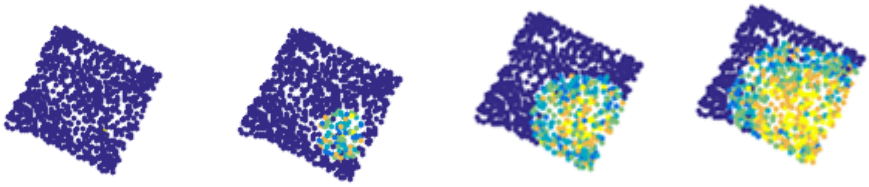
Imperial College London

Department of Electrical and Electronic Engineering

Final Year Project Report 2016



Project Title:	**Diffusion Source Detection in Social Networks with Multiple Observations**
Student:	**Roxana Alexandru**
CID:	**00743319**
Course:	**4T**
Project Supervisor:	**Professor Pier-Luigi Dragotti**
Second Marker:	**Dr Tania Stathaki**

Abstract

As social networks have developed and the spreading of information has greatly amplified, the dynamics of information dissemination within a network have attracted considerable attention in the past years. Recently, however, several authors have begun considering the more challenging reverse problem, of detecting the source responsible for the spreading of rumors. The state-of-the-art approaches focus on detecting rumor sources in simple topologies such as trees or random geometric graphs, based on the ideal assumption that there is access to information at all the nodes in the network.

This project addresses the problem of estimating the source of an infection on a general graph of known topology, using observations from a finite set of monitor nodes, at well-known times after the initial infection. This report describes the network topologies, the infection model and the Matlab environment required to simulate a spreading of rumors, and the mathematical formulas which model the probability of rumor dissemination. The theoretical derivations are compared against observations from the randomly selected sensor nodes, in order to allow the exploitation of an algorithm for the inference of the rumor source. Experiments were carried out on synthetic networks, and the results obtained show the convergence of the derived theoretic probabilities to the ones obtained through simulations, as well as an accurate identification of the rumor source, with higher probability of correct detection compared to state-of-the-art solutions.

Acknowledgements

I would like to extend my gratitude to my project supervisor Prof. Pier-Luigi Dragotti for giving me the opportunity to work this very interesting and challenging project, for his invaluable advice and continuous support and insightful feedback at every stage of the project.

I would also like to express thanks to John Murray-Bruce for his support, help in brainstorming and in overcoming challenges, as well as encouragement throughout the project.

Last but not least, I would like to thank my parents for their love, and continuous support throughout my studies, and for providing me with the opportunity of an outstanding education at Imperial College London.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This Project Report aims to give a description of the problem of localizing diffusion sources of rumors in social networks, and to present a mathematical formulation of a solution to this problem, as well as the evaluation methods used to assess the performance of this solution.

The structure of this report is the following. Firstly, Chapter 1 presents a mathematical formulation of the problem. Chapter 2 provides a description of the state-of-the-art approaches related to this problem, and lays out the required mathematical and graph theory-related background.

Furthermore, Chapter 3 describes the analysis and design involved in the derivation of a mathematical formulation for the theoretic probability of rumor spreading in a network of arbitrary topology. The design includes the initial derivation of the probability of dissemination, as well as a refined solution to the problem.

Moreover, Chapter 4 presents a description of the Matlab environment, including the design of the synthetic networks used for simulations and the epidemic model used to simulate the spreading of rumors. In addition, a rumor source detection algorithm is presented, with a description of the motivation behind each algorithm enhancement.

Chapter 5 presents the evaluation methods used to assess the performance of the estimation algorithm, which includes tests assessing the individual enhancements presented in Chapter 4, as well as tests for the complete algorithm. Finally, the results are discussed in Chapter 6, and conclusions are drawn in Chapter 7, which include any future works to be conducted.

Motivation

The aim of this project is to successfully infer the source responsible for spreading of data within a certain network, motivated by applications such as: localizing individuals who set trends in social networks and who successfully spread rumors or images, determining the causes of cascading failures in large systems such as financial markets or sensor networks, finding the contaminant in a water distribution network, and identifying the origin of infectious diseases.

One of the most interesting of these applications is the dissemination of information in social networks and finding the influencer in this case could be of great interest, for example, in the case of identifying the leader of a spy or political network. Described as the *Achilles' heel* property of social networks, this property ensures that such networks are robust to random failures, but fragile to attacks. Moreover, as social networks expand and become more popular, the information propagation becomes faster and less controllable, with some influential people having the power to disseminate pervasive rumors without confirmation or certainty to

facts. Consequently, finding the source of the rumor is very important, as it could help control and prevent the risk of allowing information to be disseminated within the network, as well increase the resistance of the network to attacks.

The project could also motivated by other interesting applications such as: determining the causes of cascading failures in large systems such as financial markets or sensor networks, identifying the origin of infectious diseases or computer viruses, or identifying the leader of a spy or political network. Some other applications of interest could be biological systems such as metabolic networks, protein-protein interaction networks, or diseases links through shared genes.

It is also of interest to analyse the problem of detecting multiple diffusion sources in the network. For example, recurring email spam and virus attacks are generally organized by criminal networks. Hence, if we consider the detection of multiple sources, we can assume that these sources are connected, consequently identifying any of them would provide necessary information to further identify all the rumor sources [1].

Mathematical Formulation of the Problem

Definition of a Rumor

A rumor can be defined as a story, a statement or any other type of information such as images, which enters circulation within a network. Generally a piece of information can either true, false or unknown, based on the judgements made after the spreading phenomenon. Only the latter two will be defined as rumors, while the former is an information confirmed as true after some time after the spreading [2].

Rumor Spreading Model

The model assumes a uniform prior probability of the source node among all nodes in the network. This assumption ensures tractability and is common in literature [1].

The problem of detecting who is spreading rumors in a social network can be translated to the problem of estimating the source node which starts disseminating information within a network of fixed topology. We will assume a susceptible-infected (SI) epidemic model, where there are two types of nodes:

- Susceptible nodes: which are not infected with the rumor yet;
- Infected nodes: which have the rumor and can spread it to any other node, including already infected nodes.

The main approach is to relate this problem to the one of estimating the sources of continuous diffusion fields. In this case, the recent results developed in the Communications and Signal Processing group address the problem by taking discrete spatiotemporal measurements of the field obtained with a network of arbitrarily distributed sensors, and by providing reconstruction schemes to recover the field by estimating the sources that induced it [3].

In a similar manner, the problem of finding the source of rumors in social networks will be addressed by collecting data from a network of randomly chosen sensors and using the available data to localize the source of rumors in the network. Nevertheless, in the case of a fixed topology network, the problem becomes more challenging as the estimation depends on the fixed topology of the network. This makes it difficult to assign a well-defined location to the nodes in the network, compared to the case of the physical phenomena of diffusion where we can give a precise location in space to the points where the field spreads. In addition, the

topology affects the dynamics of the spreading of rumors due to additional connectivity constraints, compared to a continuous field where there is no notion of connectivity between any points in the diffusion medium. [4]

Assumptions

The following assumptions are made and justified below:

a. Network Topology: We will study the dynamics of spreading of rumors and test the source estimation algorithms on the following network types: tree structure, random geometric graph, small world graph and scale-free networks. The motivation for choosing these network types is the following: the tree represents a simple graph which allows more insight into the dynamics of rumors, the random geometric graph is a mathematically simple spatial network with real-world applications in the modelling of ad-hoc networks, while the small-world and scale-free properties describe more complex graphs and are generally used to model the social networks.
b. Network Topology: The number of nodes in the network, and in addition, the network topology (the connections between any two nodes) are known in advance. This agrees to a real-world application, where the connections between the members of a social network are known.
c. Network Evolution: We will assume that the network is constant at least for the duration of the observation. Hence, the connections between the nodes in the network are fixed. In addition, there is no network growth over the time window when observations are taken.
d. Number of Sources: The research will focus on estimating a single instantaneous source of spreading of rumors. Moreover, as a future development, the solution could be adapted to the problem of detecting multiple rumor sources.
e. Number of Rumors: The main motivation of this project is to successfully localize the information dissemination source in a network. This assumes that the source of rumor will start a large number of attacks (as is the case of a source of rumors in a social networks or a hacker launching a series of (viral) attacks on an institution's infrastructure etc.). Therefore, the model assumes that data resulted from multiple rumors will be available (e.g. 20 rumors).
f. Rumor Persistence: The source is assumed to be instantaneous, with time-invariant intensity, based on the fact that typically, the information does not change while being transmitted from one person to another. In addition, once a node is infected with the rumor information, it will not be possible for it to eliminate the information.
g. Probability of Rumor Spreading: We will assume that the rumor spreads with constant probability, defined as the probability to pass the rumor between any two connected nodes. Even though in a social network, some people have a higher tendency to spread the rumors than others, we can assume that on average, the rumor will spread with a constant probability.
h. Monitoring Nodes: The sensor nodes are selected randomly from the set of all nodes in the network. Since we have access to a limited number of sensor nodes, we can gain additional information through measurements over time.
i. Time Measurements: We will assume that we have access to measurements at various nodes in the network, at well-defined time instants after the rumor spreading has started. This is the only assumption which may not agree entirely with a real-world application, as in a real-life case we would not be able to precisely know at what point in time we are taking measurements in the network, after the source emission has started. However, this assumption is necessary in the initial phase in order to provide a starting point for tackling the problem.

Project Aims and Objectives

The main project deliverables are the following: research and clear formulation of the problem, derivations of expressions to recover a single source responsible for spreading of multiple rumors, development of Matlab environment and Matlab simulations to evaluate the proposed solutions and the algorithms developed, on synthetic data.

The following objectives have been achieved. Firstly, the initial requirement of the project revolved around diffusion processes, Markov chains and random walks and how these theories could be applied to the problem of spreading of rumors in a social network. A set of research papers have been read and understood and more in-depth knowledge was gained using other materials such as books or reviewing previous modules. The initial research helped understand the dynamics of the rumor spreading in networks, as a similar phenomenon to a diffusion field and how this could be related to a Markov process.

Secondly, a Matlab environment has been set-up in order to better visualize the dynamics of rumor spreading and to understand how the different network topologies and parameters affect this spreading. This environment consists of a network defined through a matrix, and artificially generated, rumor spreading processes. This gave an initial understanding of the dynamics of the rumors in various network topologies: tree, random geometric graph, small-world and scale-free graph.

The subsequent research focused on identifying related research topics and on the state-of-the-art solutions to the problem of detection of the diffusion source in social networks. This helped further understand the problem, the challenges associated with it, and various evaluation methods typically used for the solutions proposed.

Based on the research of the state-of-the-art solutions, as well as vast research concerning topics such as graph theory, statistics, probability distributions, or mathematical simplifications, an initial approach to the problem was developed. This approach has not been studied before, and involves the derivation of an analytical formulation for the theoretic probability of a node being infected, as a function of the time since the rumor initiation, as well as of the shortest distance to the source emitting the rumor. The research completed as part of this approach includes: shortest-path Dijkstra algorithm, mathematical tools such as Stirling's formula, approximating the sum of binomial distribution into a Gaussian distribution, topics related to Markov processes such as path counting of constrained random walks.

The mathematical formulation of the theoretic probability of rumor infection further lead to the development of an algorithm for detection of the source. Herein, robust schemes will be developed, which have a high probability of correct detection, on an arbitrary network whose topology is known and in particular on graphs which accurately model the properties of a real social networks, such as small-world or scale-free.

The problem of recovering multiple sources responsible for the spreading of data within a network is a challenging problem and will remain as part of the requirements for future work. Moreover, some other topics which will be included in the future works are: relaxing the assumption that the rumor spreads with constant probability within the network, or assuming that the time at which we take sensor measurements is unknown.

Challenging Aspects

The problem of identifying rumors and their sources in social networks is a hard problem to solve, which has largely remained unexplored until recently.

Besides this, one other challenging aspect of the project is due to the limited related research literature on the topic of rumor detection in a social network. Most of the current research focuses on identifying how the dynamics of various networks affect the spreading of rumors, and not on the inverse problem of detecting the source. In addition, there exists some related research, which aims to find a detection algorithm, however the problem is often over-simplified by: in some cases only simple networks such as tree graphs are considered, others assume that we have snapshots of all the infected nodes in the network.

Furthermore, deriving a precise analytical formula for the probability of spreading of rumor is mathematically challenging and the approximations used to simplify these derivations may decrease the accuracy of the results, hence leading to erroneous detection of the source. Moreover, modelling complex real-world networks could be challenging. In addition, a wide range of simulations and analysis of results are required in order to understand the dynamics of spreading of rumors in various networks, and to evaluate the source detection solution.

Chapter 2

State-of-the-Art

Survey of Related Literature

This section gives an overview of the relevant literature research, summarizing some of the state-of-the-art approaches used to solve the problem of rumor source detection.

In the paper **"Rumors in a Network: Who's the culprit?"** [5] the authors propose a rumor spreading model based on the susceptible-infected (SI) model. In this model, there are two types of nodes: nodes susceptible to infection, and nodes which currently infected and can thus continue spreading the rumor. The paper then addresses the source estimation problem, simplified by considering the case of regular trees, where every nodes has the same degree, and where only one node can be a source of rumors. The estimation is done by modelling the time for a node i to transmit the rumor to its neighbor j as an exponential random variable. Since each node is equally likely to be the source, the best estimator of the actual rumor source will be the Maximum Likelihood (ML) estimator. The ML estimator is given by $\hat{v} = \arg\max_{v \in G_N} P(G_N | v^* = v)$, where v^* is the actual rumor source. The paper then proves the fact that in a regular tree network, the ML estimation is equivalent to a combinatorial problem, if we have access to the rumor graph, i.e. if we know exactly all the nodes that have the rumor at a certain time, which form a subgraph G_N. This is equivalent to a metric called *rumor centrality* which represents the likelihood of a particular node to be a source node and hence, the source would be the node in the infected subgraph with the highest rumor centrality. The evaluation of the solution is given by calculating the rumor source estimator detection probability for line and geometric trees, versus the number of nodes in the graph and for various values of the parameter characterizing the tree, denoted by α. This parameter is used to give upper and lower bounds on the maximum number of nodes located at a distance d from a node. In addition, the solution is also evaluated by looking at the estimator error, given by the number of hops between the estimated source and the actual one. It is shown that the detection probability of the rumor source estimator is approximately $P = 0.9$, for a small geometric tree with less than 100 nodes, decreasing to $P \cong 0.2$ for a size of $N = 400$ nodes. In both cases, the parameter of the regular tree is $\alpha = 0$. When the parameter $\alpha = 1,2,3, or\ 4$, the probability of correct detection is $P \in [0.9,1]$. In addition, the frequency of an estimator error equal to $e = 1\ hop$ is approximately 80%. The algorithm was also tested on small-world and scale-free networks, where the performance is reduced compared to the case of tree graphs. In this case, the source estimator error is 0 only in 15% of the time.

In the paper **"Rumor centrality: A universal source detector"** [6] the authors extend the solution to random graphs. In this work, the authors propose an approach that takes advantage of knowing all infected nodes in the graph. As such, this approach might not be best suited for a real-world application, considering the challenges of having access to this information, as well as the complexity of the algorithm for large network sizes.

In the paper **"Spotting Culprits in Epidemics: How many and which ones?"** [7] the authors provide an algorithm to identify the likely sets of source nodes, given a snapshot of the network after the rumor has been spreading for some time. This is achieved through the Minimum Description Length method, which simulated the spreading of rumors starting from the estimated set of seed nodes and chooses the set which best described the given snapshot. The best set of nodes can be identified without knowing the number of spreaders a priori.

In the paper **"Rumor Source Detection under Probabilistic Sampling"** [8] the authors analyse the problem where the nodes in the network randomly report their infection state, hence having access to an incomplete snapshot of the infection state. The evaluation is done on regular trees, using the susceptible-infected model.

In the paper **"Inferring the origin of an epidemic with a dynamic message-passing algorithm"** [9] the authors study the problem of detecting the single source of an epidemic outbreak, by having access to a snapshot of the network at a certain time and using the susceptible-infected-recovered model. The algorithm proposed is based on dynamic message-passing equations, giving the probabilities that a certain node i is in a given state at time t, where the possible states are susceptible, infected or recovered.

Furthermore, in the paper **"Identifying Rumors and their sources in social networks"** [10] the authors are analysing the problem of finding the rumor source using observations at a finite set of monitors. The algorithm proposed finds a minimal set of candidate sources, based on the number of infected nodes that the candidate source reaches (which should be high for a more likely source), as well as the number of susceptible nodes that can be reached from the source (which should be low for a more likely source). The method is evaluated using different strategies of selecting the monitor nodes, such as random selection or selection based on the largest *betweeness centrality* which depends on the distance and number of edges between the monitor nodes. The solution is evaluated on a directed graph of 30146 nodes. The results show that as the number of monitors increases, the rank of the actual source decreases. For example, for 20 monitors (0.06%), the rank of the actual source is approximately 1000, dropping to below 10 when more than 650 monitors (2.15%) are used. In addition, the distance between the main suspect (rank 1) and the actual source is in all cases smaller than 3 hops.

In the paper **"Routing out the rumor culprit from suspects"** [11] the authors are using *a priori* knowledge of the set of suspect nodes and a single observation of all the nodes in the network, in order to construct a maximum *a posteriori* estimator to identify the rumor source. This is based on the assumption that in a real-life application, some individuals might be more likely to initiate the rumor spreading, or another example are the frequent travellers who will be more likely to cause an epidemic outbreak. The evaluation of the method is performed on a regular tree network of 1000 nodes, where the infection is started by a source randomly selected from a set of suspects. The results show that as the suspect size decreases, the probability of correct detection increases. For example, when the set of suspects has cardinality $k = 2$, the detection probability is $P \cong 0.55$ for a node degree of $\delta = 3$, increasing for a larger node degree of $\delta = 20$ to $P \cong 0.95$. For a larger suspect size, there is a small drop in the correct detection probability. In summary the authors consider the problem of identifying a single source out of a pre-defined set of suspected nodes, using the susceptible-infected model, along with a single observation of the entire network. The results prove that the performance of the detection algorithm is improved when a set of suspects is known. Nevertheless, this assumption, as well the assumption of having access to the state of the entire network, might be unrealistic for most real-world applications.

In the paper **"Rooting our Rumor Sources in Online Social Networks: The Value of Diversity from Multiple Observations"** [1]**,** the authors address the problem of detecting the source of rumor spreading, using multiple observations, which increase the reliability of source detection in a network. The authors study the problem of a single rumor source and evaluate the solution in degree-regular trees. In addition, the case of multiple

connected sources is also studied for general trees, as well as general graphs. Moreover, the detection algorithm assumes that multiple snapshots of the entire network are available, i.e. all the nodes need to be observed. The source detection method consists of observing the entire network at some time and finding a subset of infected nodes. Then, a Maximum Likelihood detector is calculated for each potential source s, as the maximum of the probability of observing the subset of infected nodes, assuming the rumor was initiated at node s. The detection algorithm is evaluated by computing the asymptotic correct detection probability against the node degree in a regular tree. As the node degree increases, so does the detection probability asymptotically. For example, for the case of two independent observations, and a node degree $d = 3$, the probability of detection is $P = 0.5$, while for a node degree $d = 16$, $P = 0.9$. Under three independent observations, a node degree $d = 6$ is sufficient to obtain a probability of detection of $P = 0.9$. Hence, it can be seen that the authors also show that in addition to the diversity of observations, richer connectivity also enhances the detection. They also evaluate their method by looking at the frequency of the detection error, measured as the shortest path between the estimated source and the real rumor source. The results obtained showed that for the estimation of a single source using k observations of the entire scale-free network, the frequency of $no_{hops} = 0$ is $f < 5\%$ for $k = 1$, increasing to $f = 90\%$ for $k = 5$ observations. The method performs less well for a small-world network, where the frequency of correct detection is $f < 20\%$ for any number of observations $k \leq 5$. The method using multiple observations leads to a better performance in the case of multiple connected sources as well. While the method proposed in this paper is highly performant, it requires knowledge of the state of all the nodes in the network, for multiple observations. This would be hard to achieve in a real-world application, where the number of nodes in the network can be several orders of magnitude.

