernández-Orallo), [jucano@disca.upv.es](mailto:jucano@disca.upv.es) (Juan Carlos Cano), [calafate@disca.upv.es](mailto:calafate@disca.upv.es) (Carlos T. Calafate), [pmanzoni@disca.upv.es](mailto:pmanzoni@disca.upv.es) (Pietro Manzoni)

15 and sometimes impossible to perform. In conclusion, the lack of fast and precise evaluation tools has delayed the deployment of opportunistic services, and simulation is basically the only reasonable evaluation method despite its computational complexity.

In this paper, we propose a new approach for evaluating the performance of opportunistic networks that is based on the combination of analytical and simulation models. The primary objective is to evaluate a generic opportunistic network scenario where a set of items need to be distributed among a set of nodes that move inside a bounded area. For example, the diffusion of messages among the visitors of a mall, or the spreading of local information among vehicles in a city.

Our proposal, called FALCON (*Fast Analysis, using a Lattice Cell model, of Opportunistic Networks*), is based on a discrete dynamic system considering the discretisation of space (cells) and time (period), which aims to reduce the computational complexity of simulations<sup>1</sup>. Although time discretisation is already used in current opportunistic simulators such as *The ONE* [4] and *OMNET++* [5], the primary factor that impacts their high computation times is related to its spatial calculations (concretely obtaining the positions and the distance between nodes). To solve this problem, FALCON is based on discretising the scenario in a lattice or grid of cells. Node positions and local information items are projected into this grid. All nodes in the same cell can exchange their items and pick up new ones. The position and state of the nodes are periodically updated based on a set of state functions. Finally, after iterating over time, we can obtain a set of metrics such as item diffusion dynamics and the ratio of diffusion. This basic model can be extended and adapted to consider different aspects of opportunistic networks. As an example, we expand it to include the setup and transmission time, that is a key aspect of the diffusion of messages. The precision and efficiency of FALCON depend on the correct selection of the time period and the cell size, whose values mainly depend on the communication range of the technology used.

In order to validate the FALCON model, we performed multiple experiments, using both synthetic movement models and real traces. We compare its results with those obtained through simulation. We show that the obtained results are mostly equivalent to the simulated ones, but with a very reduced computation time. To be more specific, computation time with simulations increases exponentially with the number of nodes, while with FALCON it increases linearly. For example, in a scenario with 1000 nodes, the simulation takes three hours to complete, while with FALCON it takes only 49 seconds. This supposes a huge speed-up improvement, allowing us to evaluate different scenarios faster and accurately.

We also present the evaluation of a meaningful use case where we determine the impact of the transmission time including the set-up time, an aspect that most simulators do not take into account. The results show the setup time has a significant impact on the diffusion of messages, and thus, evaluations obtained with current simulators should be reconsidered.

The paper is organised as follows. After some discussion about current evaluation methodologies and reviewing related work in section 2, we introduce the FALCON approach and its formalisation in section 3. The validation of the FALCON model is detailed in section 4. In section 5 we describe a sample case which considers and evaluates the impact that different communication issues have on opportunistic networks. Finally, in section 6 we conclude the paper.

---

<sup>1</sup>Following physics definition, we consider a lattice model as a model that is defined on a lattice, as opposed to the continuum of space or spacetime

## 2. Discussion and Related work

Opportunistic networks are mainly characterised by the ephemeral and variable duration of contacts between pairs of mobile devices. They can be considered as a special case of Mobile Ad-Hoc Networks (MANETs) [6] where the mobile devices are usually carried by humans and can communicate directly when they are in range using some wireless technology (that is, when they have an opportunity of contact). They can also be considered as a kind of Vehicular Ad-hoc Networks (VANETs) [7], as vehicles are steered by humans.

Evaluating the performance of these opportunistic networks is a challenging problem. Three main methods can be considered. *Analytical models* usually require strong assumptions and simplifications about the system to evaluate, typically considering the diffusion of a single message, so that they can produce some unrealistic results. Despite that, analytical models can provide a fast and generic performance evaluation, where we can identify the key mechanisms underlying the information diffusion. Two classes of analytical models have been proposed for modelling such network dynamics: deterministic models based on Ordinary Differential Equations (ODEs) [8, 9, 10, 11], and Markovian models [12, 8, 13, 14, 15, 16]. *Simulation* can provide more realistic results depending on the complexity of the network and the mobility model. A common approach is to combine a network simulation tool, with realistic mobility traces, such as the ones in the CRAWDAD repository [17]. This approach can reproduce the real dynamics and interactions of mobile nodes. Finally, *experiments* performed using real scenarios and equipment can be very expensive, and sometimes even impossible to perform. Nevertheless, some complete evaluations in controlled places have been performed in [18, 19, 20].

One of the most significant aspects of selecting the evaluating method is its precision and cost. Analytical methods are very fast, but the results can be unrealistic. On the contrary, simulation can obtain a precision very close to real testbeds using sophisticated simulation models, despite its computational cost. Concretely, when the workload or the scenarios used are complex, simulation can last weeks or even months, only for obtaining the results of one simple scenario.

The ONE (Opportunistic Network Environment) [4] is a simulation tool specially designed for opportunistic networks, which has a solid user community. This simulator is based on periodically updating the movement of the nodes for detecting contacts, and implements a wide variety of diffusion protocols. OMNeT++ [5] is a generic and extensible network simulator, that, combined with the INET Framework can be used for evaluating opportunistic networks[21, 22]. There are other options such as Adyton [23], which is a new alternative to The ONE for simulating opportunistic networks, and finally, the more generic ns-3 simulator combined with some specific frameworks[24].

Figure 1 shows the computation time of simulating the same scenario using four different simulators [3]. We can clearly see that the computation cost increases exponentially with the number of nodes. Performing the whole simulation (24 days of trace and 500 active nodes) can last about 3600 hours (5 months). The conclusion is clear: current simulation tools are not scalable in handling the dimensions that opportunistic networks require (thousands of nodes). In all these simulators, obtaining the contacts and its duration is based on calculating the distance between nodes. In each iteration, the distance is evaluated for each pair of nodes. As we will see, this is the primary factor affecting simulation complexity. Thus, a way to reduce this complexity must be an alternative approach for obtaining the contacts between nodes. This is the idea behind FALCON.

The FALCON model is based on some of the concepts introduced in [26]. In this paper, the authors introduced a data dissemination model based on grids. The data is located in a grid, and items can be picked by nodes and exchanged when two nodes are in range. The model keeps a

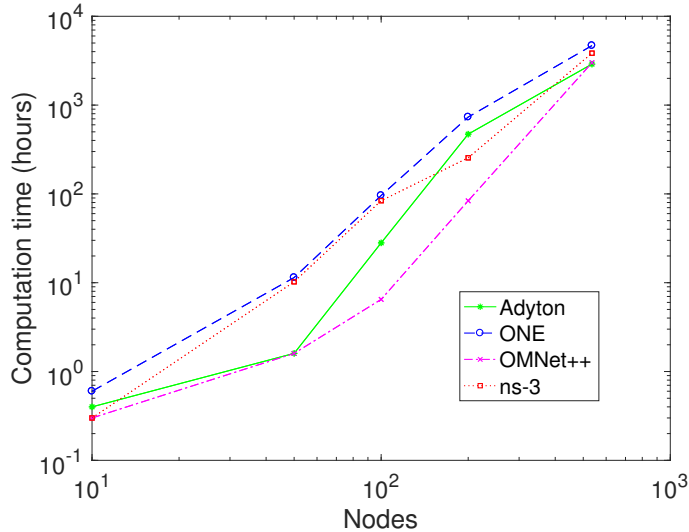


Figure 1: Computation time (in hours) for simulating the same scenario in four different simulators. The scenario used is known as the SFO Trace [25]. This dataset contains GPS coordinates of approximately 500 taxis collected over 24 days in the San Francisco Bay Area. The simulation consisted of the diffusion of periodical messages between all the nodes, varying the number of active nodes from 50 to 500. All the simulations were performed in virtualised Linux servers with 4MB of dedicated RAM and one-core 3Ghz Intel Core i5 for each virtual machine (more details in [3]).

matrix with the state of all items that have the nodes, and it is periodically updated. Nevertheless, movement and range calculation are not discretised (remaining equivalent to typical simulators) so computation time is still high.

105 Other works use grids for reducing the complexity of calculations or for improving the diffusion of data. The authors of [27] propose an opportunistic forwarding scheme (LAOF) by exploiting the node mobility in mobile DTNs, using a rectangular-grid to partition the network field into different regions. In [28], the author uses a grid model similar to FALCON for obtaining an expression for predicting the next position of nodes in order to select the best candidate to transmit the message.  
 110 The goal is to reduce the end-to-end delay when transmitting messages with a limited lifetime.

### 3. The FALCON model

As previously discussed, the main problem with simulations is their computational cost. Analytical models can provide fast results, but due to the simplification imposed by the mathematical modelling, they do not take into account several important aspects of opportunistic networks such as spatial or temporal variations (i.e., they are usually homogenous models). We, therefore, propose  
 115 a new approach that is a combination of analytical and simulation models, considering spatial dimensions and reducing simulation complexity. Particularly, FALCON is based on the discretisation of space (Cells) and the discretisation of time (Period).