In the paper **"A fast Monte Carlo algorithm for source localization on graphs"** [12] the authors describe a method of estimating the source of rumors from a small set of sensor nodes, by considering measurements within a fixed time interval at some unknown time after the initial rumor spreading. Within the considered time window, it is assumed that the nodes observed could be classified in the following three categories: infected nodes (which already have the rumor at the time window), susceptible nodes (nodes which do not have the rumor throughout the duration of the time window, but could receive it) and transition nodes (which will get the rumor for the first time at a certain time within the considered window). Hence, the set of observers can be partitioned as follows:

$$O = O_T + O_I + O_S, \text{ where } \begin{cases} O_T = nodes\ which\ transition\ from\ susceptible\ to\ infected \\ O_S = susceptible\ nodes \\ O_I = infected\ nodes \end{cases}$$

Moreover, the authors define the index of the first observation at which a node transitions to an infected state as:

$$m_i = \{1,2, \dots T-1\}, where\ T\ is\ the\ lenght\ of\ the\ time\ window.$$

In addition, it is assumed that the infection time of node indexed 1 is τ_1, and hence the relative infection times will be $\tau_i - \tau_1$, for each node i. Hence, for each potential source s, a log-pseudolikelihood function can be calculated as follows, assuming that the relative infection delays are independent:

$$l(s) = \sum_{i \in O_{T \setminus \{1\}}} logP\{\tau_i - \tau_1 = m_i - m_1 | s\} + \sum_{i \in O_S} logP\{\tau_i - \tau_1 \geq T - m_1 | s\} + \sum_{i \in O_I} logP\{\tau_i - \tau_1 \leq -m_1 | s\}$$

In the above equation, $m_i - m_1$ are known from observations. Moreover, we can approximate the infection times as independent Gaussian random variables. Hence, in order to estimate the marginal for the relative times, it remains to find an estimation for mean and variance of the infection times, corresponding to each source s. The authors achieve this by sampling the set $(\tau_1, \tau_2, \dots, \tau_n)$ for several iterations for each source s,

and finding the mean and variance of those samples. The sampling method consists of assigning the value of the shortest distance between the source node s and the monitor node i, $\tau_i = d(s,i)$. The log-pseudolikelihood function will be evaluated for each node, and the potential sources will be ranked according to the value of the function. The source estimation algorithm is evaluated by considering the cumulative distribution function for the rank of the source, i.e.
$P\{true\ source\ rank \leq rank\ interval\ I\ |a\ fraction\ f\%\ of\ observes\}$.
The tests have been performed on a random geometric graph of size $N = 100$ nodes. For example, for $f = 100\%$ (i.e. all the nodes in the network are observed), the probability that the real source is within the top 10 ranked sources is $P\{rank \leq 10\} \cong 0.9$. When $f = 10\%$, the same probability drops to $P\{rank \leq 10\} \cong 0.7$, while for $f = 5\%$, $P\{rank \leq 10\} \cong 0.5$. The method has been evaluated using the susceptible-infectious model, assuming a single rumor source, and for small regular networks, such as random geometric graphs and trees.

In the paper **"Spread of a Rumor"** [13]**,** the epidemic framework used is the susceptible-infected one, where each infected node is able to infect only one of its susceptible neighbours at any given time. The authors model the number of individuals to be told the rumor at a given discrete time, as a random variable with hypergeometric distribution. This is used to derive a difference equation characterizing the expected number of persons who know the rumor at a given time step.

Summary of State-of-the-art

In summary, most of the relevant research papers provide solutions to the problem of spreading of rumors in a social network, following the susceptible-infected model, where the nodes can either be infected or susceptible and once a node has received the rumor, it cannot recover from it.

In terms of topology, a significant focus of current methods is represented by tree-like topologies or random geometric graphs.

Moreover, most methods are based on the assumption of a complete snapshot of the network, which is generally difficult to achieve in practice.

Although most methods presently focus on detection of a single source and there are a few methods that can be used to identify multiple sources; however there are typically more computationally expensive and challenging to implement.

With regards to the rumor spreading probability, current methods assume that the infection probabilities are equal across the entire network. More recent methods have been extended to variable probabilities of infection across different edges, which gives a more realistic model.

In terms of complexity, most current methods are computationally-expensive, with complexity ranging from $O(NlogN)$ to $O(N^k)$, where N is the network size.

Network Topology

The social network will be described through a graph where the nodes represent individuals and the edges correspond to the interactions between them. It has been shown that most real-world networks exhibit the small-world and scale-free properties. Our network will be initially modelled as a small-world graph, and following the initial results obtained on this type of graph, the algorithms will also be applied to a scale-free network.

Some of the typical characteristics of complex networks are the clustering coefficient and the average distance between any two nodes. The clustering coefficient of a node is defined as the ratio between the numbers of existing edges between his neighbouring nodes, divided by the number of total possible edges that could exist between his neighbours, while the clustering coefficient of the network is the average over all the nodes. The average distance represents the number of edges corresponding to the shortest path between the two nodes [14].

In has been demonstrated that the small-world and scale-free effects are properties displayed by most real-life networks. These are neither completely regular nor completely random, and hence, their properties are a mixture between those of regular networks (great clustering coefficient, long average distance), and those of random networks (small clustering coefficient, short average distance) [14].

The fact that the node degree in real social networks follows a power-law degree distribution has been explained by models such as the Preferential Attachment Model*,* as proposed by Barabási, who suggests that the main difference between a random and a scale-free network is the fact that the scale-free network has a large number of small degree nodes, most of which are absent in a random network. Moreover, the probability of a high-degree node or hub is several orders of magnitude higher in a scale-free than in a random network. Furthermore, the more nodes a scale-free network has, the larger are its hubs, since the size of the hubs grows polynomially with network size [15].

Taking into account these two properties, the small-world and scale-free networks can be described as follows. The small world network is defined by a graph where most nodes are not neighbours of each other, but most nodes can be reached from any other node by a small number of steps. Small-world networks have small average distance and great clustering coefficient. The scale-free have even smaller average distance and great clustering coefficient and in addition, they exhibit the property that the degree distribution follows a power-law distribution. In other words, the probability of a node having k connections to other nodes is given by $P(k) \sim k^{-\gamma}$, where γ is a parameter in the range $(2,3)$. The degree distribution is the main difference between a scale-free and a small-world network, the latter having a Poisson distribution of the node degree. The existence of *hubs,* or high-degree nodes in scale-free networks will prove to be an important factor for the derivation of the probability of rumor spreading in such a network, as seen in the next chapter [14].

We should also note that in a small-world network the average distance scales as $L \sim lnN$, while in the scale-free network this scales as $L \sim lnN / \ln(lnN)$, where N is the number of nodes in the network [16].

We show in Figure 1 below, realisations of some common network models.

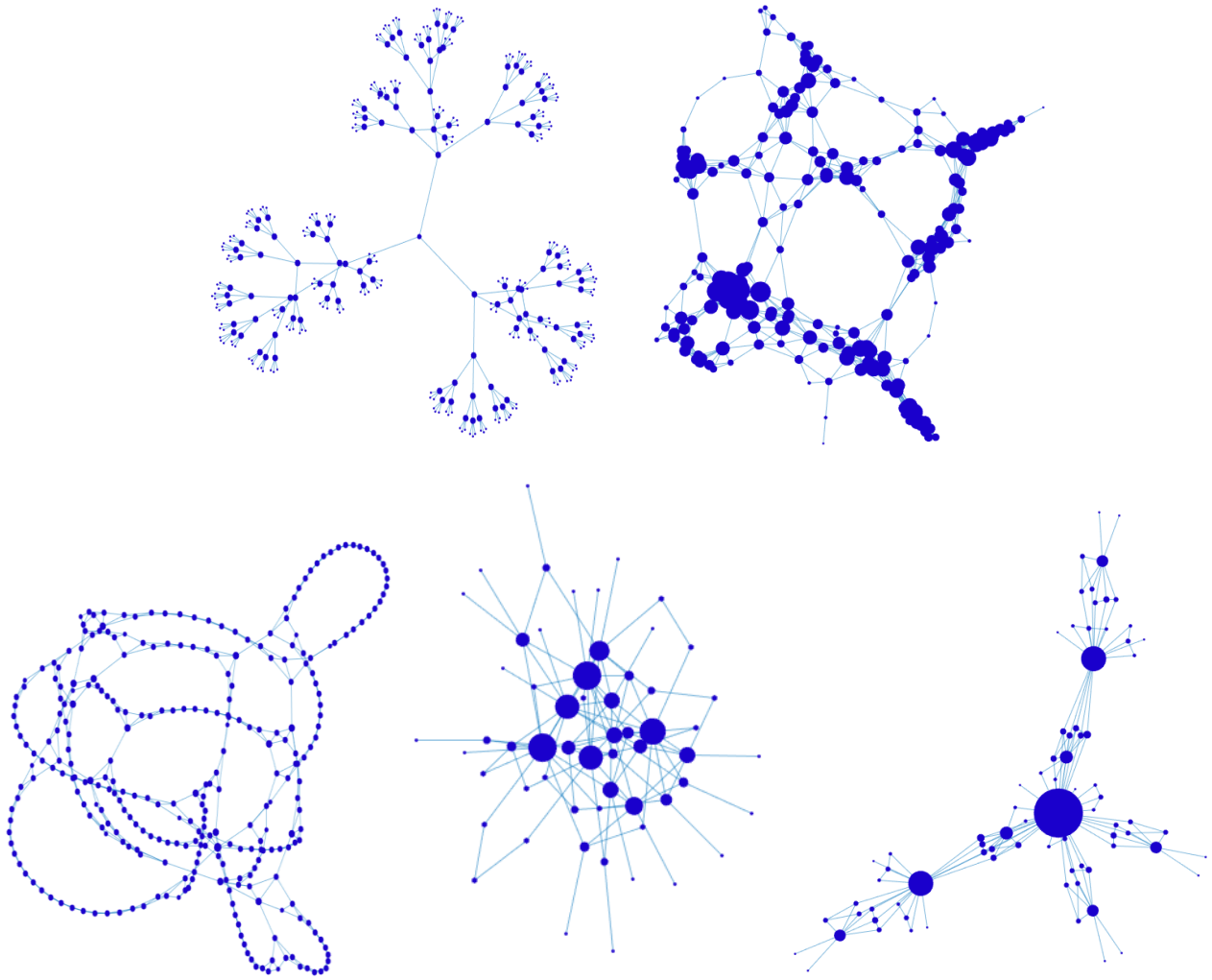


Figure 1: From top left to bottom right: Tree Graph, Random Geometric Graph, Small-world Network, Random and Deterministic Scale-free Network

Random Walks Theory

The background to the theory of random walks consists of the following: understanding of Markov Chains theory, understanding the random walk as a Markov Chain process, defining the spreading of information within the network as a random walk process, constrained on the dynamics of rumors in the fixed topology network [17] [18].

Since the spreading of rumor is a Markov process, some results of the random walk in $1D$ could be applied to our epidemic model (e.g. number of paths a random walk could follow to reach a certain point). General properties of Markov Chains and Random Walks are summarized below.

a. Markov Chains [17]

A Markov chain is a mathematical model of a random phenomenon evolving with time in a way that the past affects the future only through the present. In Mathematics, a phenomenon which evolves with time in a way that only the present affects the future is called a dynamical system.

A sequence has the Markov Property if for any random variable in the sequence, the future process (index m >n) is independent of the past process (index m<n) conditionally on X_n. In other words, the future is independent of past given the present. The random variables take values in some countable set S, called the State Space. The elements of S are frequently called States. Since S is countable, we call X_n a Markov Chain.

A simple example of a Markov Chain would be the case of a 2-state Space, where a mouse moves from Cage 1 (State X=1) to Cage 2 (State X=2) with probability p, and vice-versa with probability 1-p. This can be represented either through a state diagram, or through a Transition Probability Matrix, where each row contains the probability related to moving from a certain point to a different point, hence, the probabilities adding to 1 on each row.

b. Transition Matrix

A transition or stochastic matrix is a matrix used to describe the transitions of a Markov chain. Each entry P_{ij} in the matrix denotes the probability of transitioning from i to j in one step.
The properties of the stochastic matrix are the following:

1. $P_{ij} \geq 0, \forall i, j.$
2. The sum of the transition probability from a state i to all other states must be 1, i.e. $\sum_j P_{ij} = 1 \,. \forall i.$
3. The probability to transition from i to j in k steps is given by P^k.

c. General Properties of Random Walks [17]

A random walk is a special kind of Markov Chain, which possesses the additional properties of time and spatial homogeneity. Time-homogeneity means that the transition probability p_{xy} does not depend on time, while space-homogeneity means that the transition probability should depend on x and y only through their relative positions in space. This translate to $p_{x,y} = p_{x+z,y+z}$, for any translation z. In other words, given a function $p(x)$, a random walk is a Markov Chain with time- and space- homogeneous transition probabilities given by $p_{x,y} = p(y - x)$.

d. Random Walks in 1D [17]

Let us denote by S_n the state of the random walk at time n. Hence, this can be represented as $S_n = \varepsilon_1 + \varepsilon_2 + \cdots + \varepsilon_n$, where ε_i are the increments of the random walk starting from 0, i.i.d. random variables with common distribution $P(\varepsilon_n) = p(x)$.

In our derivation, we will be interested in calculating the n-step transition probability, i.e. the probability of reaching a state y located n steps away from the starting point x. This has the following expression: $p_{xy}^{(n)} = P_x(S_n = y) = P(\text{x}+\varepsilon_1 + \varepsilon_2 + \cdots + \varepsilon_n = \text{y})$.

For the random walk in 1D, the random vector represented by the n increments $(\varepsilon_1, \varepsilon_2, \ldots \varepsilon_n)$ can take values in $\{-1, +1\}^n$, since the walk can either move one step to the right or to the left. We are interested to find the probability of event A, where the A={random walk starts from (0,0) and ends up at (n,y)}.

The total possible paths in the 1D space, which are followed in n steps is given 2^n since at each step, there are only 2 possibilities for the increment, either to the right or to the left. Hence, the probability of each path is $P(\varepsilon_1 = \alpha_1, \varepsilon_2 = \alpha_2, \ldots + \varepsilon_n = \alpha_n) = 2^{-n}$.

Hence, $P_A = \frac{\#A}{2^n}$, which shows that if we can count the number of elements of A we can compute its probability.

If there are no additional constraints on the random walk, besides the specific starting and ending points, then the number of paths in A is given by $\#A = \binom{n}{\frac{n+y}{2}}$, which means that $P_A = \binom{n}{\frac{n+y}{2}} \times 2^{-n}$ [17].

e. Random Walks on General Networks: Rayleigh's Shortcut Method [19]

Let us assume we have a general network, with any two nodes i and j connected by edges to which we assign a resistance value R_{ij}. The conductance of this edge will therefore be $C_{ij} = 1/R_{ij}$.
We can define the transition matrix of a random walk on this network to be given by:
$P_{ij} = \frac{C_{ij}}{Ci}$, where $C_i = \sum_j C_{ij}$
If we assume that the resistance values are constant within the network, then the probabilities of a node to spread the rumor to any of its neighbours will be equal.
If a node is more likely to spread the rumor to only some of its neighbours, then the spreading probabilities will be inversely proportional to the resistance between the nodes, and hence, a higher spreading probability means a lower value of resistance.

Mathematical Methods for Expression Simplification

Stirling's Formula

In mathematics, Stirling's formula represents a powerful approximation for factorials, which leads to accurate results. The formula has the following form:

$$n! \sim \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$$

The following table illustrates the accuracy of the approximation:

n	Actual $n!$	Approximation $\sqrt{2\pi n}\left(\frac{n}{e}\right)^n$
1	1	0.922
2	2	1.919
3	6	5.836
4	24	23.506
10	3,628,800	3,598,695.619

Table 1: Stirling's Approximation Accuracy

De Moivre-Laplace Formula

The de Moivre-Laplace theorem is an approximation to the binomial distribution, given by the normal distribution. If the probability of success is p and the number of independent Bernoulli trials is n, then the normal distribution to which the probability mass function of the random number of successes observed has the following parameters: mean np, and standard deviation $\sqrt{np(1-p)}$.The below formula is valid for large values of n.

$\binom{n}{k}p^k q^{n-k} \cong \frac{1}{\sqrt{2\pi npq}} e^{-\frac{(k-np)^2}{2npq}}$, $where\ p+q=1, p, q>0$, and k is in a neighbourhood of np.

n	k	p	Actual $\binom{n}{k}p^k q^{n-k}$	Approximation $\frac{1}{\sqrt{2\pi npq}} e^{-\frac{(k-np)^2}{2npq}}$
5	2	0.5	0.3125	0.904
20	2	0.5	0.000181	0.000296
20	15	0.5	0.0147	0.0026
50	2	0.5	$1.08x10^{-12}$	$7.29x10^{-11}$
50	30	0.5	0.0418	0.0415

Table 2: De Moivre-Laplace Approximation Accuracy

Chapter 3

Analysis and Design

This chapter provides a description of multiple approaches considered for the problem of identifying rumor sources in social networks. For each approach, a high-level overview is given, as well as the motivation for the approach, any assumptions made, a more detailed mathematical description, and advantages and disadvantages.

Novel Aspects

The solution to the problem of estimating the sources of spreading of rumors in a social network will focus on relating this problem to that of estimating the localized sources of diffusion fields. The recent research into this field conducted within the Signals and Communications Group at Imperial College London presents an efficient method for the estimation of sources of diffusion fields in [3].

In this respect, one of the novel aspects of this design is represented by the calculation of the theoretic probability of rumor infection, as a function of the time delay since the rumor initiation and the distance from the source. The derivation of the theoretic probability is motivated by the similarity between the rumor dissemination within a network and the diffusion process of a physical phenomenon. In the case of a diffusion process, the source could be estimated using spatiotemporal samples of the field obtained through a sensor network [3]. Similarly, if we could find the intensity of the rumor as a function of space and time, then we could use measurements at some monitoring nodes in order to retrieve the rumor source. Nevertheless, the latter problem is more challenging due to the additional constraints imposed by the network, such as the fact that there is no exact notion of location of the nodes.

Theoretic Probability of Rumor Dissemination: Initial Solution

This approach formulates the spreading of rumors as a random walk process in $1D$. The nodes in the network will be arranged based on the length of the shortest path to the source who starts the spreading of rumors. Analytical formulas for the probability that a node located at a certain distance d will get the rumor in k steps will be derived, as a function of the minimum distance d. For a selection of nodes, these theoretical probabilities will be compared against probabilities obtained through simulations, in order to find an estimate for the distance and hence to be able to determine based on this, which node is the source of rumors.

a. Notations

A list of commonly used notations is presented below:

1. N is the number of nodes in the network;
2. k is the number of discrete time steps between initial infection at the source node, to the time when we take the measurement at another node;
3. K is the total number of time steps;
4. L is the total number of experiments;
5. d is he minimum distance between the source and a certain node;
6. U_i is a matrix where element $U_i(k,j)$ is the measured probability that a node j gets the rumor after k time steps, at each experiment i ;
7. V is a matrix where element $V(k,j)$ is the measured average probability that a node j gets the rumor after k time steps, over the number of experiments L ;
8. P is a matrix where element $P(k,j)$ is the predicted probability that a node j gets the rumor after k time steps;
9. $q_{d,k}$ is the estimated probability of a node located at distance d to get the rumor for the first time after k steps; Compared to the elements of P, this carries additional information regarding the distance between the sensor and the source node;
10. $Q_d(k)$ is the estimated probability of a node located at distance d to have the rumor at time step k;
11. μ is the probability of spreading of rumors within the network, i.e. the probability to transmit the rumor between any two connected nodes at each step;
12. α is the probability for the rumor to be passed from a node at distance d to a node at distance $d+1$, β is the probability for the rumor to be passed from a node at distance d to a node at distance d, and γ is the probability for the rumor to be passed from a node at distance d to a node at distance $d-1$;

b. Overview

As an overview, this approach aims to find an analytical model for the probability $P(k,j)$ as a function of the distance d, i.e. the shortest path between node j and the source. Using this formula and the measurements obtained from the sensor nodes at different points in time, an estimate of the distances d_j will be obtained, for some sensor node j, which will thus be used to accurately detect the source.

The approach begins with an initial re-structuring of the network of nodes. In this sense, the nodes will be rearranged according to their minimum distance from the source.

The approach then models the spreading of rumors as a random walk in $1D$ and tries to find a formula for the probability that a node located at a minimum distance d from the source will get the rumor in k steps. This formula should resemble that of a diffusion field. In addition, the histogram of this probability over different time steps k, and for different values of distance d should have a similar behaviour to the plot of measured probability when simulating a spreading of rumors in the network. In other words, the plot of $P(k,j)$ over k should be the same as the plot of $V(k,j)$ over k. If this is the case, we can then use measurements of the probability at a random node in the network, which will enable us to find the minimum distance between this and the source. By applying this at several points in the network, we could localize the source using the trilateration process.

The values in the matrix V are derived as follows. At each step, the rumor is allowed to spread within the network. The values in the matrix U are updated by determining which nodes in the network currently have

the rumor information. After a set number of K time steps, the experiment stops and matrix U is in its final form. The experiment is repeated for a number of times equal to L experiments, and the matrices U_i are obtained, with $i = 1,2 \dots L$. These experiments can be seen as simulating the spreading of new rumors within the network starting from the same source. The average of the elements of these matrices is taken over the number of experiments L, in order to obtain V.

One of the most challenging and interesting aspects of this approach is represented by the modelling of the spreading of rumor as a random walk. In this sense, the nodes will firstly be re-arranged according to their distance to the source. Hence, if a node is located at distance d from the source, it will be positioned on level d, if the distance is d+1, then the node will be on level d+1 etc. At each time step, if a node on level d has the rumor, it can either pass it to a node on level d, to a node on level $d + 1$, or to a node on level $d - 1$, and this is repeated for a number of steps until the rumor reaches our sensor node. As we are interested in calculating the probability that the rumor reaches from the source to the sensor node for the first time after k steps, we would like to calculate the total number of paths between the source and the sensor node, and the probability of the rumor taking each of these paths. This will lead to a formula for the estimated probability $q_{d,k}$ of a node located at distance d to get the rumor for the first time after k steps. In order to calculate the probability of a node to have the rumor after k steps, we need to sum all the probabilities of the node getting the rumor in $1,2, \dots, k$ steps. Hence, the probability of a node to have the rumor after k steps is $Q_{d,k} = \sum_{j=1}^{j=k} q_{d,j}$.

We will assume that the probability of spreading is constant, denoted by μ. This probability could be used to derive the probabilities of the rumor to spread from a node on level d to a node on level d (α), to a node on level $d + 1$ (β) , and respectively to a node on level $d - 1$ (γ). In the case of a non-self-avoiding walk, the following assumption will be made, as explained below: $\beta \gg \alpha$ and $\gamma \cong 0$, whilst the mathematical derivations related to this approach are presented in the *Mathematical Formulation* section below.

c. Assumptions

To begin with, we assume that the network topology is known and that the network does not evolve in time, which ensures that all the connections between the nodes remain fixed and that no nodes are removed or added to the network throughout the duration of the observation. In addition, we assume the susceptible-infected spreading model, where any infected node can spread the rumor to any of its neighbours (including the ones which already have the rumor), and once a node holds the rumor information it cannot recover from it. The rumor spreading will therefore be modelled by a non-self-avoiding random walk. This assumption is justified by a real-world application of the problem, that of spreading of rumors in a social network, where it is possible that a person hears the same information from more than one other person.

Secondly, the approach assumes that the spreading of rumors happens with a constant probability throughout the entire network, which means that each node can spread the rumors to any of its neighbours with a fixed probability of spreading.

Furthermore, another assumption made is the fact that a node located at a certain distance from the source will get the rumor for the first time after a time interval approximately equal to the shortest path between the sensor and the source.

We will further justify these assumptions through numerical experiments and mathematical derivations, as described in the sub-section called *Mathematical Formulation* below.

d. Related Literature Research

The relevant literature research for the derivations used in this approach includes: theory of Markov Chains and Random Walks [17], Stirling formula, Taylor approximation, dynamics of trees, random geometric graphs, scale-free and small-world networks.

e. Motivation for Approach

The decision to re-structure the network based on the minimum distance between the source and all the other network nodes is useful in order to give a notion of location to the nodes, which is one of the most challenging aspects of the problem. This could help us apply the concept of random walk in $1D$ to the derivation of the probability that a node located at a minimum distance d from the source will get the rumor in k steps. Moreover, this also provides an advantage concerning the better visualization of the gradient of the diffusion field, when plotting the values of $V(k, d)$, i.e. the intensity of the rumor information at all the nodes in the network.

f. Advantages and Disadvantages

The main advantage of this approach is the fact that it can be applied to any of the considered network topologies, as long as the probability of spreading of rumors is known. Moreover, the derivations do not require any knowledge regarding the vertex degree, which means they can be applied to real-world networks, which are inhomogeneous in degree.

The drawbacks would be the following. Firstly, the performance of the Dijkstra algorithm, which is required in the source estimation algorithm using the below derived theoretical probabilities of rumor infection. This algorithm calculates the minimum distance between any two nodes in the network. The time complexity of the Dijkstra algorithm is $\mathrm{O}(n^2)$ is very large, leading to increased computational complexity in calculating the minimum distances in a large network.

In addition, another disadvantage of this derivation is represented by the constant *connectivity index* used in the theoretic probability formula (see Mathematical Formulation section below). While it would be preferable to have a more exact formula for the theoretic probability (avoiding constant terms), this parameter gives a degree of freedom, allowing the algorithm to find the optimal probability formula for the particular network topology and network parameters given. Moreover, as seen in the results in the Evaluation section, by choosing an optimal connectivity index, the theoretic probability becomes a very good approximation of the average simulated probability.

g. Mathematical Formulation

The aim of this derivation is to find a mathematical expression for the probability $Q_d(k)$, that a node at distance d, will have the rumor after k time steps. Let $q_{d,k}$ be the estimated probability of a node located at distance d to get the rumor for the first time after k steps.

Furthermore, if a node has the rumor at time step k, then it could have gotten this rumor for the first time, at time step $k, k-1, k-2 \dots d$, since the rumor cannot reach a node at minimum distance d in less than d steps, Hence, we could write $Q_d(k)$ as: $Q_d(k) = \sum_{t=d}^{k} q_{d,t}$

Next, we would like to find an analytical form for $q_{d,k}$, by looking at the diffusion of the rumor as a random walk process. The figure below illustrates the concept of a random walk, where the nodes in the network are represented by green dots and arranged according to their minimum distance to the source, located at the origin. Two possible random walks are shown in blue and orange, as paths of $k = 7$ steps between the source and a node located at distance $d = 5$ from the source.

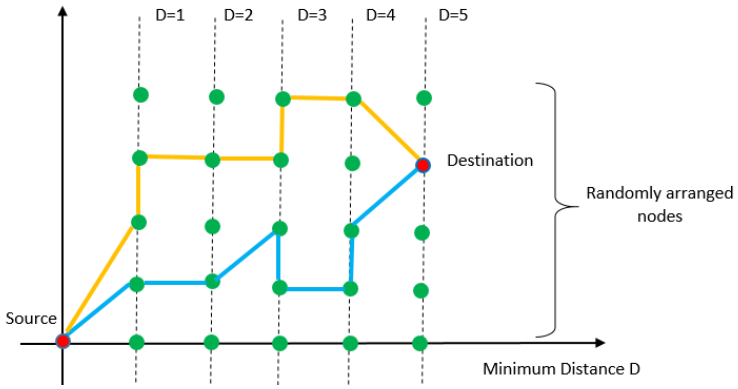


Figure 2: Illustration of possible Paths of Random Walk of Rumor in the Network

As illustrated above, the rumor can reach the destination by following a wide range of paths, each of total length k. Hence, there are many ways in which the rumor can reach the destination for the first time, exactly after k time steps.

Let there be N_{PATHS} possible ways in which the rumor can get to the destination from the source, in exactly k steps, and let the set $\{S1, S2, S3 \dots\}$ be the set of all possible paths.

Hence, since we can assume that any two paths are independent and since all the paths can happen with equal probability, the probability that the destination node located at distance d gets the rumor for the first time after k steps is:

$$q_{d,k} = p(S1 \cup S2 \cup S3 \dots) = p(S1) + p(S2) + p(S3) \dots = \sum_{i=1}^{N_{PATHS}} p(S_i) = N_{PATHS} \times p(S)$$

Number of Paths

As it can be seen above, the probability that the destination node gets the rumor for the first time is the sum of the probability of each of the N_{PATH} paths being the one followed by the rumor.

In order to calculate the number of possible paths N_{PATHS}, we need to find the number of all possible random walks in $1D$, each containing k segments. When the walk starts from the source, we can write:

$S = \varepsilon_1 + \varepsilon_2 + \cdots + \varepsilon_k$, where ε_i are independent and identically distributed random variables corresponding to a segment the rumor follows at each time step. This can take two values according to the type of segment the rumor follows. If the rumor advances in the network to a node further away from the source and closer to the destination node (any of the blue segments in figure below), then the corresponding $\varepsilon_i = 1$. Else, if the rumor will not advance in the network, and will move to a node at the same distance from the source (yellow segments in figure below), then the corresponding $\varepsilon_i = 0$. Else, if the rumor will follow any other segment (one of the orange segments in figure below), then the corresponding $\varepsilon_i = -1$.

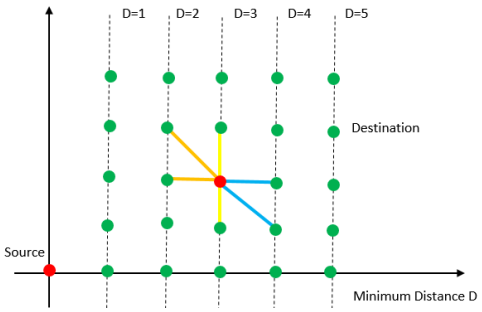


Figure 3: Possible Paths the Rumor can follow starting from one Node in the Network

Let us denote by $\#A$ the total number of blue segments in the path, by $\#B$ the number of yellow segments in the path and by $\#C$ the total number of orange segments in the path. Then, assuming the random walk is non-self-avoiding, the following equations hold:

(1) $\#A - \#C = d$, where d is the distance from the source to the destination.
(2) $\#A + \#B + \#C = k$, where k is the total number of time steps.

In addition, a large number of experiments have shown that generally, the probability of a node at distance d has the rumor increases significantly after a very small number of time steps, typically, for $k \geq d + 2$. From the above results, one approximation that could be made is that $\#C \cong 0$, which means that $\#A \cong d$ and $\#A + \#B = k$. This approximation could also be motivated by the method presented in the paper **"A fast Monte Carlo algorithm for source localization on graphs"** and described in the *State-of-the-Art* section above. In this paper, the authors consider the following two random processes: the infection process X and the alternate representation of the process X, denoted by Y. [12]

The process $Y = (y(0), y(1), \ldots)$ is defined as follows:

$$y_i(t) = \begin{cases} 0, if\ t < d(s,i) \\ 1, otherwise \end{cases}$$

where node s is the random source node initiating the rumor, $d(s,i)$ is the shortest path between the source and node i.

Hence, the process Y contains information about the times at which vertex i first becomes infected, and hence contains all the information of the process X. The authors prove the fact that $P\{y_i(t+1) = 1 | y(t)\} = P\{x_i(t+1) = 1 | x(t)\}$ and hence $Y = X$. This allows the sampling of the first time indices at which each vertex receives the rumor, $(\tau_1, \tau_2, \ldots \tau_N)$, by the shortest path between the source node and the corresponding monitor node.

This further justifies the assumption that the number of backward paths $\#C \cong 0$ and hence $\#A \cong d$ and $\#A + \#B = k$, with $\#B$ very small.

Hence, the total number of paths will be $N_{PATHS} = \binom{Path\ length - 1}{\#A - 1} = \binom{k-1}{d-1}$. This is because the first segment followed by the rumor right after it is initiated by the source will always be a forward segment (blue).

This represents the total number of possible ways in which we can choose which of the k segments in the path are of blue type. A different way to see it is to call $T = t\{i | i = 1, \ldots k\}$ the set of time indices, and to find all the possible ways in which we can form a subset of T, of size d, which contains the indices at which the rumor will follow a forward (blue) segment.

In order to find the probability as a function of distance, an approximation of the above formula should be used. This is derived by applying the Stirling approximation, followed by Taylor's approximation. A general case of the derivation is given by $\left(\frac{n}{\frac{n-m}{2}}\right) \cong \frac{2^{n+1}}{\sqrt{2\pi n}} \times e^{(-\frac{m^2}{2n})}$, and a mathematical proof is given below:

Result

$$\left(\frac{n}{\frac{n-m}{2}}\right) \cong \frac{2^{n+1}}{\sqrt{2\pi n}} \times e^{(-\frac{m^2}{2n})}$$

Useful formulas

Stirling's Formula: $\ln(n!) \cong n \times ln(n) - n + \frac{1}{2} \times \ln(2\pi n)$

Taylor's Approximation: $\ln(1+x) \cong x - \frac{1}{2}x^2$, which holds for $|x| \ll 1$

The following steps show the derivation for an approximation of $\left(\frac{n}{\frac{n-m}{2}}\right) = \frac{n!}{\frac{(n-m)}{2}! \times \frac{(n+m)}{2}!}$

Step 1. Logarithm on both sides

$$ln\left(\frac{n}{\frac{n-m}{2}}\right) = \ln(n!) - \ln\left(\frac{n-m}{2}!\right) - \ln\left(\frac{n+m}{2}!\right)$$

Step 2. Stirling Approximation

$$ln\left(\frac{n}{\frac{n-m}{2}}\right) = \{n \times ln(n) - n + \frac{1}{2} \times \ln(2\pi n)\} - \{\frac{(n-m)}{2} \times ln\left(\frac{(n-m)}{2}\right) - \frac{(n-m)}{2} + \frac{1}{2} \times \ln\left(2\pi\frac{(n-m)}{2}\right)\} - \{\frac{(n+m)}{2} \times ln\left(\frac{(n+m)}{2}\right) - \frac{(n+m)}{2} + \frac{1}{2} \times \ln\left(2\pi\frac{(n+m)}{2}\right)\}$$

Since $ln\left(\frac{(n-m)}{2}\right) = -\ln(2) + \ln(n-m)$, then the formula becomes

$$ln\left(\frac{n}{\frac{n-m}{2}}\right) = n \times ln(n) + \frac{1}{2} \times \ln(2\pi n) + \frac{(n-m)}{2} \times ln(2) - \frac{(n-m)}{2} \times \ln(n-m) - \frac{1}{2} \times \ln\big(\pi(n-m)\big) + \frac{(n+m)}{2} \times ln(2) - \frac{(n+m)}{2}\ln(n+m) - \frac{1}{2} \times \ln\big(\pi(n+m)\big)$$

$$ln\left(\frac{n}{\frac{n-m}{2}}\right) = n \times ln(n) + \frac{1}{2} \times \ln(2\pi n) + n \times ln(2) - \frac{(n-m)}{2} \times \ln(n-m) - \frac{1}{2} \times \ln\big(\pi(n-m)\big) - \frac{(n+m)}{2}\ln(n+m) - \frac{1}{2} \times \ln\big(\pi(n+m)\big)$$

$$ln\left(\frac{n}{\frac{n-m}{2}}\right) = n \times ln(n) + \frac{1}{2} \times \ln(2\pi n) + n \times ln(2) - \frac{n}{2} \times (1-\frac{m}{n}) \times \ln\left(n(1-\frac{m}{n})\right) - \frac{1}{2} \times \ln\left(\pi\left(n \times (1-\frac{m}{n})\right)\right) - \frac{n}{2} \times (1+\frac{m}{n}) \times \ln\left(n \times (1+\frac{m}{n})\right) - \frac{1}{2} \times \ln\left(\pi n \times \left(1+\frac{m}{n}\right)\right)$$

Step 3. Taylor's Approximation

$$ln\left(\frac{n}{\frac{n-m}{2}}\right) = n \times ln(n) + \frac{1}{2} \times \ln(2\pi n) + n \times ln(2) - \frac{n}{2} \times \left(1 - \frac{m}{n}\right) \times \left[\ln(n) - \frac{m}{n} - \frac{m^2}{2n^2}\right] - \frac{1}{2}$$
$$\times (\ln(\pi n) - \frac{m}{n}) - \frac{n}{2} \times (1 + \frac{m}{n}) \times \left[\ln(n) + \frac{m}{n} - \frac{m^2}{2n^2}\right] - \frac{1}{2} \times (\ln(\pi n) - \frac{m}{n})$$

$$ln\left(\frac{n}{\frac{n-m}{2}}\right) = n \times ln(n) + \frac{1}{2} \times \ln(2\pi n) + n \times ln(2) - n \times \left(\ln(n) - \frac{m^2}{2n^2}\right) - \frac{m^2}{n} - \ln(\pi n)$$

$$ln\left(\frac{n}{\frac{n-m}{2}}\right) = \frac{1}{2} \times \ln(2\pi n) + n \times ln(2) + \frac{m^2}{2n} - \frac{m^2}{n} - \ln(\pi n)$$

$$ln\left(\frac{n}{\frac{n-m}{2}}\right) = \frac{1}{2} \times \ln(2\pi n) + n \times ln(2) - \frac{m^2}{2n} - \ln(\pi n) = \ln(\frac{\sqrt{2\pi n}}{\pi n} \times 2^n) - \frac{m^2}{2n}$$

$$ln\left(\frac{n}{\frac{n-m}{2}}\right) = \ln(\frac{2^{n+1}}{\sqrt{2\pi n}}) - \frac{m^2}{2n}$$

$$\left(\frac{n}{\frac{n-m}{2}}\right) = \frac{2^{n+1}}{\sqrt{2\pi n}} \times e^{(-\frac{m^2}{2n})}$$

It is important to note that due to Taylor's Approximation, this expression holds for $|\frac{m}{n}|$<<1

Hence, when calculating the number of paths for our problem, we need the following replacement of variables:

1. n by $k-1$
2. $\frac{n-m}{2}$ by $d-1$, from where $m = k-1-2(d-1) = k-2d+1$
3. The approximation conditions are $|\frac{m}{n}|$<<1, equivalent to $|\frac{k-2d+1}{k-1}|$<<1, i.e. $k-2d+2 \ll k$. Since $k \geq d$, the expression holds for $k \cong d$. This condition will be proved in the following section.

Therefore, the number of paths becomes:

$$N_{PATHS} = \binom{k-1}{d-1} = \frac{2^k}{\sqrt{2\pi(k-1)}} \times e^{(-\frac{(k-2d+1)^2}{2(k-1)})}, \text{ which holds for } k \cong d.$$

Next, we need to find the probability of each path. We will denote by $p_A = P(\varepsilon_i = 1)$, i.e. the probability that at least one of the blue segments is followed at step i, and by $p_B = P(\varepsilon_i = 0)$, i.e. the probability that an yellow segment is followed by the rumor at step i. We assume that once the rumor will either follow a blue or yellow segments and hence the following must hold: $p_A + p_B = 1$.