Therefore, this section is devoted to explaining and formalising the FALCON model. Firstly,  
 120 in order to explain this new approach, we introduce a simple scenario. Secondly, we formalise a base model, describing and detailing its main elements and equations. The goal of this model is to evaluate the diffusion of a set of items by a set of mobile nodes. This model can be easily

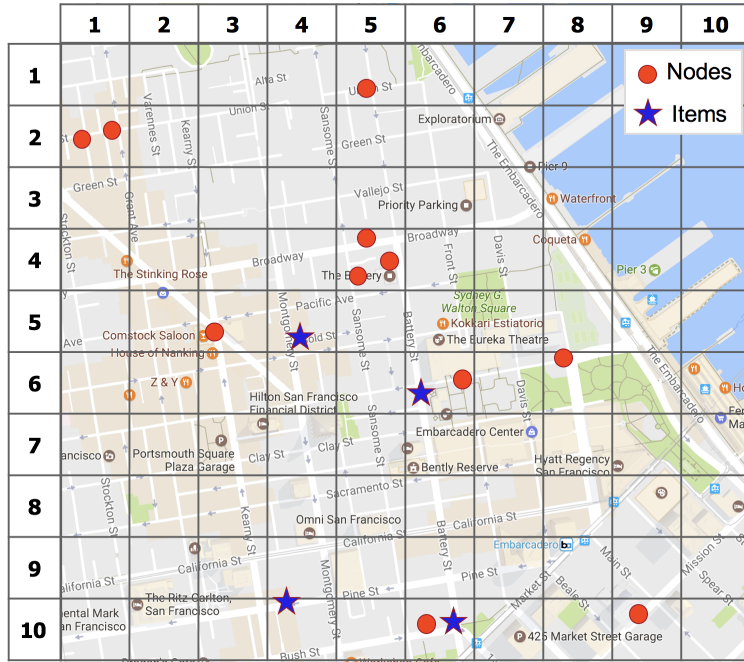


Figure 2: Sample scenario of the city of San Francisco with a grid. Each cell corresponds to approximately 100x100m. There are 8 nodes and 4 items.

modified and expanded to fit the specific scenarios of our evaluation. Finally, as an example of this adaptation, we refine this base model to include the transmission time, which is a key factor on the performance on opportunistic networks.

### 3.1. Overview of the FALCON model

In order to explain this new approach, we first introduce a simple scenario that will be formalised in the following subsection. We consider the diffusion of messages in a sector of the city of San Francisco (see figure 2). Nodes in this scenario (for example vehicles) can gather different information items from the city, and then distribute this information among other nodes using opportunistic contacts (that is, exchanging information when nodes are within communication range). Given this system, we can evaluate *how fast* is performed the diffusion of information and *when* the nodes receive this information (coverage).

Our approach is based on space and time discretisation. In the considered example, the scenario is discretised (*space discretisation*) in a grid of  $R \times C$  cells where  $R = C = 10$ , and each square cell has a side  $S$  of about 100m. Regarding the information,  $Q$  messages or data items are located in the grid cells. The scenario has  $N$  mobile nodes, that move according to a given mobility model or trace. Nodes positions are projected to the grid, so all nodes in a cell have the same position regarding the grid. The movement of the nodes (their projection) and state is updated periodically. This is the second discretisation of our approach (*time discretisation*). Nodes can store in their buffer the items collected or exchanged. In each period, the diffusion of information is performed as follows: when a node is in a cell containing an item, it picks it and stores it in its buffer. When two or more nodes are in the same cell, they exchange all their stored items, so that, when this period ends, all nodes in the cell share all the items.

145 For example, figure 2 represents the position of nodes and items at a given period. The nodes in cells (6,6) and (10,6) pick and store the item, while nodes in cells (2,1) and (4,5) exchange their items. On the next period, nodes can move to another cell, picking or exchanging more items. When the system progresses in time, we can obtain different performance parameters such as the diffusion time or the information coverage.

150 It is clear that the discretisation used in this model can affect the precision of the obtained results. Time discretisation is already used in current simulators. For example, the ONE simulator is configured by default with a time period of 1s. That is, at every period, the ONE simulator evaluates the status of the simulation, for example by determining which nodes are in contact, or if a transmission has finished. Increasing this time period to e.g. 10s, can reduce the number of contacts, as some contacts are not observable, thereby reducing the overall performance of the information diffusion, and thus the precision of the obtained results. Regarding space discretisation, its impact is determined by two aspects: the communication range and nodes mobility. In current simulators, the communication range is used to determine if two nodes are in contact in order to start a transmission. With cell discretisation, all nodes that are in the same cell are considered to be in range. This can be considered a serious limitation of the model, but as it will be shown later on, the key issue is to set the appropriate cell size. Now, we formalise this model by first considering that there is no transmission time, and then improving it to include setup and transmission times.

### 3.2. FALCON system model

This subsection is devoted to formalise the FALCON model as a *discrete dynamic system* using difference equations. We start by introducing a base model that evaluates the diffusion of a set of items by a set of mobile nodes, as described in the previous subsection.

We consider a system with  $N$  nodes, in a discretised area of  $R \times C$  cells with  $Q$  information items and cell size  $S$ . Using systems model notation, the system state at period  $k$  denoted by  $\mathbf{x}_k$ , is a function of the previous state and the inputs of the system  $\mathbf{u}_k$ :

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_k) \quad (1)$$

In FALCON, we consider that the state comprises the items that have all the nodes in the system. The input of the system consists of the following elements: data item positions  $\mathbf{d}_k$ , position of the nodes  $\mathbf{n}_k$  and communication grid  $\mathbf{c}_k$ , so the state function is:

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{d}_k, \mathbf{n}_k, \mathbf{c}_k) \quad (2)$$

The evolution of this system from a given initial state  $\mathbf{x}_0$ , depending on the set inputs, is known as the *state-space* of the system:  $\{\mathbf{x}_k\}$ . Now, we describe the different variables of the previous equation:

- 170 • Data items vector  $\mathbf{d}_k$ : is a vector of grid coordinates with the location of the  $Q$  items at iteration  $k$ . That is, element  $\mathbf{d}_k(i)$  contains the row and column  $(y, x)$  of the  $i^{th}$  item. Note that, in some systems, the data items can be constant (do not vary along time) and so the subindex can be dropped.
- Communication grid  $\mathbf{c}_k$ : is a matrix of  $R \times C$  elements that determines the probability of communication on the cell. It can be binary (0: no communication, 1: communication) or a probability (from 0 to 1: full communication). This allows to model areas with better or worse communication; for example, open spaces can be set to 1, and areas with interferences or obstacles can be set to lower values.

- Nodes position vector  $\mathbf{n}_k$ : is a vector of coordinates with the current position (row and column) of the nodes at iteration  $k$ .
- Data state  $\mathbf{x}_k$ : is a matrix of  $N \times Q$  elements where each row  $n$  corresponds to the state of the  $n^{\text{th}}$  node. Each column  $i$  indicates if the  $i^{\text{th}}$  item is stored in the node. That is, if  $\mathbf{x}_k(n, i) = 1$ , then node  $n$  has item  $i$ .

In order to evaluate the state, the state function 2 can be simply defined using the logical OR:

$$\mathbf{x}_k = \mathbf{x}_{k-1} \oplus f^d(\mathbf{d}_k, \mathbf{n}_k) \oplus f^c(\mathbf{x}_{k-1}, \mathbf{n}_k, \mathbf{c}_k) \quad (3)$$

introducing two functions:

- Direct function:  $\mathbf{x}_k^d = f^d(\mathbf{d}_k, \mathbf{n}_k)$ , which are the items collected directly by nodes when they are in a cell with items. This function returns a state variable with the items collected, and it only depends on the node and item positions; it can be obtained as:

$$\mathbf{x}_k^d(n, i) = 1 \quad \text{if } \mathbf{d}_k(i) = \mathbf{n}_k(n) \quad (4)$$

- Communication function:  $\mathbf{x}_k^c = f^c(\mathbf{x}_{k-1}, \mathbf{n}_k, \mathbf{c}_k)$ , that are the items exchanged by the nodes when they are in the same cell with the probability given by  $\mathbf{c}_k$ . We can derive an expression considering the case of a binary probability communication grid:

$$\mathbf{x}_k^c(n, i) = 1 \quad \text{if } \exists m \text{ where } \begin{cases} \mathbf{x}_{k-1}(m, i) = 1 & \wedge \\ \mathbf{n}_k(n) = \mathbf{n}_k(m) & \wedge \\ \mathbf{c}_k(\mathbf{n}_k(n)) = 1 \end{cases} \quad (5)$$

Finally, we can obtain the evolution of the system iteratively by resolving expression 3 up to a given iteration  $k$  from a given initial state  $\mathbf{x}_0$  (that can be a zero matrix considering that initially, the nodes have no items). From the obtained state-space we can obtain metrics such as the diffusion of items depending on time, the ratio of diffusion, and the diffusion time. Considering the duration of a period as  $\delta$ , then the time at iteration  $k$  is  $t = \delta k$ , and so we can define the following metrics:

- Diffusion of item  $i$  at time  $t$ , that is, the number of nodes that have that item:

$$D_i(t) = \sum_{n=1}^N \mathbf{x}_k(n, i) \quad k = \lceil t/\delta \rceil \quad (6)$$

- Diffusion of all items at time  $t$ , as the sum of all items that have all the nodes:

$$D(t) = \sum_{i=1}^Q D_i(t) \quad (7)$$

- Ratio of diffusion of all the items at time  $t$ :

$$R(t) = \frac{D(t)}{N \cdot Q} \quad (8)$$



	1	2	3	4	5
1		★ 2		★ 1	④ ★ 5
2			★ 6	① ★ 4	
3					
4	②	⑤		★ 7	
5		★ 3	③		

Figure 3: Grid used as an example. The circles represent the initial position of the nodes, and the stars represent the position of the items. The grey cells represent cells with full communication.

- Diffusion time for a given ratio  $R$ , which is the time when the diffusion is greater than a given ratio:

$$t = \delta k \quad \{ \min k \quad R(\delta k) \geq R \} \quad (9)$$

190 As shown later on, the precision and efficiency of FALCON depend on the selection of the grid size and time period. The cell dimensions mainly depend on the communication range of the technology used. The time period can affect different on the number of effective contacts and the transmission effectiveness. The reason for this is that, in one period, all nodes exchange all messages and pick up all messages. So, it depends on the speed of the nodes: very fast nodes  
195 and a low period imply that some positions are not *sampled*, missing contact opportunities. These aspects will be studied in depth in section 4.

### 3.3. Example

In order to understand the different elements and components of the FALCON model and its dynamics, we include a simple example considering a system with five nodes ( $N = 5$ ), seven items  
200 ( $Q = 7$ ) and a cell grid of  $5 \times 5$  ( $R = C = 5$ ). The state of the system (nodes and items position) at iteration  $k$  is shown in figure 3.

First, the position of the items is defined by the data items vector, and do not vary over time:

$$\mathbf{d} = [(1, 4) (1, 2) (5, 1) (2, 4) (1, 5) (2, 3) (4, 4)] \quad (10)$$

The communication grid is:

$$\mathbf{c} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (11)$$

The position of nodes at iteration  $k$  are:

$$\mathbf{n}_k = [(2, 4) (4, 1) (5, 2) (1, 5) (4, 1)] \quad (12)$$

and finally, the previous state, for instance, can be the following:

$$\mathbf{x}_{k-1} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \quad (13)$$

where, for example, node 1 has items  $\{1, 4, 6\}$ , and node 3 has no items. Using the previous variables, we can obtain the state from the direct function as:

$$\mathbf{x}_k^d = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (14)$$

and also the communication function as:

$$\mathbf{x}_k^c = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \quad (15)$$

where it can be seen that nodes 2 and 5 exchange their items. Finally, adding all the terms (logical OR) using expression 3, yields the updated state:

$$\mathbf{x}_k = \mathbf{x}_{k-1} \oplus \mathbf{x}_k^d \oplus \mathbf{x}_k^c = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & \mathbf{1} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & \mathbf{1} & 1 & 0 \\ 0 & \mathbf{1} & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \quad (16)$$

Note that the changes from the previous state are marked with bold font.

### 3.4. Considering transmission time

The previous model assumes that nodes that are in the same cell can transmit all messages in one iteration. This is not a realistic assumption as transmission time depends on setup time and messages size. Thus, we expanded the previous FALCON model in order to take into consideration the duration of the contacts and the transmission times. Specifically, the data state matrix and the nodes position vector must include the following additional information:

- Nodes position vector  $\mathbf{n}_k$ : it is necessary to include the remaining time a node will stay in the cell that will be used to calculate the duration of a contact. Thus, each element of  $\mathbf{n}_k$  contains the row, column and remaining time of the nodes at iteration  $k$ .

- Data state  $\mathbf{x}_k$ : a last column is added to indicate whether a node is transmitting or not. That is, if  $\mathbf{x}_k(n, Q + 1) = 1$ , then node  $n$  is transmitting.
- Setup time  $\tau$ : the time needed since a contact starts until the nodes start transmitting the items
- Item time  $\mathbf{t}$ : the time required to transmit each item is defined as a vector  $\mathbf{t}$  of  $Q$  elements, where  $\mathbf{t}(i)$  corresponds to the transmission time of the  $i^{\text{th}}$  item.

Note that the item time depends on the size of each item, and the speed (bandwidth) of the link. For example, an item of  $1MB$  (for example a photo) transmitted using Bluetooth technology with an effective bandwidth of  $2.1Mb/s$ , will last  $(1 \times 8)/2.1 = 3.8s$ . Considering both the setup time and transmission time of the message, when two nodes  $n$  and  $m$  start a transmission, the required communication time to exchange their messages will be:

$$T_c = \tau + \sum_{\forall i \mathbf{x}(n,i) \neq \mathbf{x}(m,i)} \mathbf{t}(i) \quad (17)$$

To consider the transmission time, the communication function  $f^c$  is modified in the following way:

$$\mathbf{x}_k^c = f^c(\mathbf{x}_{k-1}, \mathbf{n}_k, \mathbf{c}_k, \tau, \mathbf{t}) \quad (18)$$

Now, we briefly describe the implementation of this function. When a contact starts, we must consider the duration of the contact and the transmission time. From the state, we can first determine if both nodes can transmit. If one of the nodes is already transmitting, no items can be exchanged in this contact opportunity (it must wait to the transmission of one of the nodes to end or try to exchange an item with other available nodes). When both nodes determine that they can transmit, we must take into account the duration of the contacts. In the FALCON model, a contact duration  $T_d$  can be obtained as the time during which two nodes are in the same cell. Using vector  $\mathbf{n}_k$ , if two nodes  $n$  and  $m$  at iteration  $k$  are in the same cell, the duration of a contact can be obtained as  $T_d = \min(\mathbf{n}(n, 3)_k, \mathbf{n}(m, 3)_k)$ . Thus, if the duration of a contact lasts less than the required communication time, we must transmit only the messages up to the contact duration<sup>2</sup>. Thus, the idea is to calculate the remaining time  $T_r$  of the contact.

When a contact starts, the remaining time is set to the contact duration  $T_r = T_d$ . First, we consider the setup time, so  $T_r = T_r - \tau$ . Then, while  $T_r$  is greater than zero, we can exchange an item  $i$  if  $\mathbf{t}(i) \leq T_r$ , updating then the remaining time. Thus, the exchange will end when all items are exchanged, or when the remaining time is consumed. It is important to note that the model cannot be aware of the contact duration. So, it can start a transmission and, if the setup time plus the transmission time of the first item is greater than the contact duration, no items will be exchanged (that is, it can be considered as a wasted contact). Finally, the order in which the items are exchanged is based on their number (that is, messages with lower numbers are exchanged first); but other models can be used in order to improve efficiency (for example, items with lower sizes first, as described in [29]).

Summing up, the FALCON model can be easily adapted for evaluating different systems. The previous models represent the most typical diffusion scenarios, but other scenarios or communication protocols can be easily considered. For example, instead of nodes picking items up, these

<sup>2</sup>Note that not all contact durations will allow completing a full exchange of items since they must be lower than the transmission time given by equation 17.

items can be generated directly by the nodes. In this case, we can modify the direct function considering, for example, that items are generated randomly between all nodes at a given rate, or even considering more complex patterns. Thus, in this extension, the data items vector is not used. We can also consider the evaluation of node-to-node protocols, such as P<sub>Ro</sub>PHET [30], and Spray and wait [31]. In this case, we must include in the data items vector, a destination, so that it can be used in the communication function to determine, when a contact occurs, how a message is transmitted according to the protocol.

#### 4. Validation of the FALCON model

This section is devoted to the validation of the FALCON model. First, we provide some details about its implementation; then, we evaluate its precision comparing the results of FALCON with analytical and simulation results, using both synthetic and real mobility traces. We also evaluate the impact of the period and the cell dimension on the results. Finally, we evaluate the computation cost compared with simulation. All validations performed in this section are referred to the base FALCON model.

We implemented the FALCON model in Matlab<sup>3</sup>. Actually, the implementation was straightforward from the definitions and the equations described in subsection 3.2. Nevertheless, some practical optimisations have been done to speed-up the calculations. The most important one is the evaluation of expression 3. Instead of generating the  $\mathbf{x}_k^d$  and  $\mathbf{x}_k^c$  vectors, and then performing the logical OR, the direct and communication functions directly modify the status matrix  $\mathbf{x}_k$ . The data items vector  $d$  is also converted to a  $R \times C$  matrix where each element contains the item in that cell. Finally, only those nodes that moved to a different cell from the previous iteration are evaluated.

Node movements are previously generated using an auxiliary custom function that can generate the vector  $n_k$  from real position traces or from synthetic mobility models, discretised to time periods and cell positions. Once this is defined, the system is evaluated iteratively up to the desired time. The final result is the whole state space  $\{\mathbf{x}_k\}$ . Our implementation of FALCON also provides a helpful graphical visualisation tool, where the status of the system is iteratively updated. For example, figure 4 shows a snapshot of the system, showing the position of the nodes and the items received at a given iteration. Finally, we can obtain several metrics, such as the ratio of diffusion ( $R_k$  using expression 8), the diffusion of an item ( $D_i(t)$  using expression 6), and the number of items and contacts depending on time, as shown in figure 5. The precision of the results will depend on the correct selection of the time period  $\delta$  and the cell size, as we will be detailed in the following subsections.

##### 4.1. Cell dimensions and range

The communication range is usually expressed as an average distance where two nodes can communicate. For example, for Bluetooth we can use 5-15 meters, and for Wi-Fi 50-200 meters. Nevertheless, the FALCON model considers that all nodes in the same cell are into range. This imposes two important limitations: firstly, two nodes that are very close but in different cells do not

---

<sup>3</sup>All implemented models and graphical tools are freely available for the research community on the GitHub site of our research group (<https://github.com/GRCDEV/falcon/>). This implementation, which is provided as a *proof of concept*, can be easily re-coded in other languages.