Hence, the probability that a rumor will reach a node located at distance d in k steps becomes:

$$q_{d,k} = p(S) \times N_{PATHS} = {p_A}^d \times (1-p_A)^{k-d} \times \frac{2^k}{\sqrt{2\pi(k-1)}} \times e^{(-\frac{(k-2d+1)^2}{2(k-1)})}$$

Probability of Advancing through Network

In order to find an analytical form for p_A, we need to determine the probability that the rumor will take any of the blue steps, such that it advances in the network, getting closer to the destination node. The probability p_A should depend on the probability of spreading of rumors (denoted by μ), as well as on the network characteristics which will be modelled through the constant factor κ, which we will denote by connectivity index. As already mentioned in the mathematical derivation above, and using results obtained through simulations on networks of various topologies and size, it can be shown that the probability of infection of a node at distance d from the source becomes significantly large after the time step $k = d + 1$. In other words, we could make the assumption that $\kappa \cong d$, and therefore the number of backward paths $\#C \cong 0$, which means $\#A \cong d$ and $\#A + \#B = k$, with $\#B$ very small. In order to model the fact that the number of paths $\#B$ is very small, we can assume that p_A is larger than p_B.Therefore, as an initial approximation, p_A can be assumed to be linearly proportional to μ and the connectivity index κ>=1, which means $p_A \sim \mu \times \kappa$.

Hence the probability of a node located at distance d to receive the rumor for the first time at step k is:

$$q_{d,k} \approx (\mu \times \kappa)^d \times (1 - \mu \times \kappa)^{k-d} \times \frac{2^k}{\sqrt{2\pi(k-1)}} \times e^{\left(-\frac{(k-2d+1)^2}{2(k-1)}\right)}$$

Consequently, the probability that a node will have the rumor at time step k is equal to:

$$Q_d(k) = \sum_{t=d}^{k} q_{d,t} \approx \sum_{t=d}^{k} (\mu \times \kappa)^d \times (1 - \mu \times \kappa)^{t-d} \times \frac{2^t}{\sqrt{2\pi(t-1)}} \times e^{\left(-\frac{(t-2d+1)^2}{2(t-1)}\right)}$$

The above formula assumes that at each step each node passes the rumor to at least one other node, i.e. that at each time step there is a segment of either the type A (forward) or B (same level). Moreover, the probability expression does not take into account any multiplication of rumors, i.e. the fact that one node can spread the rumor to more of its neighbours at the same time. Although the rumor spreading model used for this project assumes that multiple rumors could be spread from a node at the same time step, we show through empirical results that the multiplication of paths does not have a significant impact on the probability of rumor infection and moreover, it could be accounted for using the *connectivity index.*

The subplots below show the probability of rumor spreading, obtained from a simulated of $R = 50$ rumors in a small-world network of $N = 200$ nodes, rewiring probability $\beta = 0.2$, and average vertex degree $V = 6$.The large value of average vertex degree aims to ensure a fair comparison of the below probabilities obtained through different spreading models. Under this conditions, the average number of neighbours a node would spread to, if it is allowed to spread to any number of neighbours with probability $P_s = 0.5$, will be larger than 1 (which represents the case when each node spreads the rumor to exactly one neighbour with a probability $P_s = 1$).

For the right subplot, the spreading model used is the one where each node spreads the rumor to any of its neighbours with a probability $P_s = 0.5$.

For the left subplot, the spreading model used is the one where each node spreads the rumor to exactly one neighbour with a probability $P_s = 1$. In addition, at each time step, the neighbour j to which each already infected node i spreads the rumor is chosen such that node j is further away from the source or at the same distance as node i. Moreover, the model assumes that each node can spread to a neighbour which is not already infected. Only if there are no neighbours which do not have the rumor yet, can the spreading occur to an already infected node.

Therefore, the assumptions of this spreading model agree to the assumptions made in the derivation of the theoretic probability formula above, and the equivalence between the two sets of assumptions is the following. A spreading probability of $P_s = 1$ means that at each time step the rumor is guaranteed to take a segment of either the type A (forward) or B (same level), as assumed in the above derivation. The fact that the neighbour j to which each already infected node i spreads the rumor is further away or at the same distance from the source as node i considers the assumption that $k \cong d$, and therefore the number of backward paths $\#C \cong 0$, which means $\#A \cong d$ and $\#A + \#B = k$, with $\#B$ very small. Finally, the assumption that each node can spread to a neighbour which is not already infected is made in order to account for the fact that $p_A > p_B$ as assumed in the derivation above, i.e. that the rumor is more likely to progress to new further away nodes, then to be stagnant.

For the middle subplot, the spreading model is similar to the one above. The only difference in this case is that each node i can spread the rumor to exactly two neighbours, if these exist and are both further away from the source compared to node i. Moreover, it can spread the rumor to exactly one neighbour, if this is located at the same distance from the source as node i. As before, if none of its neighbour are further away or at most at the same distance from the source, node i can spread the rumor to exactly one of its neighbours. This model will better reflect the assumptions used in the derivation of the probability formula, where a constant connectivity index $k > 1$ is used to ensure $p_A = k \times \mu$ has a larger value, which means that the rumor is more likely to advance through the network to reach its destination in a number of steps $k \cong d$.

In the left subplot we can observe that for the initial time steps the probability has a slow rise, followed by a steep rise after a certain time delay. This is a result of the fact that at a short time after the rumor initiation, very few nodes have the rumor to be able to spread it further. By comparing this graph with the plot of the actual rumor spreading to any neighbours with probability $P_S = 0.5$ (right), we can notice that the probability values are lower in the former case, particularly at small time steps.

The middle subplot is a better approximation of the actual rumor spreading, reflecting a faster rise in the infection probability shortly after the rumor initiation. By comparison with the graph of the actual rumor spreading (right), we can see that this model represent a better approximation of the real rumor spreading, compared to the model described above.

In summary, the above experiments illustrate that the mathematical formulation for the theoretic probability is a good approximation of the rumor infection model, by showing how the connectivity index k used in $p_A = k \times \mu$ could account for rumor multiplication (where each node can spread to multiple neighbours). The connectivity index depends on the topology and characteristics of the network, and could be determined by simulating a spreading of rumors in the matrix and choosing κ_{OPT} which minimizes the error defined as $\epsilon = \sum_k \sum_d \|Q_d(k) - V_d(k)\|$, where $Q_d(k)$ is the estimated probability, and $V_d(k)$ is the probability obtained through simulation. This is further detailed in the *Implementation* section.

Last but not least, we notice that the probability of advancing through the network $p_A = k \times \mu$ is also proportional to the rumor spreading probability μ. This is in order to model the fact that a rumor which has a lower spreading probability will take more time to advance through the network until it reaches its destination node.

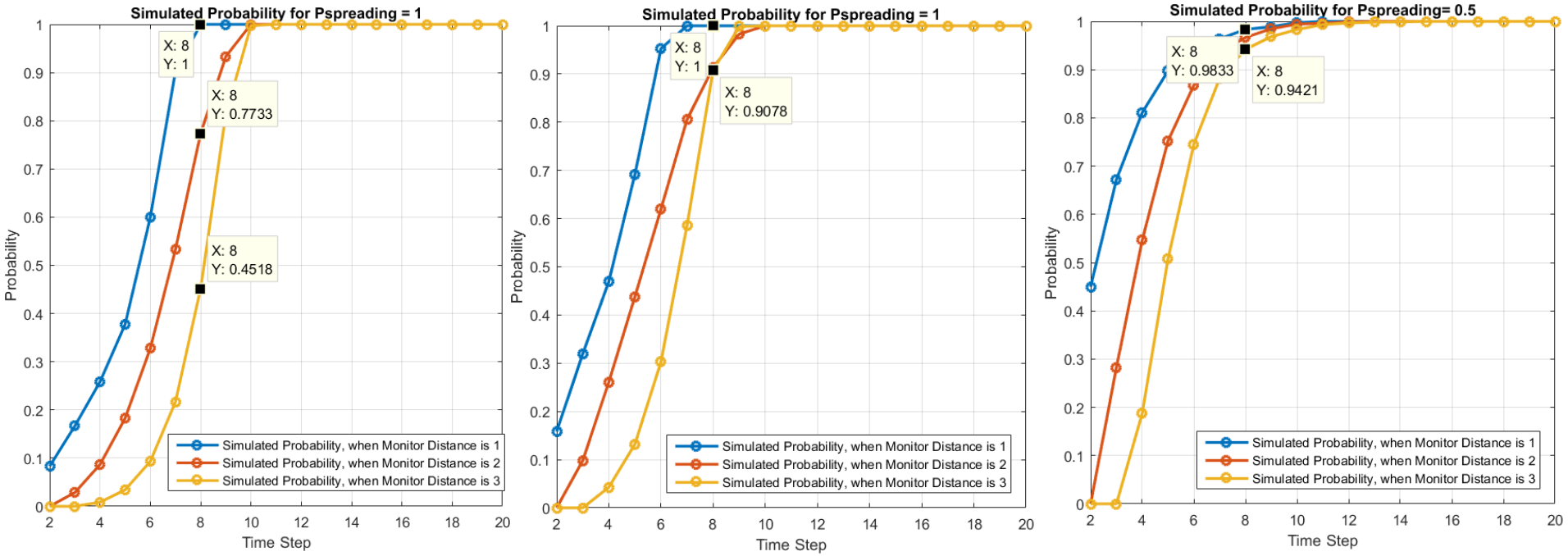


Figure 4: Simulated Rumor Probability for Two Different Rumor Spreading Models: Exactly 1 Neighbour (further or at the same distance from source) with Probability Ps=1 (left), Exactly 2 Neighbours (further from the source) and 1 Neighbour (at the same distance from source) with probability Ps=1 (middle), Any Neighbours with Probability Ps = 0.5 (right)

Theoretic Probability of Rumor Dissemination: A Robust Solution

As before, this approach models the spreading of rumors as a random walk process in $1D$, aiming to derive an analytical formulation for the probability of infection of a node located at a certain distance d, after k time steps.

a. Assumptions

The rumor spreading model used is the susceptible-infected model. The nodes can be in one of the two states, susceptible or infected, and once a node is infected it cannot recover. In addition, a node which already has the rumor, is able to receive it again from the same or different sources. Moreover, we assume that the probability of rumor spreading is constant throughout the network.

On the other hand, as opposed to the *Initial Solution*, no assumption will be made regarding the time of infection of a node, i.e. the time of infection k is not necessarily approximately equal to the shortest path between the sensor and the source, d. This should provide a more robust mathematical formulation for the theoretical probability of infection.

b. Related Literature Research

The relevant literature research for the derivations used in this approach includes: theory of Markov Chains and Random Walks [17], de Moivre-Laplace formula, properties of trees, random geometric graphs, scale-free and small-world networks.

c. Motivation for Approach

A refinement of the initial solution has as objective the derivation of a more accurate theoretical formula for the probability of rumor spreading.

d. Advantages and Disadvantages

The refined solution aims to give a more accurate approximation of the theoretic probability of infection. In this sense, the main advantage this solution provides consists in a more precise calculation of the number of paths the rumor could follow, with no restrictions on whether the rumor goes backwards or not (as opposed to the initial solution, which assumes that the number of backward paths $\#C$ is approximately zero). Consequently, this formula could be used to model a wider range of rumor spreading models.

The main disadvantage of this formulation is the dependence on the probabilities of advancing through the network: probability of forward path pA, probability of stationary path pB, and probability of downward path pC. These probabilities are dependent on the network topology and network characteristics, for example the average number of neighbours a node connects to in all the three directions considered (A, B, and C). Moreover, the calculation of these probabilities might be involved in a scale-free network (which best models the social network), as the node degree follows a power-law distribution, and using the average vertex degree

to compute $p_A, p_B, and\ p_C$ might not lead to accurate results in this case (as there are nodes with very large degree such as *hubs*, and nodes with very small degree).

Furthermore, another disadvantage is represented by the challenging simplification of the probabilities formula. While the initial solution provides a closed-form expression for the probability, the solution presented in this section is more complex and a closed-form solution will remain part of the requirements for future work.

e. Mathematical Formulation

The nodes will be arranged according to the distance from the source and a theoretical model will be derived for the probability of a node located at a distance d to get the rumor in k steps.

We assume that the rumor can take any of the following three paths, from any node in the network:

- $A-type$: the rumor is transmitted from a node at distance x to a node at distance $x+1$, where $x < d$.
- $B-type$: the rumor is transmitted from a node at distance x to a node at distance x, where $x \leq d$.
- $C-type$: the rumor is transmitted from a node at distance $x+1$ to a node at distance x, where $x < d$.

We will denote by $\{\tau_1, \tau_2, \tau_3, \dots \tau_k, \}$ the time steps from the start of the rumor, up to time k. At each time step, one of the above three possible paths will happen, and hence, we would be interested to count the number of possible sequences of paths that the rumor will follow in k time steps.
Ignoring the fact that a node can spread the rumor to multiple neighbors at the same time, and assuming that there are no restrictions on the succession of paths $A, B, or\ C$, the following two methods describe the calculation of the number of paths and the probability of the rumor reaching the node at distance d in k time steps.

Method I

The following two equations hold:

- (1) $A + B + C = k\ (time\ steps)$
- (2) $A - C = d\ (shortest\ distance)$

The number of paths is therefore:

$$N_{PATHS} = \sum_{\substack{A+B\leq k \\ A-C=d}} \binom{k}{A}\binom{k-A}{B} \xrightarrow{using\ (1)+(2)} N_{PATHS} = \sum_{d\leq A\leq k} \binom{k}{A}\binom{k-A}{k+d-2A}$$

We will assume that the probability of each of the three possible segments are the following: p_A, p_B and p_C for $A-type$, $B-type$ and respectively $C-type$.

Therefore, the probability of each path will be:

$$q_{d,k} = \sum_{d\leq A\leq k} \binom{k}{A}\binom{k-A}{k+d-2A} {p_A}^A {p_B}^{k+d-2A} {p_C}^{A-d}$$

Method II

The number of paths could also be calculated using the following approach. We consider the independent and identically distributed random variables $\{\varepsilon_1, \varepsilon_2, \varepsilon_3, \dots \varepsilon_k\}$, which represent the displacement of the rumor at each time step, and which can have the following values:

$$\varepsilon_i = \begin{cases} 1, if\ A - type\ displacement \\ 0, if\ B - type\ displacement \\ -1, if\ C - type\ displacement \end{cases}$$

Then the following equation holds, signifying that the distance taken by the rumor in k time steps is equal to d: $\sum_{i=1}^{k} \varepsilon_i = d$.

Moreover, the distribution of each of the random variables is given by:

$$f_{\xi_i}(\varepsilon_i) = \begin{cases} p_A, if\ \varepsilon_i = 1 \\ p_B, if\ \varepsilon_i = 0 \\ p_C, if\ \varepsilon_i = -1 \end{cases}$$

Taking into account the fact that the probability mass function of the sum of independent random variables is the convolution of their individual probability mass functions, the following holds:

$$P\left(\sum_{i=1}^{k} \varepsilon_i = d\right) = f_{\xi_1}(\varepsilon_1) * f_{\xi_2}(\varepsilon_2) \dots f_{\xi_k}(\varepsilon_k)$$

Evaluating the formula above, we are able to find the probability that a node located at distance d gets the rumor for the first time after k steps, i.e. $q_{d,k}$.

As a result, the probability of a node located at distance d to have the rumor at time k is:

$$Q_d(k) = \sum_{t=d}^{k} q_{d,t} = \sum_{t=d}^{k} \sum_{d \le A \le k} \binom{k}{A} \binom{k-A}{k+d-2A} {p_A}^A {p_B}^{k+d-2A} {p_C}^{A-d}$$

Alternatively, this could be written as:

$$Q_d(k) = \sum_{t=d}^{k} q_{d,t} = \sum_{t=d}^{k} P\left(\sum_{i=1}^{k} \varepsilon_i = d\right)$$

f. Refinement of the Number of Paths

Method I

The counting of the number of paths above assumes that the sequence of $A, B, or\ C - type$ segments in the rumor path is unconstrained and therefore, does not account for any illegal sequence. For example, if the rumor would be spread in the following sequence $\{A, C, C, C\}$, its x- coordinate would become -2, which is not possible, as all nodes are located at a positive distance from the source. Therefore, a refinement of the counting of the number of paths is required to account for such scenarios.

`

The method used is based on the reflection principles of a random walk in $1D$ [17].
The graph below illustrates a path the rumor could take and its reflection around the vertical line. The reflection starts at the point where the path touches the vertical line. Moreover, we should note that the first segment in the path (purple) is always an $A - type$ segment. We also note that reflection can only occur through the node which is the source of rumors, as this is the only node located on level $d = 0$.

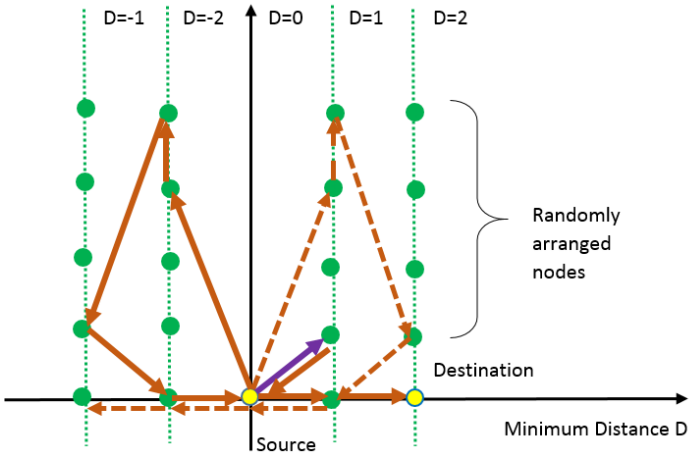


Figure 5: Reflection Principle of Random Walk in 1D

Let us assume that the destination node is located at distance $d = x$. As we see in the figure above, if the actual rumor path will reach the coordinate x, then the reflected path reaches the coordinate $-x$.

Therefore, in order to count the number of paths which cross the zero vertical axis and which have as a destination a node at distance $d = x$, after k time steps, we could instead count the number of reflected paths that reach the destination $d = -x$, after k time steps. We should note that the reflected paths we are interested in start at the origin in the positive direction and after a certain time delay, become negative and end up at a negative coordinate. In other words, this counts strictly the paths which will cross the zero-axis, and not the paths which start at the origin and remain in the negative left-hand plane throughout the entire duration of k time steps.

A further illustration and explanation of the reflection principle is given in [17].

As a result, the following holds:

$$\#\{(0,0) \to (k,d); start\ in\ positive\ direction, and\ touch\ zero\} =$$
$$= \#\{(0,0) \to (k,-d); start\ in\ positive\ direction\}$$

Consequently:

$$\#\{(0,0) \to (k,d); remain > 0\}$$
$$= \#\{(0,0) \to (k,d)\} - \#\{(0,0) \to (k,d); start\ in\ positive\ direction, and\ touch\ zero\}$$
$$= \#\{(0,0) \to (k,d)\} - \#\{(0,0) \to (k,-d); start\ in\ positive\ direction\}$$

As proven in the section above, the number of paths reaching distance d in k time steps is derived from $A + B + C = k$ and $A - C = d$, and is equal to:

$$\#\{(0,0) \to (k,d)\} = \sum_{d \le A \le k} \binom{k}{A} \binom{k-A}{k+d-2A}$$

For the calculation of the number of paths reaching distance $-d$ in k time steps we need to count the number of paths which start in the positive direction ($A-type$ segment) and which after a certain time step return back at the origin, continuing in the negative direction until distance $-d$. The following two equations hold:

- (1) $A+B+C=k$
- (2) $A-C=-d$

In addition, we assume that the time interval for which the path remains in the right half plane is equal to y steps, therefore for the remaining $k-y$ time steps the random walk will be negative. This is illustrated in the figure below.

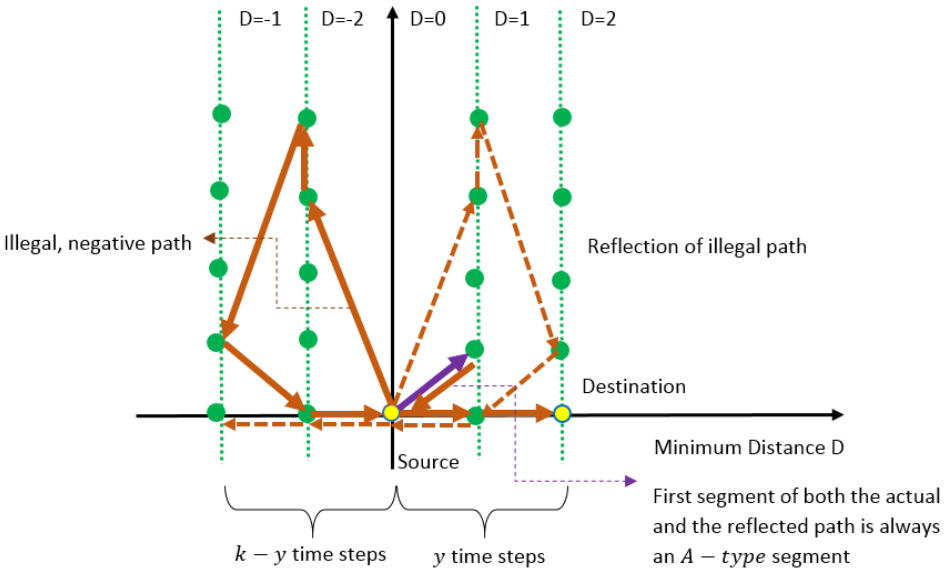


Figure 6: Illustration of Illegal Path and its Reflection, of k = 9 Time Steps, and Distance d = 2

Moreover, we denote the number of $A-type$ segments the path takes in the right half plane by A_P and the number of $A-type$ segments the path takes in the left half plane by A_N. Since the random walk returns to the origin after y steps, the number of $C-type$ segments the path takes in the right half plane must also be equal to A_P.

The number of possible paths the random walk takes in the right half plane before the return to the origin is denoted by N_P and is equal to:

$$N_P = \sum_{A_P=1}^{floor(\frac{y}{2})} \binom{y}{A_P}\binom{y-A_P}{A_P}$$

The formula above represents a counting of the possible ways of choosing a number of $A-type$ segments equal to A_P from a total of y segments available, and a number of $C-type$ segments equal to A_P, from the remaining $y-A_P$ segments available. The $B-type$ segments are fixed once we choose the former two segment types.

We note that the upper limit of the number of $A-type$ segments should be $\frac{y}{2}$ since the same number of $C-type$ segments must happen in order for the random walk to return to origin.

The number of possible paths the random walk takes in the left half plane is denoted by N_N and is equal to:

$$N_N = \sum_{A_N=1}^{floor(\frac{k-y-d}{2})} \binom{k-y}{A_N}\binom{k-y-A_N}{A_N+d}$$

The formula above represents a counting of the possible ways of choosing a number of $A-type$ segments equal to A_N from a total of $k-y$ segments available in the left half plane, and a number of $C-type$ segments, from the remaining $k-y-A_N$ segments available. The $B-type$ segments are fixed once we choose the former two segment types. The number of $C-type$ segments has been derived as follows:

- The total number of $A, B, C-type$ segments are denoted by A, B, and C respectively. As described above, we know that $A+B+C=k$ and $A-C=-d$.
- Moreover, we know that $A_P-C_P=0$ since the random walk must return to the origin before going to the left half plane. Therefore, $A_N-C_N=A-C-(A_P-C_P)=-d$. Hence, $C_N=A_N+d$.

In addition, we note that the upper limit of the number of $A-type$ segments, i.e. A_N is $\frac{k-y-d}{2}$. This is because $A_N+B_N+C_N=k-y$, from which we get that $A_N=k-y-C_N-B_N$ and hence, $A_N \leq k-y-C_N$. Since $C_N=A_N+d$, it follows that $A_N \leq k-y-A_N-d$, from which we get $A_N \leq \frac{k-y-d}{2}$.

Consequently, the total number of paths that start from the origin in the positive direction and which reach distance $-d$ is equal to:

$$\#\{(0,0)\to(k,-d)\}=\sum_{y=2}^{k-d}\Big[\sum_{A_P=1}^{floor(\frac{y}{2})}\binom{y}{A_P}\binom{y-A_P}{A_P}\sum_{A_N=0}^{floor(\frac{k-y-d}{2})}\binom{k-y}{A_N}\binom{k-y-A_N}{A_N+d}\Big]$$

In the formula above, we note that the upper limit of y is $k-d$. This is because there must be at least d segments taken in the left half plane in order to ensure we reach a node at distance $-d$. Moreover, we also note that for the same reason, the upper limit of the $A-type$ segments in the left half plane, A_N must be smaller than $k-y-d$. In addition, the lower bound of y is 2, since for any $A-type$ segment, there will be a corresponding $C-type$ in order to ensure the return to origin in y steps, and since the first segment in the path is a forward $A-type$, then there must at least one other $C-type$ segment, which leads to $y \geq 2$.

Furthermore, by comparing the actual and the reflected paths, we notice that the following equations hold:

- $C_{REFLECTED}=A_{ACTUAL}=A$. Hence $C_P+C_N=A$, which means $A_P+A_N+d=A$.
- $A_{REFLECTED}=C_{ACTUAL}=A-d$. Hence $A_P+A_N=A-d$, which is equivalent to the condition above.

This observation imposes a restriction on the number of $A-type$ paths that take place in the left half plane. In this sense, $A_N=A-d-A_P$. Therefore, the value of A_N will not range over the interval $\left[0,\frac{k-y-d}{2}\right]$, but will be instead fixed. As a result, the total number of paths that start from the origin in the positive direction and which reach distance $-d$ is equal to:

$$\#\{(0,0)\to(k,-d)\}=\sum_{y=2}^{k-d}\Big[\sum_{A_P=1}^{floor\left(\frac{y}{2}\right)}\binom{y}{A_P}\binom{y-A_P}{A_P}\binom{k-y}{A-d-A_P}\binom{k-y-(A-d-A_P)}{A-A_P}\Big]$$

Last but not least, we should note that the random walk must always have an $A-type$ segment as a starting point. Therefore,

$$\#\{(0,0)\to(k,d)\}=\sum_{d-1\leq A\leq k-1}\binom{k-1}{A-1}\binom{k-A}{k+d-2A+1}$$

$$\#\{(0,0)\to(k,-d);start\ in\ positive\ direction\}$$
$$=\sum_{y=2}^{k-d}\Big[\sum_{A_P=1}^{floor(\frac{y}{2})}\binom{y-1}{A_P-1}\binom{y-A_P}{A_P-1}\binom{k-y}{A-d-A_P}\binom{k-y-(A-d-A_P)}{A-A_P}\Big]$$

As a result, the total number of paths of a random walk that starts at the origin and reach distance d in k time steps is:

$$\begin{aligned} N_{PATHS} &= \#\{(0,0) \to (k,d); remain > 0\} \\ &= \#\{(0,0) \to (k,d)\} - \#\{(0,0) \to (k,-d); start\ in\ positive\ direction\} \\ &= \sum_{d \le A \le k} \binom{k-1}{A-1}\binom{k-A}{k+d-2A+1} \\ &\quad - \sum_{y=2}^{k-d}[\sum_{A_P=1}^{floor(\frac{y}{2})} \binom{y-1}{A_P-1}\binom{y-A_P}{A_P-1}\binom{k-y}{A-d-A_P}\binom{k-y-(A-d-A_P)}{A-A_P}] \end{aligned}$$

Consequently, the probability of a node at distance d to get the rumor after k steps is:

$$\begin{aligned} & q_{d,k} \\ &= \sum_{d \le A \le k} (\binom{k-1}{A-1}\binom{k-A}{k+d-2A+1} {p_A}^A {p_B}^{k+d-2A} {p_C}^{A-d} \\ &- \sum_{y=2}^{k-d}[\sum_{A_P=1}^{floor(\frac{y}{2})} \binom{y-1}{A_P-1}\binom{y-A_P}{A_P-1}\binom{k-y}{A-d-A_P}\binom{k-y-(A-d-A_P)}{A-A_P}] {p_A}^{A-d} {p_B}^{k-d-2(A-d)} {p_C}^A]) \end{aligned}$$

In the above formula, the exponents of the probabilities of advancing through the network, $p_A, p_B, and\ p_C$ are derived as follows:

- The total number of $A-type$ segments is equal to $A_P + A_N$. In Addition we know that $A_P + A_N = A - d$. This is the exponent of p_A.
- Therefore, $k - d - 2(A_P + A_N) = k - d - 2(A - d)$. This is the exponent of p_B.
- The number of $C-type$ segments is equal to $C_P + C_N = A$. This is the exponent of p_C.

Finally, the probability of a node at distance d to have the rumor after k time steps is:

$$\begin{aligned} & Q_d(k) \\ &= \sum_{t=d}^{k} q_{d,t} \\ &= \sum_{t=d}^{k}\{\sum_{d \le A \le k} (\binom{k-1}{A-1}\binom{k-A}{k+d-2A+1} {p_A}^A {p_B}^{k+d-2A} {p_C}^{A-d} \\ &- \sum_{y=2}^{k-d}[\sum_{A_P=1}^{floor(\frac{y}{2})} \binom{y-1}{A_P-1}\binom{y-A_P}{A_P-1}\binom{k-y}{A-d-A_P}\binom{k-y-(A-d-A_P)}{A-A_P}] {p_A}^{A-d} {p_B}^{k-d-2(A-d)} {p_C}^A])\} \end{aligned}$$

If we assume that the probabilities of advancing through the network are equal, then the above formula becomes:

$$Q_d(k) = \sum_{t=d}^{k} \left\{ \sum_{d\le A\le k} \binom{k-1}{A-1}\binom{k-A}{k+d-2A+1} - \sum_{y=2}^{k-d}\Big[\sum_{A_P=1}^{floor\left(\frac{y}{2}\right)} \binom{y-1}{A_P-1}\binom{y-A_P}{A_P-1}\binom{k-y}{A-d-A_P}\binom{k-y-(A-d-A_P)}{A-A_P} \right\} \times P^k, where\ P = \frac{1}{3}$$

Method II

Alternatively, a simplified way of calculating the reflected paths could be:

$$\#\{(0,0)\to(k,d)\} - \#\{(0,0)\to(k,-d); start\ in\ positive\ direction\} = \sum_{d\le A\le k} \binom{k-1}{A-1}\binom{k-A}{k-d-2A+1}$$

This formulation does not consider the constraints imposed in the previous derivation and simply assumes that $A+B+C=k$ and $A-C=-d$, where $A, B, and\ C$ are the number of segments of $A-type, B-type, and\ C-type$ respectively, in the reflected path.

Therefore, the probability of a node at distance d to have the rumor after k time steps is:

$$Q_d(k) = \sum_{t=d}^{k} q_{d,t} = \sum_{t=d}^{k}\sum_{d\le A\le k} \left\{ \binom{k-1}{A-1}\binom{k-A}{k+d-2A+1} - \binom{k-1}{A-1}\binom{k-A}{k-d-2A+1} \right\} {p_A}^A {p_B}^{k+d-2A} {p_C}^{A-d}$$

As before, if we assume that the probabilities of advancing through the network are equal, then the above formula becomes:

$$Q_d(k) = \sum_{t=d}^{k}\sum_{d\le A\le k} \left\{ \binom{k-1}{A-1}\binom{k-A}{k+d-2A+1} - \binom{k-1}{A-1}\binom{k-A}{k+d-2A+1} \right\} \times P^k, where\ P = \frac{1}{3}$$

In summary, the following two formulations have been derived, in order to model the probability of a node at distance d to have the rumor after k time steps:

1. $Q_d(k) = \sum_{t=d}^{k}\{\sum_{d\le A\le k}(\binom{k-1}{A-1}\binom{k-A}{k+d-2A+1} {p_A}^A {p_B}^{k+d-2A} {p_C}^{A-d} - \sum_{y=2}^{k-d}[\sum_{A_P=1}^{floor\left(\frac{y}{2}\right)} \binom{y-1}{A_P-1}\binom{y-A_P}{A_P-1}\binom{k-y}{A-d-A_P}\binom{k-y-(A-d-A_P)}{A-A_P}] {p_A}^{A-d} {p_B}^{k-d-2(A-d)} {p_C}^A])\}$
2. $Q_d(k) = \sum_{t=d}^{k} q_{d,t} = \sum_{t=d}^{k}\sum_{d\le A\le k}\{\binom{k-1}{A-1}\binom{k-A}{k+d-2A+1} - \sum_{d\le A\le k}\binom{k-1}{A-1}\binom{k-A}{k-d-2A+1}\} {p_A}^A {p_B}^{k+d-2A} {p_C}^{A-d}$

Both results above could be simplified using Stirling's approximation or de Moivre-Laplace formula.

Chapter 4

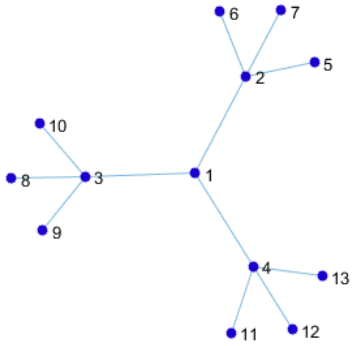
Implementation

Matlab Environment: Network Model

Construction of Tree Graph

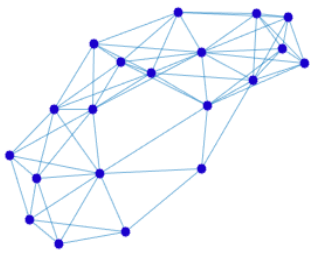
The tree is constructed according to the network size (number of nodes) and its depth (the maximum distance from the root node to children nodes). Indexing the nodes as shown in the figure below, we are able to find the indices of the children node for any given parent node, therefore being able to define the edges of the tree graph.

We assume that each node has C children and that the tree depth is D. Hence, the indices of the nodes located at depth d range between $lower_{bound} = \frac{C^{D-1}-1}{C-1} + 1$ and $upper_{bound} = \frac{C^{D}-1}{C-1}$. Therefore, for each of the nodes indexed between these values, the indices of its children will be $index_{child\ j} = lower_{bound} + C^{d-1} + \left(index_{parent} - lower_{bound}\right) \times C + j - 1$, where $j = 1,2 \dots C$.



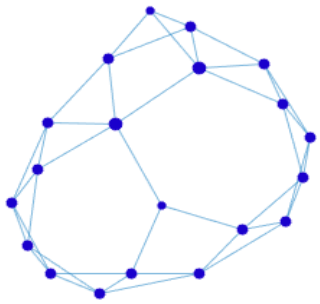
The figure on the left shows an example of a tree of depth D=3, with $C = 3$ children. At iteration 1, the depth is 1, and the only node is the root of the graph, node 1. At iteration 2, the depth is 2, and the indices of the added nodes range from $\frac{C^{D-1}-1}{C-1} + 1 = 2$ to $\frac{C^{D}-1}{C-1} = 4$. At iteration 3, the depth is 3, and the indices of the added nodes range from $\frac{C^{D-1}-1}{C-1} + 1 = 5$ to $\frac{C^{D}-1}{C-1} = 13$.

Construction of Random Geometric Graph



Firstly, N random $2D$ locations are generated in a square plane of a given dimension (e.g. $1x1$). Secondly, any two nodes will be connected through an edge if the distance between them is smaller than a given radius (e.g. $r = 0.2\ for\ 1x1\ grid$).

Construction of Small-World Graph



The network is created using the Watts-Strogatz algorithm, which can be summarized as follows. Firstly, each node is connected to its K next and previous neighbors (in terms of the nodes' indices). Hence, each node will have $2K$ neighbors. Then, the nodes are rewired with a given probability β. In this sense, for every node n_i and every edge (n_i, n_j), the edge is replaced by (n_i, n_k) with probability β, where k is a uniform random variables from all the nodes that do not have a connection with node n_i yet, and which avoid self-loops $(k \neq i)$. The Matlab codes can be found in Appendix E.

Construction of Scale-Free Network

Method I

A scale-free network is a network which has a power law degree distribution. In other words, the probability of a node having k connections to other nodes is given by $P(k) \sim k^{-\gamma}$, where γ is a parameter in the range $(2,3)$.

The network is created using the Barabási–Albert (BA) model. The algorithm begins with an initial network of N_0 connected nodes. A new node is added, by connecting it to $n \leq N_0$ existing nodes. The probability that the new node is connected to node i is given by:

$$P_i = \frac{k_i}{\sum_j k_j}$$

where k_i is the degree of node i and the range of the sum is the set of all pre-existing nodes.

It can be seen that high degree nodes , or hubs, will have a higher probability of connecting to a new node, hence accumulating even more links.

Method II

This method implements the algorithm described by the authors of the paper "Deterministic scale-free networks". The model is described below, generating a deterministic scale-free network, with power law degree distribution [20].

Step 1. We designate a single node as the root of the graph
Step 2. Two additional nodes are added, and each of them is connected only to the root defined above. Up to this point the adjacency matrix is M_2
Step 3. Two additional unit of two nodes each with the same structure as matrix M_2 are added, and only the *bottom* nodes of each of these units will be connected to the root. No additional connections are made.
Step *n*. The rule can be generalized as follows. At step n, two units identical to the matrix obtained at the previous step, M_{n-1} will be added to the network. The bottom 2^n nodes of these units will be connected to the root of the network, with no other additional connections being made.

The distribution of the node degree of the network generated using the method above is $(k) \sim k^{-\frac{ln\,3}{ln\,2}}$.
The difference between the two methods is illustrated in the figures below:

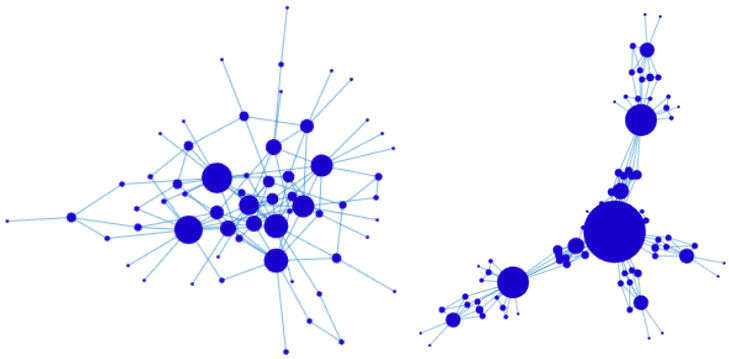


Figure 7: Implementation of a Scale-free Network using Method I (left) and Method II (right), for N=81 Nodes

Matlab Environment: Epidemic Model

The model we consider is the SI model, where each node can be in one of the two states, susceptible or infected. According to its state, a value is associated to each node, which is $v = 0$ is the node is susceptible and $v = 1$ if the node is infected. We will assume that initially, at time $t = 0$, a node is chosen uniformly at random to be the infectious source node, from which the spreading of rumors starts. After $t = 1$, the source node is able to transmit the rumor to any of its neighbours, with constant probability μ. At the next step, any of the infected nodes can transmit the infection signal to any of their already infected or susceptible neighbors (irrespective of whether the neighbors already has the rumor information), with the same probability μ. The process of rumor spreading is a Markov process, since the state vector $x(t+1)$ depends on the previous states only through $x(t)$. The rumor is allowed to spread for a fixed number of time steps, and the experiment is repeated for a set number of times, with the rumor initiated from the same source. Then, the mean over all experiments is taken, to find the average probability that a node will have the rumor at a certain time step. The code can be found in Appendix E.