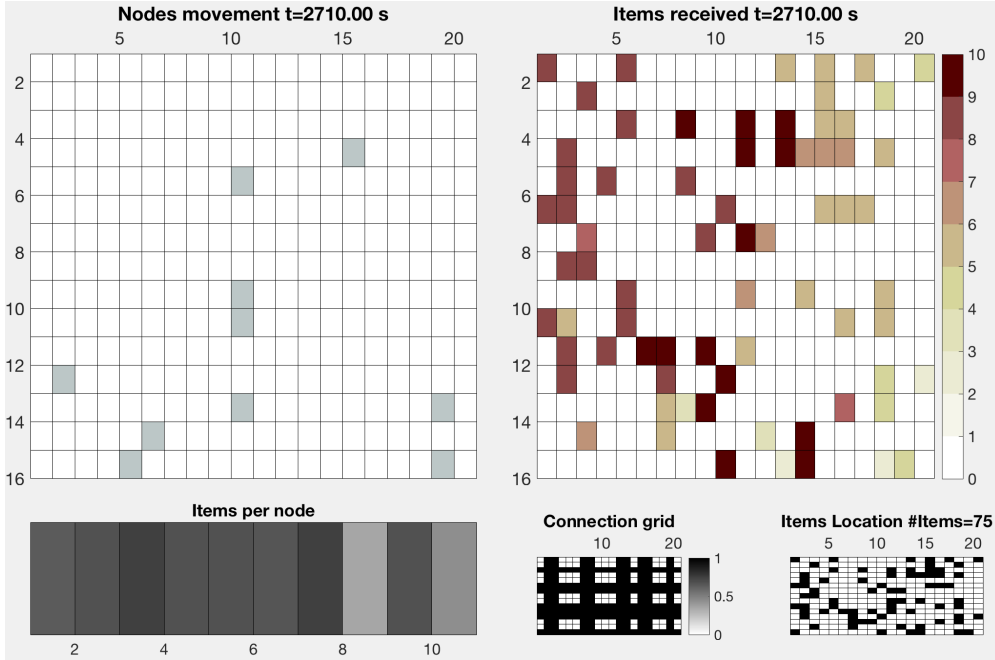


Figure 4: Snapshot of the evolution of an example system in the Matlab implementation. The grid has 15 rows and 20 columns with a cell size of 10 meters. The number of nodes is 10, and the number of items is 75, which were randomly positioned into the grid. Note that, the graphical model is optional and this evaluation takes less than one second in batch mode.

280 communicate, and secondly, the range is homogeneous (not affected by interferences or obstacles<sup>4</sup>). This is different from simulation models where two nodes are considered to be in range based on their Euclidean distance<sup>5</sup>. That is, opportunistic networks simulators determine that a contact starts when the distance between two nodes is less than or equal to a fixed range, and it is finished when the distance is greater. In FALCON, a contact starts when two nodes meet in the same cell,  
 285 and finish when at least one of them leaves the cell.

Thus, we must establish a relationship between a distance-based range and the cell side as  $S = F_s \cdot range$ , where  $F_s$  would be a scale factor. This factor was obtained empirically based on the following fact: for a given range and cell side  $S$ , and considering a set of mobile nodes, the number and duration of the contacts must be the same in both models (distance-based and cell-based).  
 290 The experiment basically consisted of finding the factor that minimises the difference between the number of contacts of both models. That is, given an initial factor  $F_s$ , a set of nodes and an area, the process is the following:

1. Generate the mobility for the set of nodes and the evaluated area,
2. get the contacts for a distance-based model using a given range,
- 295 3. get the contacts for the FALCON model using cell side  $S = F_s \cdot range$

<sup>4</sup>Although it can be modified by the use of the Communication grid  $c_k$

<sup>5</sup>This is the model used in opportunistic networks simulations. Other network simulators can use more complex models considering radio propagation and obstacles (i.e Nakagami), making the simulation even more computationally intensive.

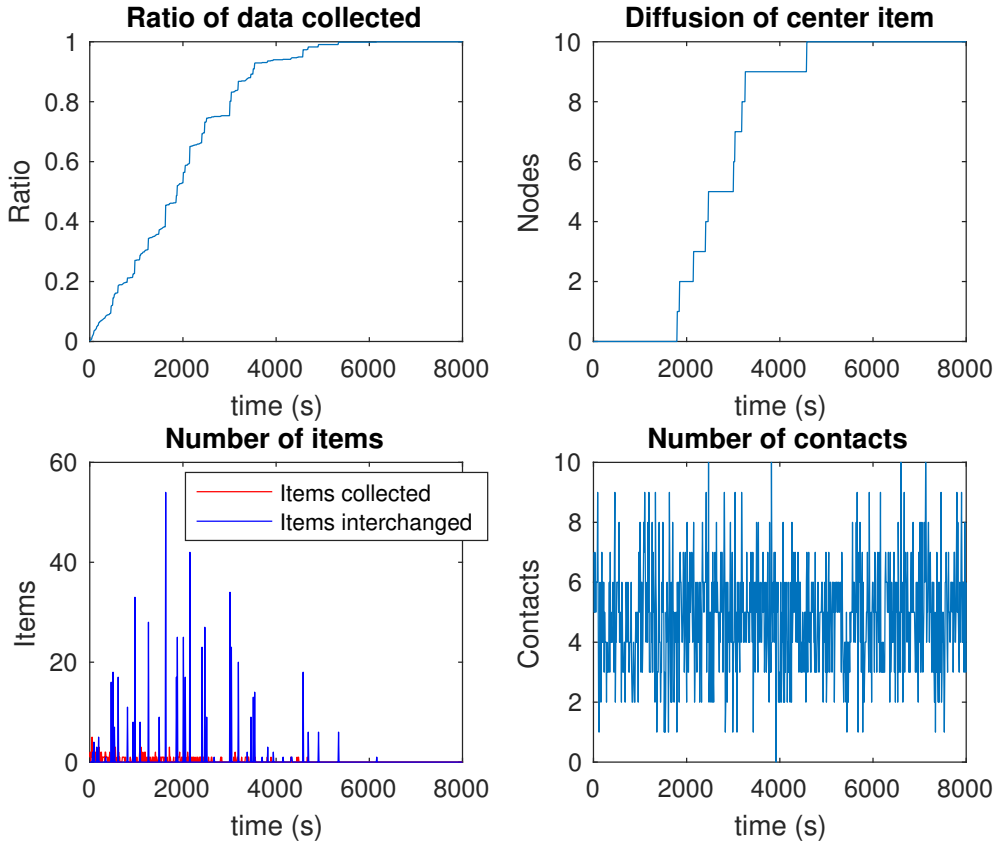


Figure 5: This figure shows some of the plots that can be obtained from the FALCON model implemented in Matlab, corresponding to the model shown in figure 4. We can see the ratio of diffusion of all the items, and the diffusion of one node. We can also see the number of items exchanged and collected depending on time, and the number of contacts generated by the model.

Using the bisection method, this process was repeated by varying the factor  $F_s$  in the range  $[1 \dots 2]$  until the difference of contacts between both models was below a given threshold error. Finally, we used several grid dimensions, number of nodes, different ranges and mobility models (even real traffic traces). The average factor was 1.395 with a 90% confidence interval of  $[1.35 \dots 1.43]$ . Thus, we used  $F_s = 1.4$  along the paper. This factor was also confirmed by the accuracy of the results obtained in the following subsections (using another factor produced less accurate results).

#### 4.2. Comparison using synthetic mobility models

In this subsection, we compare FALCON results with other performance evaluation models, such as analytical models and simulation using synthetic mobility models. Four different experiments using two different scenarios were evaluated, with the aim to validate the precision of the FALCON model.

The comparison process is outlined in figure 6. First, a mobility trace is generated from a synthetic mobility model. This trace can be generated using the *BonnMotion* tool [32], which is a tool that can create different mobility scenarios ranging from simple synthetic models such as RWM (Random Waypoint Model) to more realistic models such as SWIM (Small Worlds In

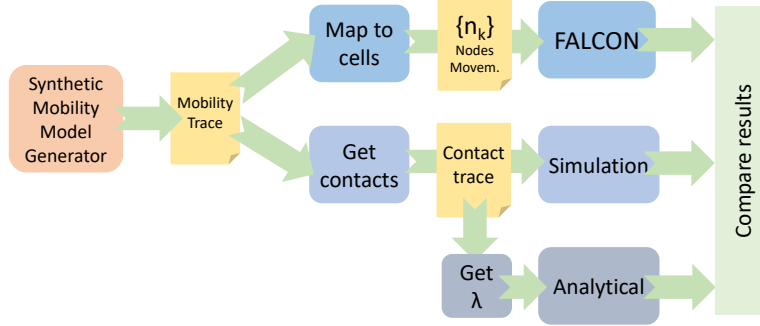


Figure 6: The process for comparing the results. This process was used in the comparison of evaluation method using synthetic mobility models but is also similar to the process used when real mobility traces were used.

Motion). This mobility trace must be mapped onto the set of nodes position vectors  $\{\mathbf{n}_k\}$ , that is, the cell location of the nodes at each period interval  $k$ , used in the FALCON evaluation. Using the same mobility trace, a contact trace is also generated considering that a contact occurs when two nodes are in range, also evaluating this condition using the same period. For the simulation, we used a simple custom contact-based simulator, that is fed with this contact trace<sup>6</sup>. Finally, from this contact trace, we can obtain the average contact rate  $\lambda$  required in the analytical model. Additionally, due to the random nature of the mobility generators, we repeated the experiments 100 times in order to obtain average results and 90% confidence intervals.