The plots below show the evolution of several rumors in a random geometric graph of $N = 1000$ nodes, over 6 time steps. The coloring of each node reflects the average probability that it is infected with the rumor at each time step, specifically it shows the proportion of rumors heard at each node. As illustrated below, the spreading of the rumor within the network follows a diffusion-like process.

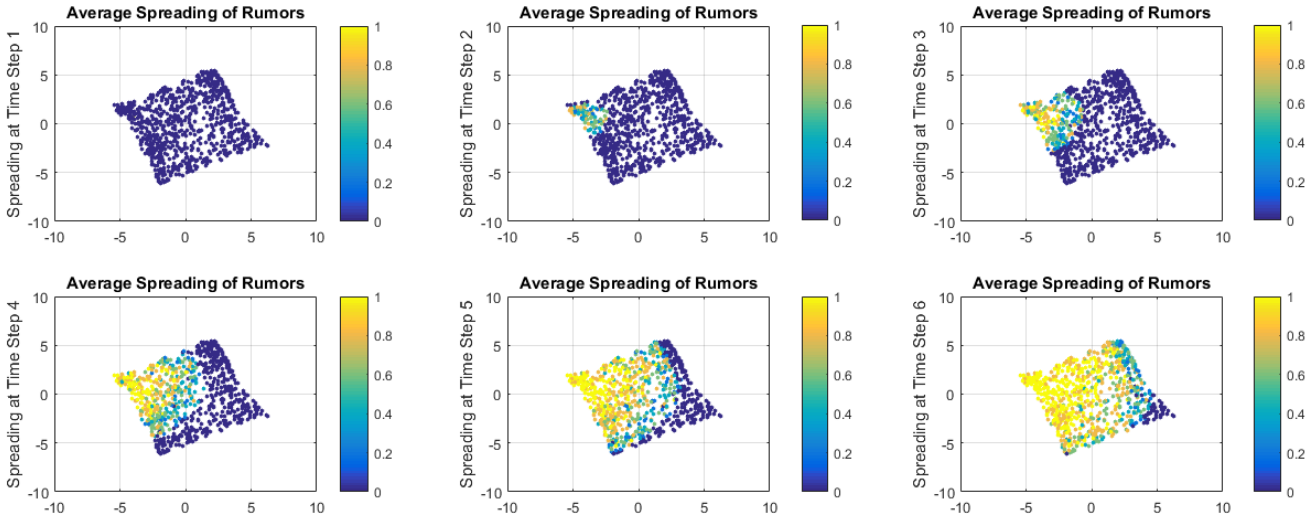


Figure 8: Average Spreading of Rumors in Random Geometric Graph of N=1000 nodes, at Different Time Steps

The plots below show the evolution of the rumor in a tree graph, of $No_{children} = 4$ and $Depth = 6$.

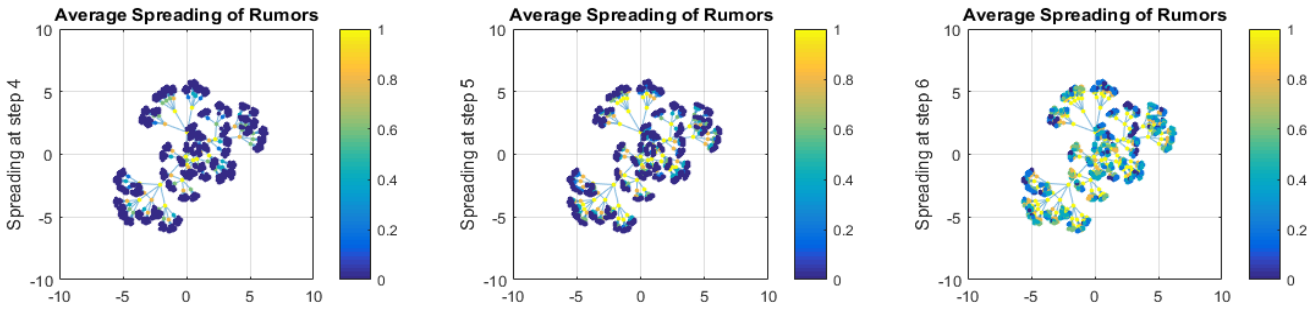


Figure 9: Average Spreading of Rumors in Tree Graph of N=1365 nodes, at Different Time Steps

While the figures above do not contain any information regarding the degree of the nodes, the plots below illustrate various rumor spreading experiments, showing how the spreading is influenced by the degree of the nodes infected. The rumor simulations have been made on the following network types: tree graph (with 3 children and depth 5), random geometric graph (with 200 nodes and connectivity radius 0.12), small-world network (with 200 nodes and rewiring probability 0.1), and scale-free network (with 81 nodes).

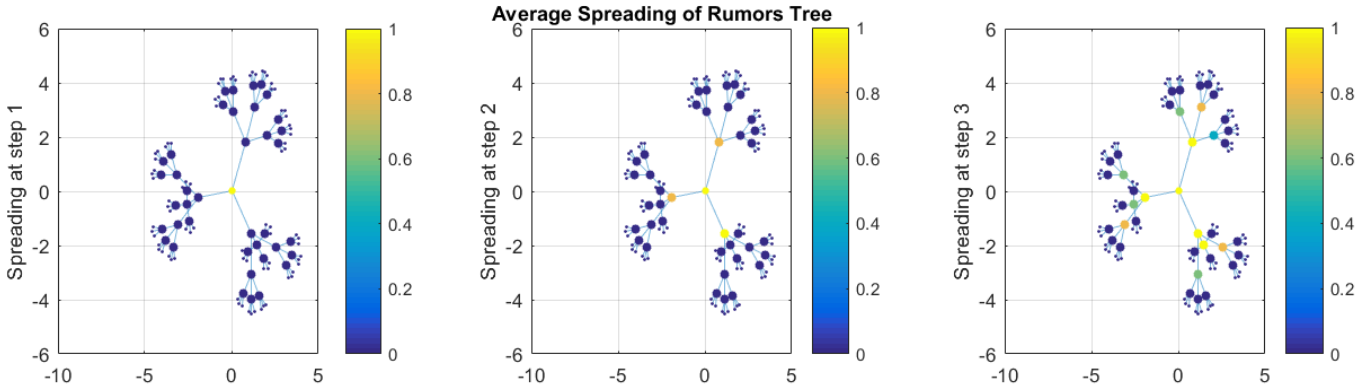


Figure 10: Simulation of Rumor Spreading in a Tree Graph for a Spreading Probability of Ps = 0.7

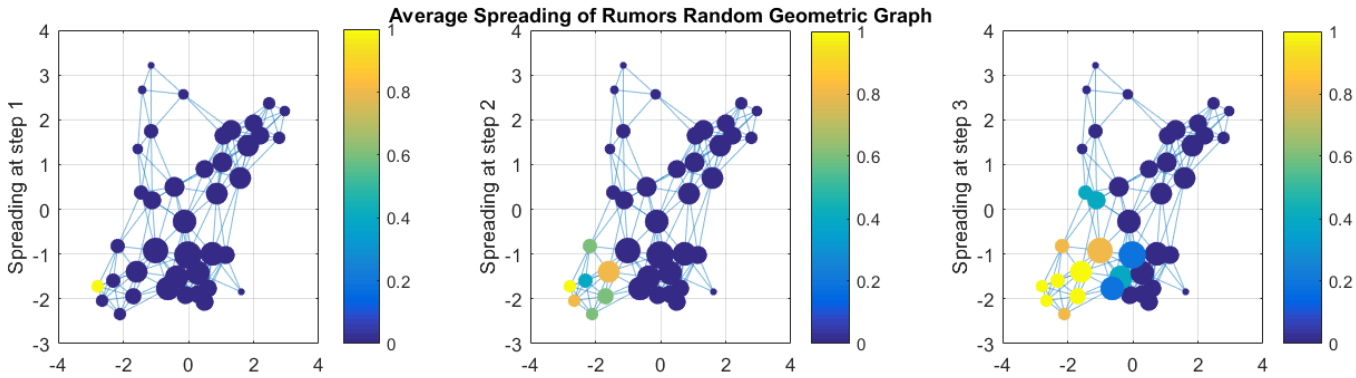


Figure 11: Simulation of Rumor Spreading in a Random Geometric Graph for a Spreading Probability of Ps = 0.7

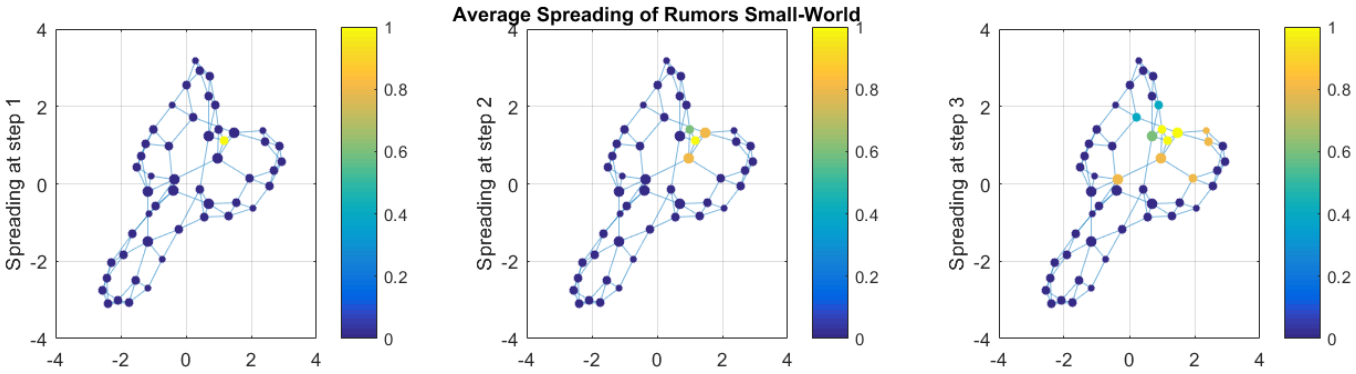


Figure 12: Simulation of Rumor Spreading in a Small World Network for a Spreading Probability of Ps = 0.7

The plot below shows a spreading of rumor initiated at time $k = 1$, up to time $k = 3$, illustrating how the *intensity* of rumor spreading increases around high-degree nodes, in a scale-free network.

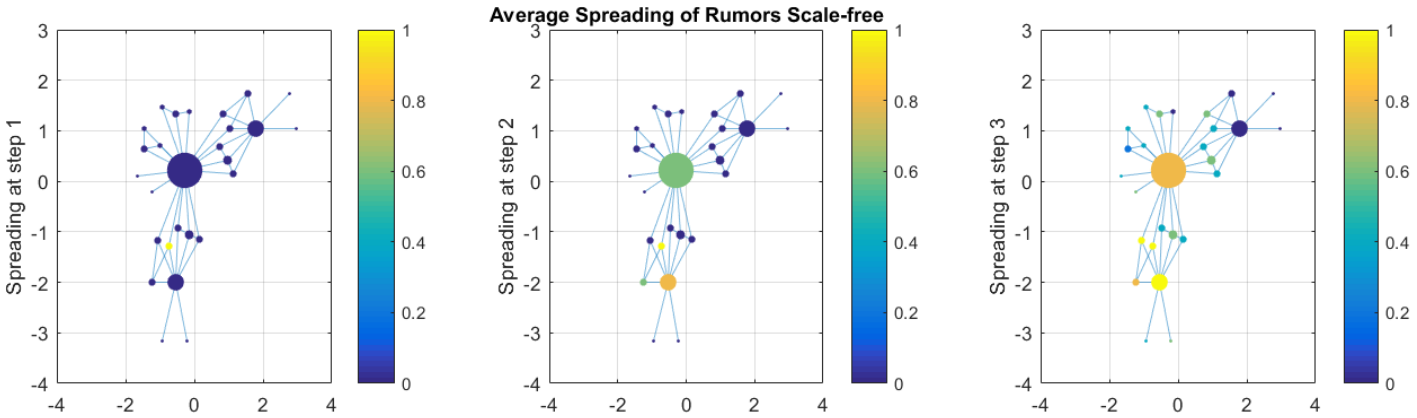


Figure 13: Simulation of Rumor Spreading in a Deterministic Scale-free Network for a Spreading Probability of Ps = 0.7

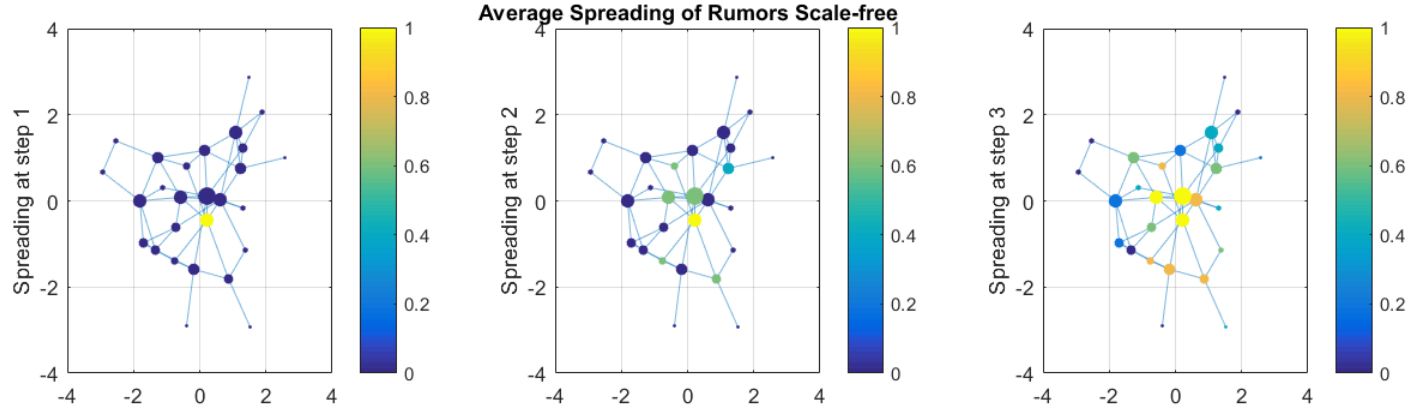


Figure 14: Simulation of Rumor Spreading in a Random Scale-free Network around a High-Degree Node, for a Spreading Probability of Ps = 0.5

The diffusion process of the rumor is further illustrated in the figure below, which rearranges the nodes according to the shortest distance from each node to the source. This also shows the motivation for the initial approach to solve the source detection problem, which is based on the minimum distances between the nodes in the graph.

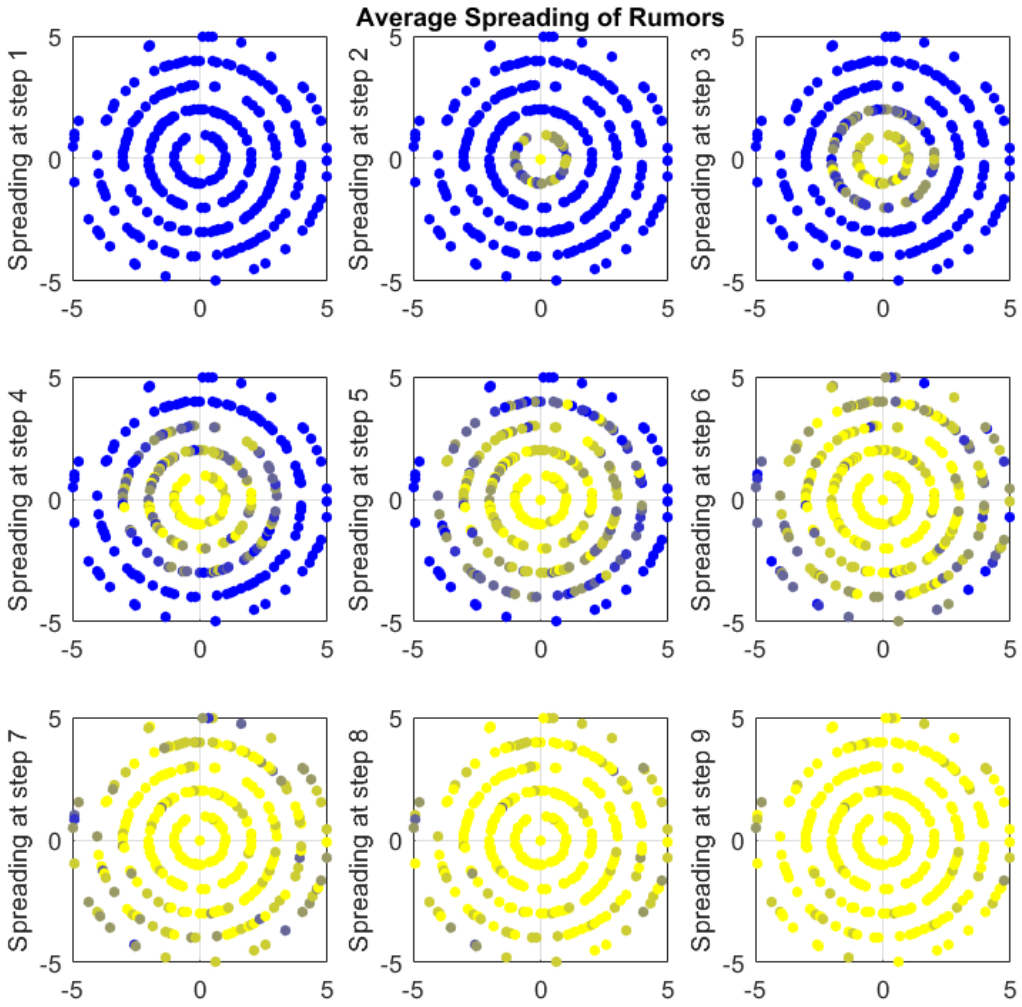


Figure 15: Average Spreading of Rumors in Random Geometric Graph of N=300 nodes, at Different Time Steps

Source Detection Algorithm

The source detection algorithm is described below, including the main steps of the algorithm and the motivation for the design choices presented. This consists of several enhancements which will be evaluated individually in the *Evaluation* Chapter of this report.

Algorithm 1: Estimation of a single rumor source using simplified probability formula

Initialisation**:**

1. Network topology:
    - Tree Graph;
    - Random Geometric Graph;
    - Small World Network;
    - Scale-free Network.
2. Network parameters:
    - Tree Graph: depth and number of children;
    - Random Geometric Graph: connectivity radius and grid dimension;
    - Small World Network: rewiring probability and average node degree;
    - Scale-free Network: deterministic or random generation model.
3. Number of nodes in the network N.
4. Susceptible-Infected Spreading model:
    - Spreading with a certain probability;
    - Spreading to exactly 1 neighbour in 1 time step;
    - Spreading to maximum x neighbours in 1 time step.
5. Probability of spreading P_S.
6. Number of rumors R available for the actual rumor spreading.
7. Number of rumors R_{avg} required to compute the average simulation probability used to derive the optimal parameters of the theoretic probability. Typically R_{avg} should have a large value.
8. Number of time steps K at which measurements are available after the rumor initiation.
   Note: This should be larger than the network radius.
9. Number of sensors N_s.
10. Maximum estimated distance between the sensor node and the candidate source $d_{estimated}^{MAX}$.
    This represents one of the criterion of ranking the sensor nodes, and the monitors for which the estimated distance is larger than the maximum set above will be discarded.
11. Time interval for considered measurements τ, the only sensor measurements considered are those occurring in the interval $[0, \tau]$.
12. Minimum cardinality of the set of estimated sources.
13. Cardinality of set of estimated sources which are selected using the rumor centrality method.

Estimation Algorithm:

Step 1. Generate a network of N nodes and specified characteristics.

Step 2. Calculate the shortest paths between any two nodes in the network using the *Dijkstra* algorithm.

Step 3. Designate the source node as the rumor starting point.

Step 4. Simulate a spreading of rumors and repeat the experiment for R times, starting from the same source node. Obtain the probability of the monitor nodes having the rumor at different time steps in the interval $[1, K]$. This is denoted by $V_{individual}$.

Step 5. Calculate an average measured probability of spreading $V_{average}$, as follows. Generate a spreading of rumors from the same source and repeat the experiment for a large number of iterations. For each experiment, calculate the average probability of a node located at a certain distance from the source. Average the results over all the experiments.

Step 6. Derive the optimal sensor maximum error ε_S.

Step 7. Derive the optimal connectivity index k as follows. The value of the connectivity index will be chosen in the interval $k \in [1,2]$ in order to minimise the mean-square error between the theoretic probability and the average simulated probability, calculated as a sum of the errors at each time step, and for each distance considered. For example, if for the average simulated probability we choose node i as the initiator of the rumor, and the furthest away node from the source node i is d, then the mean-square error is calculated as follows:
$\epsilon = \sum_d \sum_k (P_{theoretic} - V_{individual})^2$.
For the calculation of the mean-square error, we could consider all the time measurements available, or on the other hand, we could consider only the measurements at a time t in a neighbourhood of the distance d, since these measurements could be sufficient to ensure accurate results, while reducing the algorithm complexity.

Step 8. Using the optimal connectivity index, calculate the theoretical probability of spreading denoted by $P_{theoretic}$.

Step 9. Randomly select the monitor nodes N_s.

Step 10. For each monitor, estimate the shortest distance between the sensor and the source node, by minimising the mean-square error between the theoretic and the measured probability of rumor spreading, $\epsilon = \sum_k (P_{theoretic} - V_{individual})^2$, where the time window for which we consider measurements is $k\epsilon[\tau_1, \tau_2]$. The start time τ_1 of the window is the time at which the sensor measurement becomes positive for the first time. This is chosen in order to ensure that non-negative values in the theoretic probability before the time τ_1 do not impact the error calculation. We could consider the example below, where the actual sensor distance is $d = 4$. If the measurement at time $k = 4$ would not be considered, then the minimum error would be achieved using the theoretic probability values for distance $d = 4$ (error $\varepsilon_1 = 0.24$). On the other hand, using the measurement at time $k = 4$, the total error becomes $\varepsilon_1' = 0.34$, compared to $\varepsilon_2 = 0.25$.

Time Step	1	2	3	4	5	6
Individual Sensor Measurement, $d = 4$	0	0	0	0	0.2	0.4
Theoretic Probability Approximate Values, $d = 5$	0	0	0	0	0.05	0.35
Theoretic Probability Approximate Values, $d = 4$	0	0	0	0.10	0.3	0.52

Table 3: Illustration of the Simulated and Theoretic Spreading Probabilities for K=6 Steps

The end time of the window τ_2 is the moment at which the measurements stop. Alternatively, it can be set as $\tau_2 = \min(\tau_1 + c, K)$, where c is a constant and K is the number of time measurements available after the rumor initiation.

After the shortest distance has been estimated using the minimum-mean square error method, the following check will be performed in order to account for possible noise in the sensor measurements.

If the estimated distance is d_{est} and the sensor measurement at a time step smaller than d_{est} is positive, it means that the estimated distance should be replaced by the time step at which the measurement becomes positive for the first time. Generally, the shortest path error will be $1\ hop$ and hence, it is expected that the estimated distance can be smaller than the estimated one by at most 1.

Step 11. Assign a confidence level to each sensor node based on the criteria below:

a. ***Criterion 1***: The estimated distance should be at most $d_{estimated}^{MAX}$. The motivation for this approach is the fact that sensors for which the estimated shortest distance is small have more accurate measurements and a higher probability of correct estimation of this distance. Nevertheless, if the maximum allowed distance is too small, then we might not have enough sensors available and therefore, a correct detection of the source might be challenging to achieve.

b. ***Criterion 2***: The number of occurrences of *A* should be large, where *A* is defined as the event when the sensor measurement at time k corresponding to a monitor at estimated distance d should be larger than the theoretic probability at time k and distance $d+1$ and lower than the theoretic probability at time k and distance $d-1$. The motivation is that the conditions $V_{average}(k,d) < P_{theoretic}(k,d-1)$ and $V_{average}(k,d) > P_{theoretic}(k,d+1)$ hold if the simulated probability is averaged over a large number of rumors. Therefore, if these conditions are satisfied by the simulated probability, they could be a measure of the convergence of the individual measured probability to the average one, and hence to the theoretical one.

c. ***Criterion 3***: The error between the theoretical and sensor measured probabilities should be lower than the error between the theoretical and the average sensor measured probabilities, i.e. $\epsilon_1 = P_{theoretic} - V_{individual}$, $\epsilon_2 = P_{theoretic} - V_{average}$ and $\epsilon_1 \leq \epsilon_2$.
The motivation for this approach is the following. While the theoretic probability ($P_{theoretic}$) converges to the average simulated probability ($V_{average}$), when this is computed using a large number of rumors (e.g. 200), the individual measurements ($V_{individual}$) might deviate from the average value since they are derived using a small number of rumors (e.g. 10). In order to ensure a correct estimation of the distance, we need to discard the measurements which significantly deviate from the average probability.

d. ***Criterion 4***: The minimum mean-square error of monitor node i should be lower than the maximum error ε_S. This is done in order to account for the case of noise in the sensor measurements, where the deviation from the expected probability value is too large.
Different criteria will carry different weightings, from highest to lowest weighting as follows: criterion 1, criterion 2 criterion 3 criterion 4. For example, not satisfying criterion 1, even though criterion 2 is satisfied will be penalized more than not satisfying criterion 2, even when criterion 1 is satisfied. This is because the accuracy of measurements mostly depends on the estimated shortest path (criterion 1), and less on the conditions of criterion 2. In order to model this, the confidence level is calculated as follows, where *1* means the criterion is satisfied and *0* means it is not satisfied.

Criteria 1	Criterion 2	Criterion 3	Criterion 4	Confidence Level
1	1	1	1	$CL = d_{est}$
1	1	1	0	$CL = d_{est} + w_4$, with $w_4 = 100$
1	1	0	1	$CL = d_{est} + w_3$, with $w_3 = 200$
1	1	0	0	$CL = d_{est} + w_3 + w_4$
1	0	1	1	$CL = d_{est} + w_2$, with $w_2 = 300$
1	0	1	0	$CL = d_{est} + w_2 + w_4$
1	0	0	1	$CL = d_{est} + w_2 + w_3$
0	0	0	0	$CL = d_{est} + w_2 + w_3 + w_4$
0	1	1	1	$CL = d_{est} + w_1$, with $w_1 = 400$
0	1	1	0	$CL = d_{est} + w_1 + w_4$
0	1	0	1	$CL = d_{est} + w_1 + w_3$
0	1	0	0	$CL = d_{est} + w_1 + w_3 + w_4$
0	0	1	1	$CL = d_{est} + w_1 + w_2$
0	0	1	0	$CL = d_{est} + w_1 + w_2 + w_4$
0	0	0	1	$CL = d_{est} + w_1 + w_2 + w_3$
0	0	0	0	$CL = d_{est} + w_1 + w_2 + w_3 + w_4$

Table 4: Confidence Level Calculation

The sensor nodes will be ordered according to their confidence levels, from lowest to highest value.

Step 12. For each node in the ordered set of monitor nodes, eliminate all source nodes whose distance to the monitor node is not equal to the estimate distance calculated at Step 9.

There are three main strategies used to eliminate the sources, based on the measurements from the ranked sensors. Nevertheless, strategy 3 will be used in the final algorithm.

a. ***Strategy 1****:* For each sensor node i from the ordered set of sensors based on their confidence levels, we create a set of nodes that would have to be eliminated from the set of candidate sources, using the measurements from monitor i(i.e. based on the estimate shortest distance between node i and the sources). From this set, we randomly select one node to eliminate, up to the point where the cardinality of the set of candidate sources is equal to the minimum value set by the user. For example, suppose the cardinality of the set of sources is currently $C = 11$, while the minimum value is $C = 10$, and that using measurements from monitor i, we would have to eliminate nodes j and k from the set of potential sources. Since eliminating both these nodes would lead to $C = 9 < 10$, we will randomly choose one of the two nodes to eliminate, either j or k.

b. ***Strategy 2***: For each sensor node i, we compute how many nodes will be eliminated from the set of candidate sources, based on measurements from this sensor. As a result, we are able to determine the number of remaining potential sources, if we were to consider the measurements provided by monitor i. If this number is greater than the minimum cardinality set by the user, then consider the estimation based on measurements from node i. Otherwise, we discard these measurements and the set of potential sources remains as before.

For example, suppose the minimum cardinality of the set of potential sources is $C = 10$. There are currently $N_S = 15$ sources in this set. However, if we consider the measurements from node i, there would be another $N_S = 9$ sources eliminated, which would bring the set of candidate sources to $N_S = 6$. This is lower than $C = 10$ and hence, the measurements from node i will be disregarded and the number of potential sources remains at $N_S = 15$ sources. This is repeated for all the monitor nodes, by considering more confident nodes first.

This means that even if measurements from a very confident node cannot be considered as they would decrease the cardinality of the set of potential sources beyond the minimum limit, measurements from a less confident node could be considered.

c. ***Strategy 3***: In order to account for the errors that could occur as a result of using less confident sensors (as in Strategy 1 above), the following method can be used to create the set of candidate sources.

As before, for each sensor node i, we compute how many nodes will be eliminated from the set of candidate sources, based on measurements from this sensor. As a result, we are able to determine the number of remaining potential sources, if we were to consider the measurements provided by monitor i. If this number is greater than the minimum cardinality set by the user, then consider the estimation based on measurements from node i. Otherwise, we discard these measurements and the set of potential sources remains as before.

Once we discard a sensor from the set of ranked monitors, we will not consider any other sensors which are less confident than the sensor we have discarded.

This method will ensure more accurate detection. Nevertheless, it might lead to a large cardinality of the set of candidate sources.

Step 13. Steps 9-12 are repeated using the same number of monitor nodes N_s, but with different values of $d_{estimated}^{MAX}$. For each case, a set of potential sources is obtained, of cardinality equal to the minimum cardinality value set by the user. The final set of candidate sources will be the union of the individual sets. This is done in order to ensure increased accuracy of the detection, by taking advantage of the following fact. Nodes with a small estimated distance have more accurate measurements. Nevertheless, if $d_{estimated}^{MAX}$ is too small, there might be insufficient monitors to ensure a small set of potential sources. Therefore, by taking the union of the candidate sources sets we increase the correct detection probability, while reducing the set of candidate sources.

Step 14. For each node in the set of potential sources, assign a rumor centrality level, based on the following criteria:

a. ***Criterion 1***: This involves the calculation of the sum of the distances from each potential source to all the monitor nodes, by considering sensor nodes for which the measured probability at time step $t = estimated\ shortest\ path + 1$ is higher than a certain threshold value. This approach is motivated by the fact that a potential source is more likely to have started the rumor is the distance to the monitor nodes which become infected after a short period of time is small on average. Therefore, the rumor centrality is calculated as follows: $RC = \sum_{i,V_i(d_i+1)>x} d_i$, where d_i is the estimated shortest path between the sensor and the source and V_i is the measured probability of node i having the rumor.

The weighting of this criterion should be positive.

b. ***Criterion 2****:* The formula below shows the calculation of the rumor centrality level, based on measurements from the sensor nodes and on the actual (not estimated) shortest paths between the candidate source and the sensors.

$$RC = \sum_{i\epsilon Set1} d_i + \sum_{i\epsilon Set2} Inf(\infty) + \sum_{i\epsilon Set3} 1000 + \sum_{i\epsilon Set4} 2000 + \sum_{i\epsilon Set5} 3000$$

In the above summation, the individual sets of each individual sum are:

$$Set\ 1 = \{node\ i\ |V_i(d_i + 1) > 0, V_i(d_i) = 0\}$$
$$Set\ 2 = \{node\ i\ |V_i(d_i) \neq 0\}$$
$$Set\ 3 = \{node\ i\ |V_i(d_i + 1) = 0, V_i(d_i + 2) \neq 0\}$$
$$Set\ 4 = \{node\ i\ |V_i(d_i + 1) = 0, V_i(d_i + 2) = 0, V_i(d_i + 3) \neq 0\}$$
$$Set\ 5 = \{node\ i\ |V_i(d_i + 1) = 0, V_i(d_i + 2) = 0, V_i(d_i + 3) = 0\}$$

The motivation behind the above penalties for each of the sets is the following.
Firstly, if the real distance between the candidate source and the monitor is d_i and there is a positive probability of this node having the rumor at time $k = d_i$, it would mean that the candidate source could not have started the rumor since the rumor could only reach the monitor node for the first time at time $k = d_i + 1$. Therefore, this candidate source will be penalized by Inf, as it cannot be the real rumor source.
Secondly, if the real distance between the candidate source and the monitor is d_i, and this monitor has not received the rumor yet at time $k = d_i + 1$, i.e. $V_i(d_i + 1) = 0$, this could be an indication of an erroneous measurement. Therefore, this would be penalized more compared to the case when $V_i(d_i + 1) > 0$.
The penalization should increase with the delay until the monitor first received the rumor, as seen in the summation above.
We should note that the constant included in the summation above could be replaced by any other values, as long as there are large.

c. ***Criterion 3***: The sum of the distances from each potential source to all the monitor nodes for which the measured probability is equal to 0. The weighting of this criterion should be negative.
d. ***Criterion 4***: The number of infected nodes, which are reachable from the potential source. The weighting of this criterion should be positive.
e. ***Criterion 5***: The number of not infected nodes, which are not reachable from the potential source. The weighting of this criterion should be negative.
The nodes in the set of candidate sources will be ordered according their rumor centrality, in ascending order of the rumor centrality level.
The top ranked node is an estimate of the source of rumor spreading in the network.

Other possible enhancements could include the following:

1. Sensor Choice Strategy
2. Implementation of Rayleigh's shortcut method for the calculation of the expected infection probability
3. Development of a source pseudo-likelihood function

The above algorithm will be tested in the *Evaluation* chapter, as follows:

Enhancement 1.1: Algorithm which employs *Criteria 1,3,*and *4* for the calculation of sensor confidence levels and *Strategy 1* to estimate the set of candidate sources.

Enhancement 1.2: Algorithm which employs *Criteria 1 and 2* for the calculation of sensor confidence levels and *Strategy 1* to estimate the set of candidate sources.

Enhancement 2.1: Algorithm which employs *Criteria 1 and 2* for the calculation of sensor confidence levels, *Strategy 1* to estimate the set of candidate sources, and *Criterion 1* for calculation of source rumor centrality.

Enhancement 2.2: Algorithm which employs *Criteria 1 and 2* for the calculation of sensor confidence levels, *Strategy 2* to estimate the set of candidate sources, and *Criterion 1* for calculation of source rumor centrality.

Enhancement 2.3: Algorithm which employs *Criteria 1 and 2* for the calculation of sensor confidence levels, *Strategy 2* to estimate the set of candidate sources, and *Criterion 2* for calculation of source rumor centrality.

Enhancement 2.4: Algorithm which employs *Criteria 1 and 2* for the calculation of sensor confidence levels, *Strategy 3* to estimate the set of candidate sources, and *Criterion 2* for calculation of source rumor centrality.

Chapter 5

Evaluation

In this chapter we describe the evaluation methods used to assess the performance of the source detection method.

The structure of this chapter is the following. First of all, the evaluation criteria are described. Then the following section assesses the correctness of the theoretical formula for the probability of rumor spreading, by comparing it with the measured probability for various network topologies, as well as network parameters. Furthermore, the performance of the estimation of the shortest paths between the monitor and the source nodes is assessed. The first evaluation method is to compute the frequency of correct distance estimation, for various values of the distance. The second evaluation method consists of computing the error in number of hops, between the estimated and the actual source. Both methods will be tested for several network sizes radii, and topological characteristics.

Moreover the source estimation algorithm is evaluated as follows. Different enhancement stages of the detection algorithm will be evaluated to assess the improvement due to each enhancement. In addition, the probability of correct detection will be computed, as well as the number of candidate sources, for various sizes of the set of monitoring nodes.

Evaluation Criteria

The following criteria will be used to assess the solution:

1. Epidemic Model: The simulation of the spreading of rumors is performed in a correct manner. The assessment is done through Matlab simulations and visualizing the evolution of the rumor in the network over time.
2. Probability Error: The plots of the predicted probabilities from analytical formulas should approach the plots of the actual measured probabilities when simulating a spreading of rumors within the network, for any network topology. The absolute error difference between the predicted probability of a node being infected, and the actual measured probability should give an indication of the correctness of the mathematical formulas derived.
3. Distance Error: The distance error is the difference between the estimated shortest path length and the actual distance between the sensor node and the source. The frequency of a certain distance error value will be given by the number of times this distance error occurs out of a fixed number of experiments of rumor spreading. The plot of the frequency of error against various values of error (e.g. error of one hop, error of two hops etc.), should give an indication of how accurately this distance is estimated. The plot should be repeated for networks of different topology and size, as well as for various number of sensor nodes.

4. Number of Sensor Nodes: The required number of sensor nodes for which the source can be accurately estimated should be plotted as a function of the network size. In addition, the fraction of observers (ratio of number of monitor nodes over total number of nodes), should also be plotted against different network parameters.
5. Number of Time Observations: In order to reduce the complexity of the algorithm, as few number of observations as possible should be used. In this sense, we would be interested in finding the minimum number of time observations that would allow accurate source detection.
6. Source Detection Accuracy (single source): Due to some distances being wrongly estimated, the algorithm might not always be able to detect a single source node, or there may be cases when the source node cannot be detected at all. The detection accuracy could be determined by plotting the number of estimated sources for different numbers of sensor nodes, and for different topologies and graph sizes. In addition, it would be interesting to verify how the accuracy degrades/improves when considering only some of the available node measurements. For example, by utilizing only those measurements from nodes closer to the source, the detection performance could be improved since the distance estimation is more accurate in these cases.
7. Source Location Error: In some cases, the source might not be correctly estimated. For these cases, we should evaluate how far the estimated source is from the actual infectious source. This could be achieved by calculating the distance between the estimated and the correct source nodes. By repeating the experiment for a set number of times, we could find the frequency at which a certain error distance occurs, which could give an indication of how good the estimation of the source is. For example, the estimation is better if an error of 1 hop occurs more frequent than an error of 2 hops.
8. Noise Robustness: The algorithm should be robust to noise in the system. One example of noise could be the wrong estimation of the shortest distances between the sensor and source nodes, or mis-information from various sensor nodes. The algorithm should be tested under various noise scenarios and the detection accuracy should be measured. The main test for the robustness to noise will be performed by using a small number of rumors for the simulation of the information dissemination. In this case, the individual sensor measurements might significantly deviate from the expected value, leading to errors in the estimation of the shortest paths between the sensors and the source.
9. Algorithm Complexity: The complexity of the Detection Algorithm should be reduced. This is done by calculating the time complexity of the different blocks of the algorithm and how this depends on the network size, and the number of observations (number of sensor nodes and number of time steps at which we monitor the nodes).

Evaluation of Theoretical Probability Formula: Initial Solution

The theoretical probability formula is evaluated using the following methods.

Firstly, the accuracy of the assumptions made in the derivation of the formula, as well as the mathematical simplifications used, will be evaluated by comparing the theoretical probability against the probability obtained through simulations. Tests will be performed for various network topologies, as well as different spreading probabilities.

Secondly, the estimation algorithm is based on the comparison between the theoretical and simulated spreading probabilities. This comparison determines the estimated shortest path between the monitor node observed and the potential source of rumors. Therefore, in order to assess the accuracy of the theoretical probability, we could measure the error probability of the estimated distance between the sensor and source nodes. The shortest path error probability will be plotted against various number of monitors, as well as different values of rumors.

The graphs below illustrate the results obtained through simulating a spreading of rumors through a small-world network of $N = 200$, compared with the theoretic probabilities computed using the formula above, for various values of distances between the monitor node and the source. The different subplots correspond to different values of spreading probability, $P_s \in \{0.5, 0.6, 0.7\}$. Moreover, the connectivity index used for the theoretic probability formula is $\kappa = 1.2$.