The first scenario is an area of about  $1Km \times 1Km$  with 100 nodes, with a communication range of 10m (i.e. Bluetooth range). For this range, the cell side is  $S = F_s \cdot 1.4 = 14m$ , with a grid of  $75 \times 75$  cells. The duration of the period  $\delta$  was set to 1s, and the simulation time  $SimTime = 25.000s$  (about 7 hours) in order to get a full diffusion of the message. Regarding the mobility model, we used the basic Random Waypoint Model (RWM) adjusted to resemble pedestrian mobility with the parameters detailed in table 1. Due to the limitations of the existing analytical models, in this first experiment we limit our comparison to the diffusion of only one message in a network of nodes, so the data items vector is empty, and the initial state is set to a vector where one randomly node has an item. That is, a single message epidemic diffusion based on nodes contacts. Thus, from [9] we can obtain the diffusion of an item (infected nodes) depending on time as:

$$I(t) = \frac{N}{1 + (N - 1)e^{-\lambda N t}} \quad (19)$$

where  $\lambda$  is the contact rate between pairs of nodes that was obtained from the mobility trace using the following expression:

$$\lambda = \frac{C_t}{SimTime \cdot 0.5 \cdot N(N - 1)} \quad (20)$$

where  $C_t$  is the number of contacts obtained from the mobility trace, as detailed in [33].

<sup>6</sup>Actually, the simulation experiments of this section were performed using the ONE simulator. Nevertheless, in the section that follows, since the ONE simulator has no setup time, we implemented a custom simulator in Matlab to validate these results. Note that both simulators are contact-based, that is, when nodes are within radio they start a transmission. The performance results of this custom simulator and the ONE simulator were equivalent when the setup time was set to zero.

Table 1: Main parameters for the synthetic scenarios experiments.

Parameters	Experiment 1	Experiment 2	Experiment 3	Experiment 4
Nodes	100		50	
Items	1	10	1	20
Range	10m (Bluetooth)		100m (Wi-Fi)	
Area	1Km × 1Km		7Km × 7Km	
Grid	75 × 75 cell, side 14m		50 × 50 cell, side 140m	
Sim. time	25.000s	30.000s	100.000s	100.000s
Mobility model	RWM Speed interval [0.2 . . . 2.2] (m/s) Pause interval [0 . . . 10]s Walk interval [2.00 . . . 20.00]s		SWIM NodeRadius 0.005 CellDistanceWeight 0.5 WaitingTimeExponent 1.45 WaitingTimeUpperBound 30000	

320 The results are shown in figure 7a, where the ratio of diffusion  $R(t) = D(t)/N$  is shown depending on time. We can see that FALCON and simulation results are very close, but the results of the analytical model are too optimistic. With regard to the analytical model results, it confirms the experiments conducted by [34, 33] that showed that a single homogeneous contact rate is unable to capture the global performance of a mobility trace.

325 In the second experiment, we evaluated the precision of FALCON using a more complex scenario where nodes can pick items and then exchange them in order to spread information. Since no analytical model exists capable to model this scenario, we compare our results only with simulation, following again the process described in figure 6. The area, grid, cell size, mobility model and period are the same than in the previous experiment (all details in column *experiment 2* of table 1). The number of items was set to 10 ( $Q = 10$ ), all of them randomly placed in the area. Finally, the simulation time was set to 30.000s (about 8 hours) in order to ensure that all items are completely diffused. The results are shown in figure 7b where the ratio of diffusion of all items (that is, using expression 8) is plotted. The results show a very good approximation of the FALCON curve with respect to simulation results.

335 The third and fourth experiments used a more realistic synthetic mobility model generator, namely the SWIM (Small Worlds In Motion) model [35], which is based on the assumption that users either select a location close to their home or a very popular location. We also changed the communication technology to a Wi-Fi based one, with an average range of 100m. Consequently, the scenario is greater: a grid of  $50 \times 50$  cells with a side of 140m. The duration of the period  $\delta$  was set to 1s, and the simulation time to 100.000s (about 27 hours) in order to get a full diffusion of the message and picked items. Finally, the number of nodes was 50. All the parameters of these experiments are detailed in table 1, including the SWIM model ones. The results are shown in figures 7c and 7d, where we can see, similarly to the previous experiments, that FALCON and simulation results are very close, confirming the validity of the proposed model.

#### 345 4.3. Comparison using real mobility traces

In the experiments of the previous subsection, we used synthetic mobility models for generating the movement of the nodes. This allowed us to compare the performance of the different models depending on different factors (mainly nodes, items and grid dimensions). Nevertheless, if we want



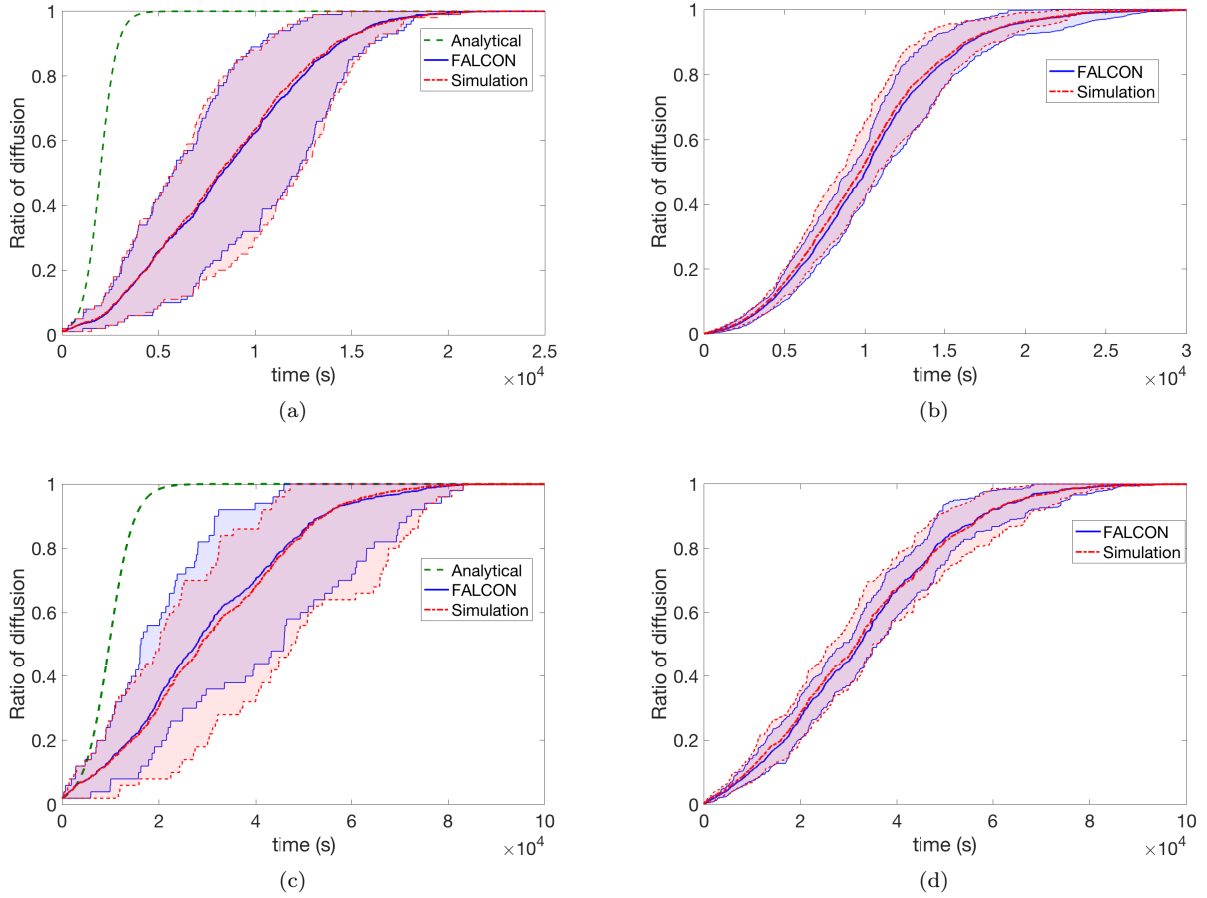


Figure 7: Comparison of the different evaluation methods using synthetic models. (a) Experiment 1: Ratio of diffusion of one message using a RWM mobility model and short-range communication (Bluetooth); (b) Experiment 2: Ratio of diffusion of 10 items for same scenario of Experiment 1. (c) Experiment 3: Ratio of diffusion of one message using SWIM and medium-range communication (Wi-Fi); (d) Experiment 4: Ratio of diffusion of 20 items for the same scenario of Experiment 3. Note that all the results for FALCON and Simulation show the area corresponding to 90% confidence intervals.

to perform a more realistic evaluation of opportunistic networks we must use real traffic traces. Thus, in this subsection, two different traces are used, both from human and vehicles mobility patterns.

In the first experiment, we used a trace of mobile users, from an experiment at the NCCU campus [36], where GPS position data was collected during two weeks (336 h) using an Android app installed in the smartphones of 115 students in an area of  $3.76Km \times 3.42Km$ . For our experiment, we selected the first 24h of this trace for evaluating the diffusion of one message using short-range communication (Bluetooth). Therefore, the cell size was set to 14m, so the grid has  $269 \times 244$  cells. As in the previous subsection, we compare the results obtained using the analytical model (that is, equation 19), the FALCON model, and simulation, following again the process outlined in figure 6. The results shown in figure 8 confirms the precision of the FALCON model<sup>7</sup>.

<sup>7</sup>Note that in this experiment, as the mobility trace is not generated randomly, the result of the diffusion is unique,

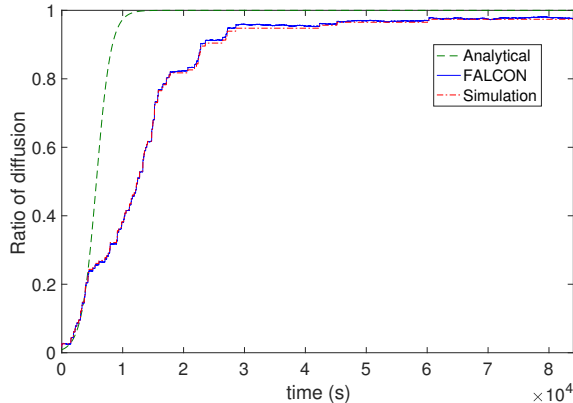


Figure 8: Ratio of diffusion of one items for Analytical model, FALCON and simulation, for the NCCU trace using short-range communication (Bluetooth)

360 For the second experiment, we used a mobility trace of taxi cabs in San Francisco, USA [37] (also referred to as SFO trace), which has a higher number of nodes. This dataset contains GPS coordinates of approximately 500 taxis collected over 24 days in May 2008 in the San Francisco Bay Area. From this trace we used only four hours of one day (May 18, 2008, from 7am to 11am), and the area was restricted to the city centre, as shown in figure 9 (between latitude range  
 365  $[37.72^\circ \dots 37.82^\circ]$  and longitude range  $[-122.47^\circ \dots -122.37^\circ]$ , an area of about  $11.13 \times 8.80$  Km)<sup>8</sup>. The location-updates are quite fine-grained, and the average time interval between two consecutive location updates is less than 10s. Nevertheless, as we required an update of 1s, we interpolated the intermediate positions. We assume that the communication technology is WiFi with an average range of 100m, and so the cell side was set to 140m. Then, the grid has  $80 \times 63$  cells.

370 We performed two different experiments. In the first one, we set twenty items regularly placed, as shown in figure 9, in locations where traffic is medium to heavy (the red points). The ratio of diffusion of these items is shown in figure 10, labelled as “Items #1”. We can see that the diffusion of these items is very fast (due to the location of the items) and that both curves are very similar. In the second experiment, we set four items in locations where the traffic is scarce or even with  
 375 no traffic at all. The diffusion is shown also in figure 10, labelled as “Items #2”. We can see that the diffusion is constrained by the items picking, and even one of the items was never collected (specifically, the second on the left). Comparing the results of both evaluation methods (FALCON vs Simulation), we can see that there is a difference in the range 0-2000s, due to that one item in the simulation which was picked earlier than in FALCON, and so its diffusion started before.  
 380 This is the effect of the cell discretisation that generates this kind of differences. In particular, the rightmost items have a very low density of cars around, and therefore, in the simulation, the first car that picked the item in the simulator, missed the item in FALCON because the item was at a surrounding cell, and not in the cell of the car. This is one the main causes that can produce differences on the results of both models.

---

so there are not confidence intervals.

<sup>8</sup>for San Francisco latitudes we used the following conversion  $1^\circ Lat \approx 111.319km$  and  $1^\circ Lon \approx 88.055km$

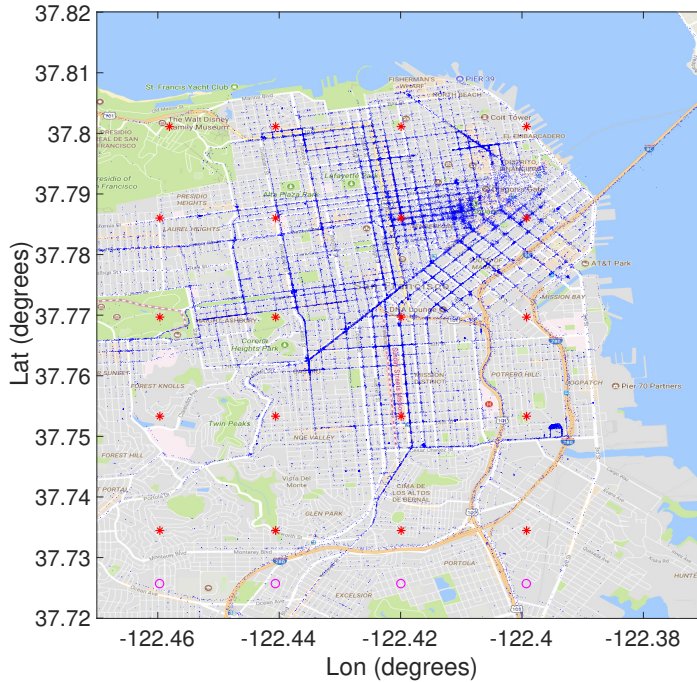


Figure 9: San Francisco trace sample of 4h of GPS coordinates from 536 taxis corresponding to May 18, 2008, from 7 am to 11 am. Although the original trace has a greater area, in the evaluation we used the area represented in the figure. The first set of items “Items #1”, that are positioned as a matrix of  $5 \times 4$ , are represented by red asterisks. The second set “Items #2” are positioned as a row at the bottom and are represented by magenta circles.

#### 385 4.4. Computation time

One of the primary motivations for introducing FALCON is reducing the cost of opportunistic networks performance evaluation while keeping a high accuracy of the results. As shown in the results of the previous experiments described in this section, the accuracy of FALCON is very close to the simulated ones. Now, we evaluate the cost of both methods.

390 Particularly, the objective is to evaluate the impact of the number of nodes and items on the computation time. The scenario is similar to previous experiments, where two different grid dimensions are evaluated:  $75 \times 75$  and  $150 \times 150$  cells with a fixed side of 14m, and node mobility is generated according to the RWM model. The simulation time was set to 10.000s (about 3 hours).  
 395 The number of nodes  $N$  varied from 50 to 1000, with step 50, and the number of items  $Q$  varied depending on the number of nodes as  $0.2N$ . For each value of  $N$ , we obtained the average execution time after repeating 10 times the calculation of the ratio of diffusion up to 10.000s.

The results are shown in figure 11 with logarithmic y-axes scale. We can clearly observe that the computation time of simulation is practically two orders of magnitude greater than the FALCON computation. In fact, it increases exponentially with the number of nodes (and items), with  
 400 a greater slope than FALCON. For example, we can compare the complexity for 1000 nodes: FALCON takes about 49 seconds, and simulation takes about 3 hours (that is, one hour of simulation represents nearly one hour of computation time). This supposes a huge speed-up, particularly when  $N$  is greater than 500 nodes, achieving speedup factors greater than 50. On the other hand,

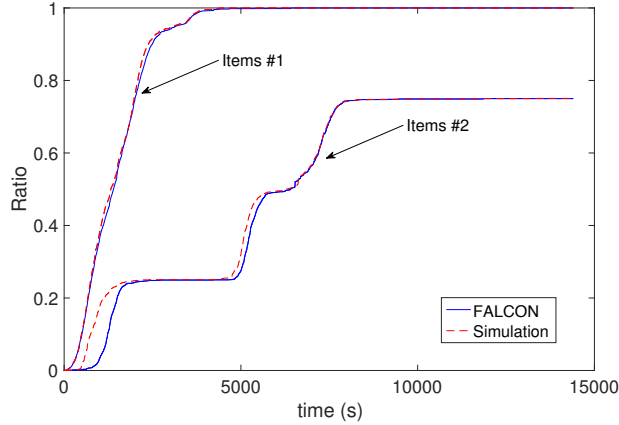


Figure 10: Ratio of diffusion of the items for FALCON and simulation for the SFO trace using medium-range communication (Wi-Fi)

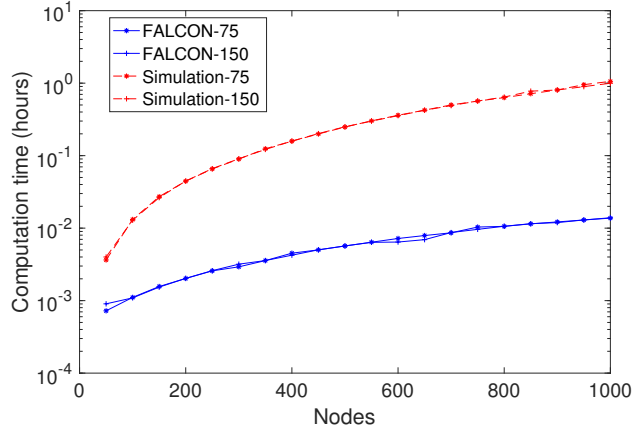


Figure 11: The average computation time of simulation and FALCON depending on the number of nodes  $N$  for two cell grid dimensions  $75 \times 75$  and  $150 \times 150$ . Note the logarithmic y-scale. All the simulation were performed using Matlab in a PC with 8MB of RAM using only one-core in a 3.4Ghz Intel i5.

405 increasing the grid dimensions does not impact on the computation time, since both curves (for 75 and 150 side cells) are very similar.

410 To justify this difference, we can analyse how FALCON and simulation are implemented for evaluating their complexity. Although we did not perform a formal evaluation of the complexity of the algorithms, we studied the main aspects that affect their computation time. In FALCON, for each iteration, only the nodes that experienced a cell change are evaluated, and determining the list of these nodes has a linear cost on  $N$ . On the contrary, for simulation, the main cost is determining when two nodes are in contact or not. This implies obtaining the distance between every pair of nodes, that have a cost of about  $O(N^2)$ , as can be clearly seen in the simulation results shown in figure 11. Thus, increasing the area or the number of cells does not have a relevant impact on the computation time, as evidenced in the results shown in figure 11.

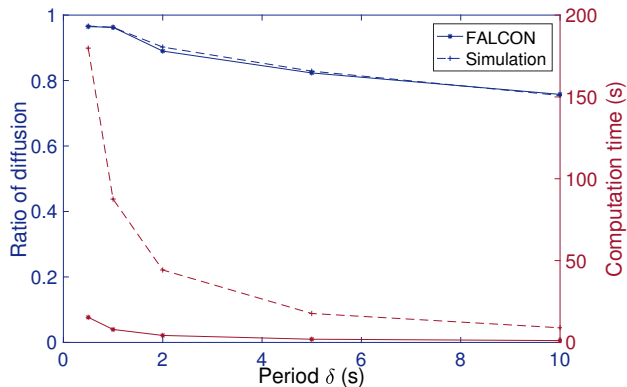


Figure 12: Impact of the period on the precision and computation time. The periods evaluated were  $\{0.5, 1, 2, 5, 10\}$ s. The two top lines represent the ratio of diffusion, and the bottom two lines represent the computation time in seconds. All the simulations were performed using only one-core in a 3.4Ghz Intel i5.

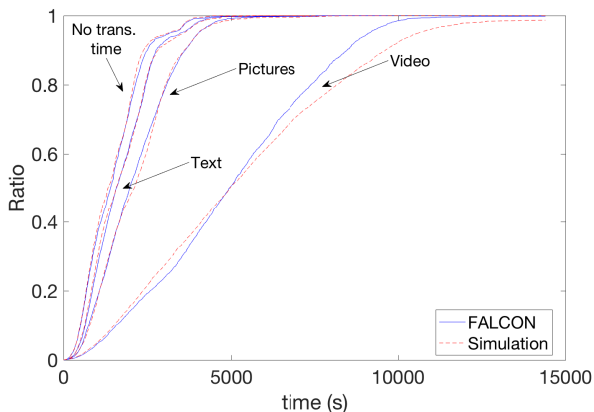


Figure 13: Ratio of diffusion for “Items #1” for FALCON and simulation considering different transmission times.

#### 415 4.5. Impact of period

Since the period is very related to the computation time, we study the impact of the period on the FALCON model. This topic has been addressed in simulators based on periods like ONE and OMNeT++. For example, in [3], it is shown that increasing the period from 1 second to 5 seconds, reduces the simulation time by a factor of 5, but the simulation results differ (specifically the rate of diffusion and the diffusion time), so the obtained results are less precise. The reason for this behaviour is clear: for greater resolutions, some of the possible contacts are missed, and so the opportunity to transmit a message is lost, reducing the overall performance. Actually, the default period of these simulators is set to one second. Reducing the period does not produce a significant improvement on result accuracy, but it increases the simulation times considerably.

425 We performed similar experiments with FALCON. Using the second scenario described in detail in subsection 4.2, that has 100 nodes, 10 items, and a grid of  $75 \times 75$  cells, we obtained the ratio of diffusion (equation 9) for  $t = 20.000s$ . The results are shown in figure 12 for different periods. We can see that the ratio of diffusion for 0.5 and 1s, are very similar, but for higher periods the results decrease, so the precision of the results become lower. The same figure shows the computation time,

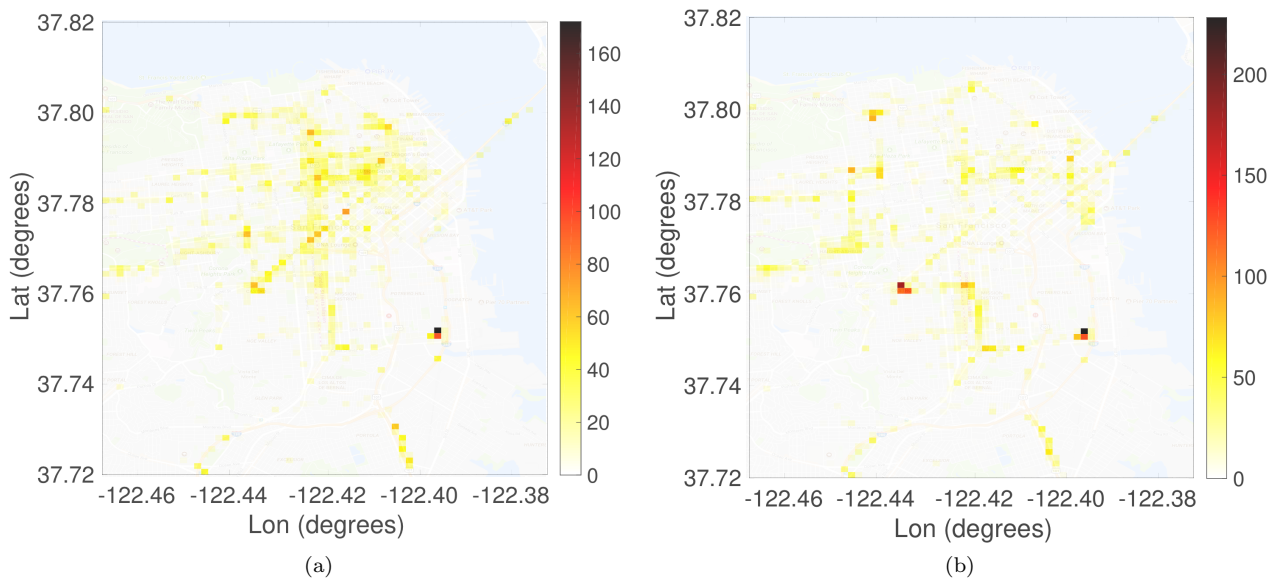


Figure 14: Density map of exchanged messages. It shows the number of messages exchanged in each cell. (a) With no restrictions on the communication; (b) With reduced communication in the centre.

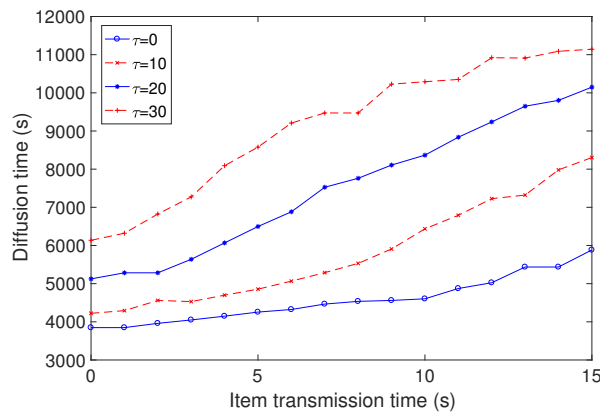


Figure 15: Diffusion time depending on the transmission times for different setup times.

430 where we can see that decreasing the period has a big impact on the computation time, especially  
 for simulation, that has a greater computation time than the FALCON model. Similar experiments  
 performed using the other scenarios obtained similar results with regard to the precision. Summing  
 up, a 1s-period is a good trade-off between precision and computation time.

### 5. Case study: impact of transmission time

435 In most opportunistic simulators, the set-up time and probability of communication are not  
 considered, being key factors for the performance of information diffusion. In this section, we  
 consider the communication time, using the FALCON model detailed in subsection 3.4. We used

the San Francisco scenario again with the first set of items considering three different sizes for the items to exchange and fixing the setup time to  $\tau = 10s$ .

- 440 • Text items: the items consist of very short texts, so the item transfer time is very low. We consider a time of  $0.1s$  for all items, that is  $t(i) = 0.1s \quad \forall i$ .
- Image items: the items transfer time is increased to  $5s$  for transmitting images of about  $1MB$ .
- Video items: the items transfer time is increased to  $20s$  for transmitting short videos.

445 For each item type we repeated the experiments of subsection 4.3. The results are shown in figure 13. First, we can see that the curves for FALCON and Simulation are very similar, confirming the precision of the FALCON model. Regarding the results, we can see, as expected, that by increasing the items transfer time the diffusion becomes slower. Besides, we can see that for video items the diffusion is very slow.

450 It is also interesting to know in which locations (cells) are messages exchanged. Using FALCON, it is easy to obtain a density map with the number of messages exchanged in each cell, as shown in figure 14a. This density plot corresponds to one of the previous experiments, specifically the one with text items (with the other items sizes the density plots are very similar). We can see that most of the messages are exchanged in the city centre and on the main roads, that is, the locations  
455 with greater contact opportunities. Using the communication grid, we can impose, for instance, that the communication in the city centre is reduced due to interferences. To model this situation, we set the communication grid to one for all the cells, except for the cells between rows 35 to 65 and columns 20 to 45 (approximately between latitude range  $[37.76^\circ \dots 37.81^\circ]$  and longitude range  $[-122.44^\circ \dots -122.40^\circ]$ ), that was instead set to 0.5. This probability reflects that only half of the  
460 opportunities to exchange messages are actually feasible. The results in figure 14b show that the density map changed slightly, evidencing the use more locations outside the city centre.

Finally, we can evaluate the impact of both the setup time and the items transfer time on the diffusion time of all items using expression 9, with  $R = 1$ . In this case, we only used FALCON, and the results are shown in figure 15. Four different curves are shown in this figure corresponding to  
465 the setup times  $\tau = \{0, 10, 20, 30\}s$ . The item transfer time varies from 0 to  $15s$ . We can see that the impact of the item transfer time is greater when the setup time is greater. The main reason is that the remaining time of the contact is reduced, meaning that fewer items can be exchanged. It is worth mentioning that the whole computation time of the previous evaluation was 40 minutes (corresponding to 60 individual evaluations). The same evaluation using a simulator would take  
470 about 25 minutes for each evaluation, that is, about 25 hours in total.

Summing up, by using FALCON we can see the impact of the transmission time in a given area, and the locations where these communications take place. We can even evaluate what happens if some areas have lower communication performance.

## 6. Conclusions

475 In this paper, we have introduced a new evaluation model called FALCON (*Fast Analysis, using a Lattice Cell model, of Opportunistic Networks*) based on a discrete dynamic system. We considered the discretisation of the space (cells) and time (period) to reduce the computational complexity of simulations.

The model, which is easy to use and implement, allows the evaluation of opportunistic networks with a precision comparable to current simulators and provides significantly more accuracy than analytical models. We showed that the computational cost of FALCON is extremely low. Specifically, when it is compared with simulation, computation time is reduced by two orders of magnitude (from hours to seconds). This allows researchers and practitioners to quickly evaluate many more different scenarios and having different network characteristics.

Finally, we demonstrated that this model is easy to adapt and expand to consider different scenarios and protocols. As a case study, we considered the transmission time, including both the setup time and the items transfer time, and showed how these parameters effectively affect message diffusion. As a future work, on the one hand, we plan to include in the model more evaluation elements such as the transmission cost and the buffer limitation; and on the other hand, we intend to implement the FALCON model as a stand-alone tool so it could be used directly or even integrated into other simulators.

## Acknowledgments

This work was partially supported by *Ministerio de Economía y Competitividad*, Spain, grant TEC2014-52690-R.

## References

- [1] M. Khabbaz, C. Assi, W. Fawaz, Disruption-tolerant networking: A comprehensive survey on recent developments and persisting challenges, *Communications Surveys Tutorials*, IEEE 14 (2) (2012) 607–640.
- [2] S. Trifunovic, S. T. Kouyoumdjieva, B. Distl, L. Pajevic, G. Karlsson, B. Plattner, A decade of research in opportunistic networks: Challenges, relevance, and future directions, *IEEE Communications Magazine* 55 (1) (2017) 168–173.
- [3] J. Dede, A. Förster, E. Hernández-Orallo, J. Herrera-Tapia, K. Kuladinithi, V. Kuppusamy, P. Manzoni, A. bin Muslim, A. Udugama, Z. Vatandas, Simulating opportunistic networks: Survey and future directions, *IEEE Communications Surveys and Tutorials* 20 (5) (2018) 1547–1573.
- [4] A. Keränen, J. Ott, T. Kärkkäinen, The ONE simulator for DTN protocol evaluation, in: *Proceedings of SIMUTools'09*, 2009, pp. 55:1–55:10.
- [5] OMNeT++, Home page. <http://www.omnetpp.org> [accessed on september, 2017] (2002).
- [6] M. Conti, S. Giordano, Mobile ad hoc networking: milestones, challenges, and new research directions, *IEEE Communications Magazine* 52 (1) (2014) 85–96.
- [7] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, H. Zedan, A comprehensive survey on vehicular ad hoc network, *J. Netw. Comput. Appl.* 37 (2014) 380–392.
- [8] Z. J. Haas, T. Small, A new networking model for biological applications of ad hoc sensor networks, *Networking, IEEE/ACM Transactions on* 14 (1) (2006) 27–40.
- [9] X. Zhang, G. Neglia, J. Kurose, D. Towsley, Performance modeling of epidemic routing, *Computer Networks* 51 (10) (2007) 2867 – 2891.
- [10] C. S. De Abreu, R. M. Salles, Modeling message diffusion in epidemical DTN, *Ad Hoc Networks* 16 (2014) 197–209.
- [11] Q. Xu, Z. Su, K. Zhang, P. Ren, X. S. Shen, Epidemic information dissemination in mobile social networks with opportunistic links, *Emerging Topics in Computing, IEEE Transactions on* 3 (3) (2015) 399–409.
- [12] R. Groenevelt, P. Nain, G. Koole, The message delay in mobile ad hoc networks, *Performance Evaluation* 62 (2005) 210–228.
- [13] T. Spyropoulos, K. Psounis, C. Raghavendra, Efficient routing in intermittently connected mobile networks: The multiple-copy case, *Networking, IEEE/ACM Transactions on* 16 (1) (2008) 77 –90.
- [14] E. Hernandez-Orallo, M. Serrat Olmos, J.-C. Cano, C. Calafate, P. Manzoni, CoCoWa: A collaborative contact-based watchdog for detecting selfish nodes, *Mobile Computing, IEEE Transactions on* 14 (6) (2015) 1162–1175.
- [15] M. Karaliopoulos, Assessing the vulnerability of DTN data relaying schemes to node selfishness, *Communications Letters, IEEE* 13 (12) (2009) 923 –925.



- [16] J. Whitbeck, V. Conan, M. Dias de Amorim, Performance of opportunistic epidemic routing on edge-markovian dynamic graphs, *Communications, IEEE Transactions on* 59 (5) (2011) 1259–1263.
- [17] U. of Dartmouth, CRAWDAD data set, Downloaded from <http://crawdad.cs.dartmouth.edu>.
- 530 [18] N. Banerjee, M. D. Corner, D. Towsley, B. N. Levine, Relays, base stations, and meshes: Enhancing mobile networks with infrastructure, in: *Proceedings of the 14th ACM International Conference on Mobile Computing and Networking, MobiCom '08*, ACM, New York, NY, USA, 2008, pp. 81–91.
- [19] V. Vukadinovic, S. Mangold, Opportunistic wireless communication in theme parks: A study of visitors mobility, in: *Proceedings of the 6th ACM Workshop on Challenged Networks, CHANTS '11*, ACM, New York, NY, USA, 535 2011, pp. 3–8.
- [20] N. Ristanovic, G. Theodorakopoulos, J. Y. L. Boudec, Traps and pitfalls of using contact traces in performance studies of opportunistic networks, in: *2012 Proceedings IEEE INFOCOM*, 2012, pp. 1377–1385.
- [21] O. R. Helgason, K. V. Jónsson, Opportunistic networking in OMNeT++, in: *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops, Simutools '08*, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 540 ICST, Brussels, Belgium, Belgium, 2008, pp. 82:1–82:8.
- [22] Z. Zhao, B. Mosler, T. Braun, Performance evaluation of opportunistic routing protocols: A framework-based approach using OMNeT++, in: *Proceedings of the 7th Latin American Networking Conference, LANC '12*, ACM, New York, NY, USA, 2012, pp. 28–35.
- 545 [23] N. Papanikos, D.-G. Akestoridis, E. Papapetrou, Adyton: A network simulator for opportunistic networks, [Online]. Available: <https://github.com/npapanik/Adyton> (2015).
- [24] M. J. Alenazi, Y. Cheng, D. Zhang, J. P. Sterbenz, Epidemic routing protocol implementation in ns-3, in: *Proceedings of the 2015 Workshop on ns-3*, ACM, Barcelona, Spain, 2015, pp. 83–90.
- [25] M. Piorkowski, Mobility-centric geocasting for mobile partitioned networks, in: *Network Protocols, 2008. ICNP 2008. IEEE International Conference on*, 2008, pp. 228–237.
- 550 [26] K. Garg, S. Giordano, A. Förster, A study to understand the impact of node density on data dissemination time in opportunistic networks, in: *Proceedings of the 2nd ACM Workshop on High Performance Mobile Opportunistic Systems, HP-MOSys '13*, ACM, New York, NY, USA, 2013, pp. 9–16.
- [27] J. Tao, Y. Xu, C. Tan, X. Wang, Location-aware opportunistic forwarding in mobile opportunistic networks, in: *2015 IEEE Wireless Communications and Networking Conference (WCNC)*, 2015, pp. 1847–1852.
- 555 [28] C. C. Lo, Y. H. Kuo, J.-Y. Jiang, Data dissemination strategy based on time validity for opportunistic networks, in: *2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN)*, 2016, pp. 1040–1045.
- [29] J. Herrera-Tapia, E. Hernández-Orallo, A. Tomás, P. Manzoni, C. Tavares Calafate, J.-C. Cano, Friendly-sharing: Improving the performance of city sensing through contact-based messaging applications, *Sensors* 16 (1523) (2016) 1–17.
- 560 [30] A. Lindgren, A. Doria, O. Schelén, Probabilistic routing in intermittently connected networks, *SIGMOBILE Mob. Comput. Commun. Rev.* 7 (3) (2003) 19–20.
- [31] 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, WDTN '05*, ACM, New York, NY, USA, 2005, pp. 252–259.
- 565 [32] N. Aschenbruck, R. Ernst, E. Gerhards-Padilla, M. Schwamborn, Bonnmotion: A mobility scenario generation and analysis tool, in: *Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques, SIMUTools '10*, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, Belgium, 2010, pp. 51:1–51:10.
- 570 [33] E. Hernández-Orallo, J. C. Cano, C. T. Calafate, P. Manzoni, New approaches for characterizing inter-contact times in opportunistic networks, *Elsevier Ad Hoc Networks* 111 (2016) 160–172.
- [34] L. Pajevic, G. Karlsson, O. Helgason, Epidemic content distribution: empirical and analytic performance, in: *Proceedings of the 16th ACM international conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, MSWiM '13*, ACM, New York, NY, USA, 2013, pp. 335–340.
- 575 [35] S. Kosta, A. Mei, J. Stefa, Large-scale synthetic social mobile networks with SWIM, *IEEE Transactions on Mobile Computing* 13 (1) (2014) 116–129.
- [36] T.-C. Tsai, H.-H. Chan, NCCU trace: social-network-aware mobility trace, *Communications Magazine, IEEE* 53 (10) (2015) 144–149.
- 580 [37] M. Piorkowski, N. Sarafijanovic-Djukic, M. Grossglauser, A Parsimonious Model of Mobile Partitioned Networks with Clustering, in: *The First International Conference on COMMunication Systems and NETWORKS (COMSNETS)*, 2009.