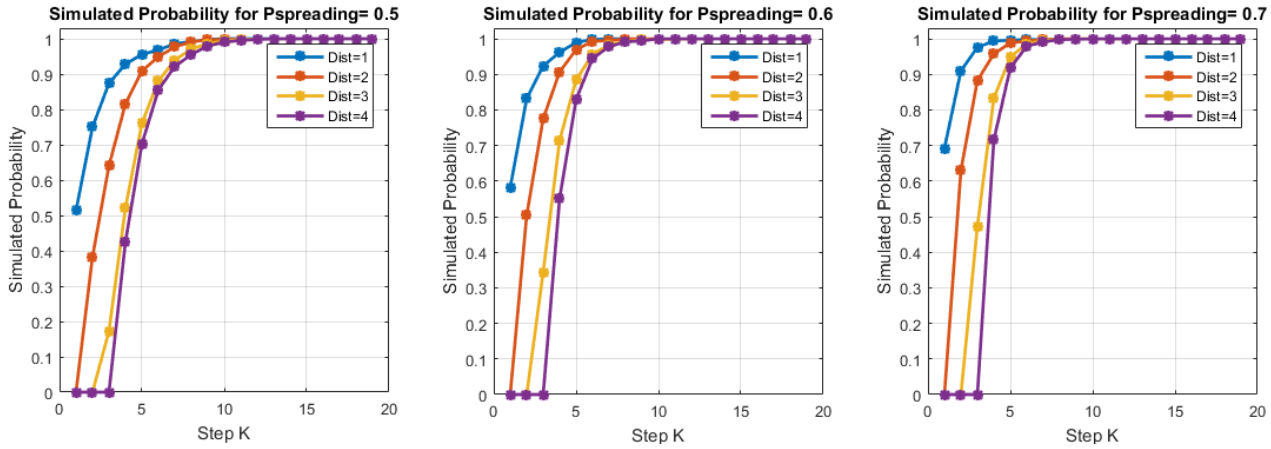


Figure 16: Simulated Probabilities of Nodes being infected, for Different Values of Spreading Probability

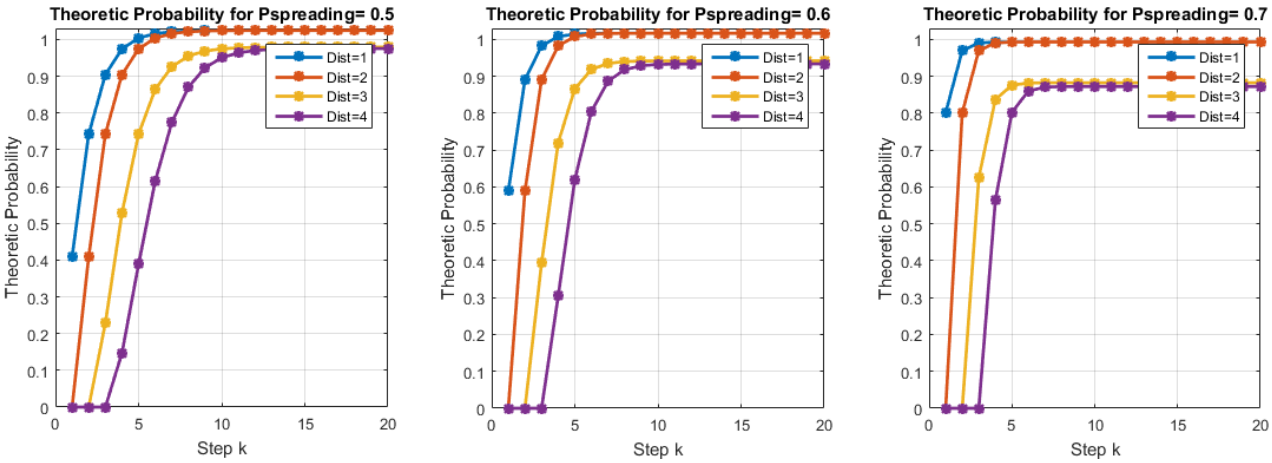


Figure 17: Estimated Probabilities of Nodes being infected, for Different Values of Spreading Probability

The plots below show a comparison between the simulated and the theoretic probabilities of rumor spreading, in a small-world network of size $N = 200$ nodes, for various values of monitor distances. The different subplots correspond to different spreading probabilities, $P_s \in \{0.2, \dots, 0.9\}$. The connectivity index is $\kappa = 1.2$.

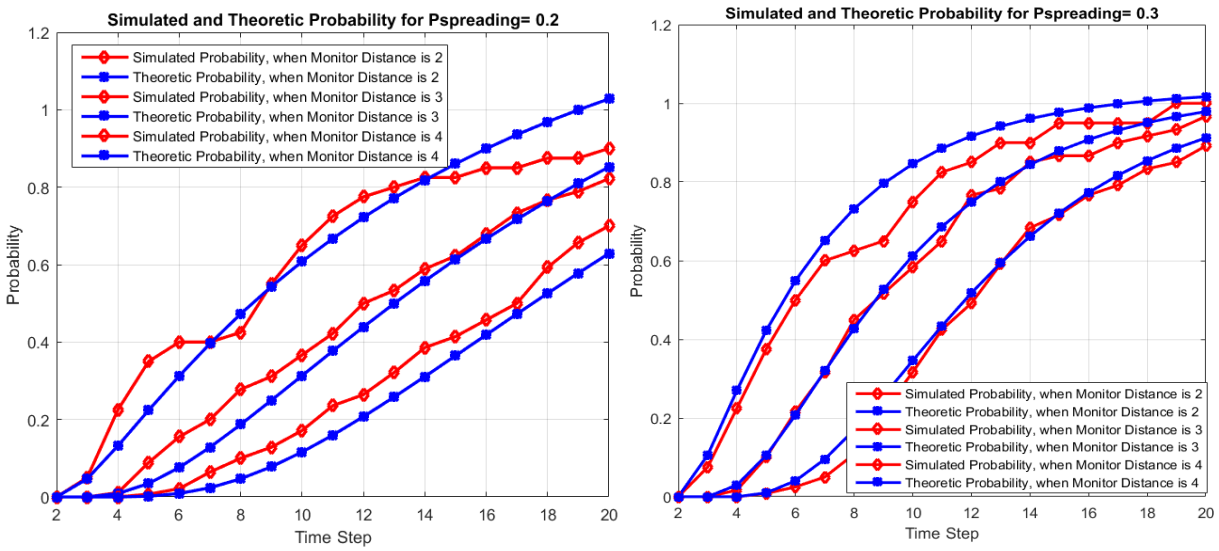


Figure 18 Simulated and Theoretic Probabilities in a Small-world Network, for Spreading Probability Ps=0.2 (left) and Ps = 0.3 (right)

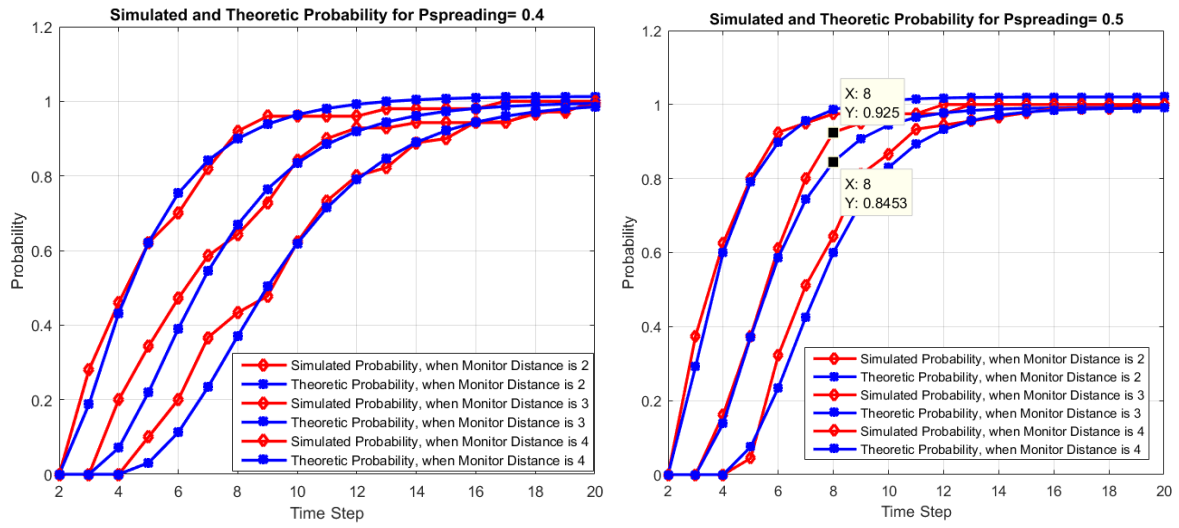


Figure 19: Simulated and Theoretic Probabilities in a Small-world Network, for Spreading Probability Ps=0.4 (left) and Ps = 0.5 (right)

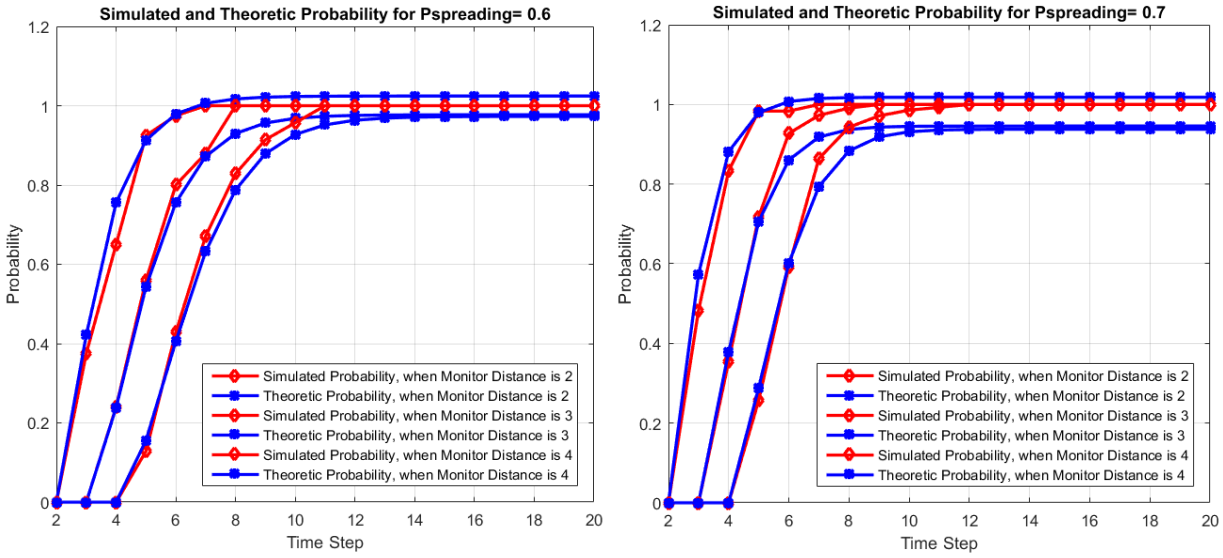


Figure 20: Simulated and Theoretic Probabilities in a Small-world Network, for Spreading Probability Ps=0.6 (left) and Ps = 0.7 (right)

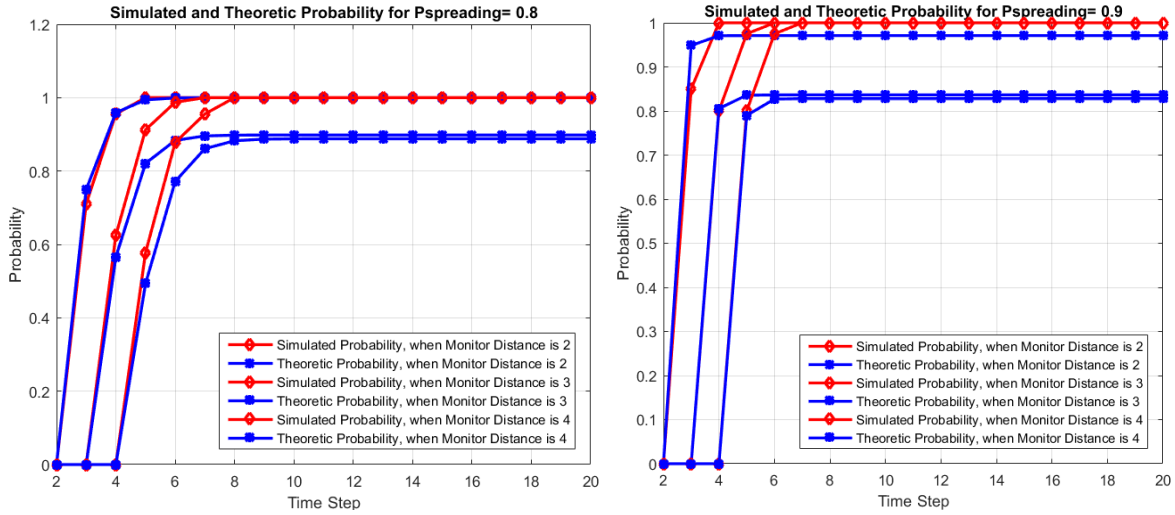


Figure 21: Simulated and Theoretic Probabilities in a Small-world Network, for Spreading Probability Ps=0.8 (left) and Ps = 0.9 (right)

The plots below show the theoretic probabilities, and the ones obtained from measurements, in a scale-free network, of size $N = 243$, in a random geometric graph of size $N = 200$, and in a tree graph of depth $d = 8$ and $no_{children} = 2$.

In all cases, the probability of spreading is $P_S = 0.5$, and the *connectivity Index* used in the probability theoretic formula differs for all the cases, as different networks have different characteristics.

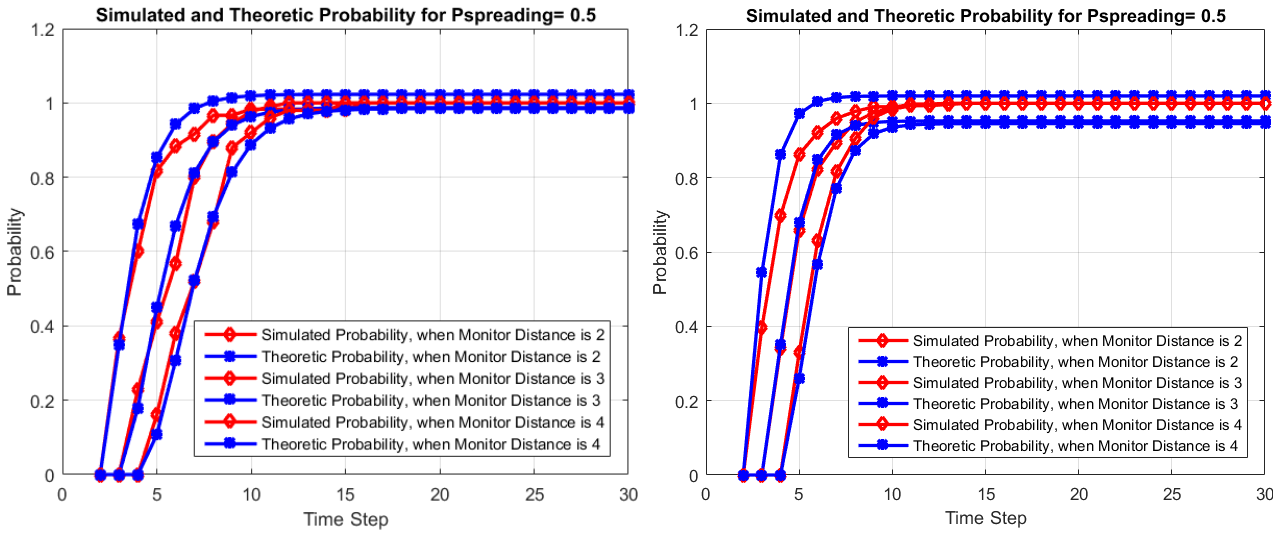


Figure 22: Simulated and Theoretic Probabilities, in a Scale-Free Network (left) and Random Geometric Graph (right), using the Initial Solution

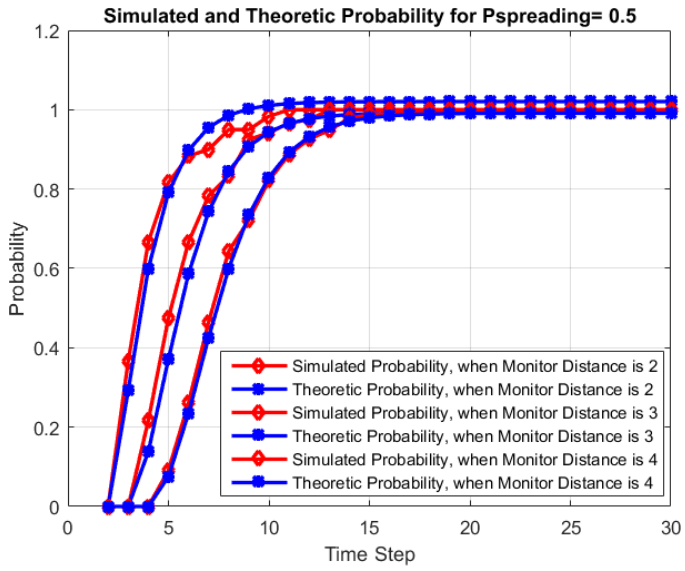


Figure 23: Simulated and Theoretic Probabilities in a Tree Graph, using the Initial Solution

While the plots above use measurements from a simulated spreading of rumors of $R = 20$ rumors, the graphs below show the comparison between the simulated and theoretic probabilities, for a very large number of rumors, $R = 200$. The simulations are performed in a small-world network, of size $N = 200$ nodes. We can see that in most cases the theoretical probability values coincide or are very close to the values obtained from simulation. Nevertheless, the theoretical probabilities do not converge to a value of $P = 1$ in some cases, and this may be a result of the approximations used in the derivations of this formula. However, the converging values are very close to $P = 1$ and the probability could be truncated to this maximum value.

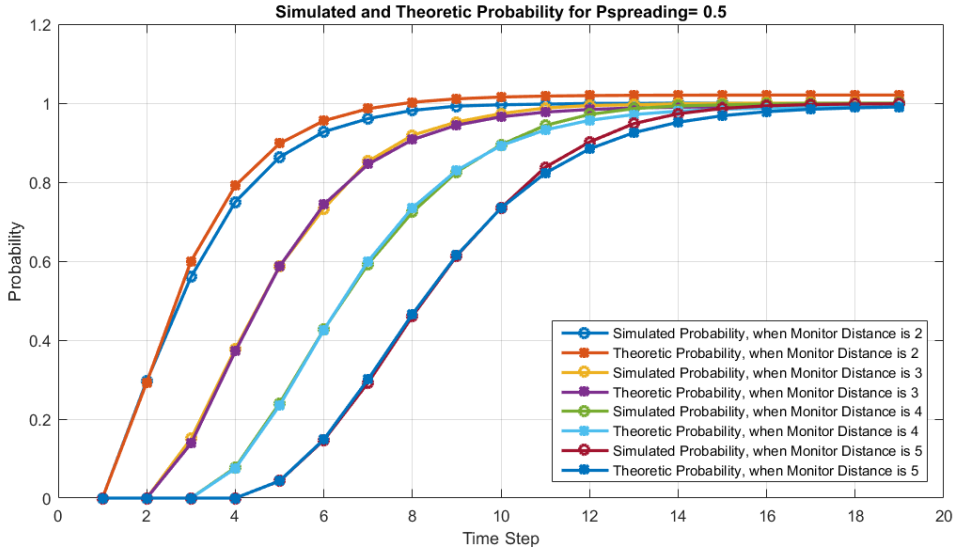


Figure 24: Simulated and Theoretic Rumor Probabilities for Pspreading = 0.5

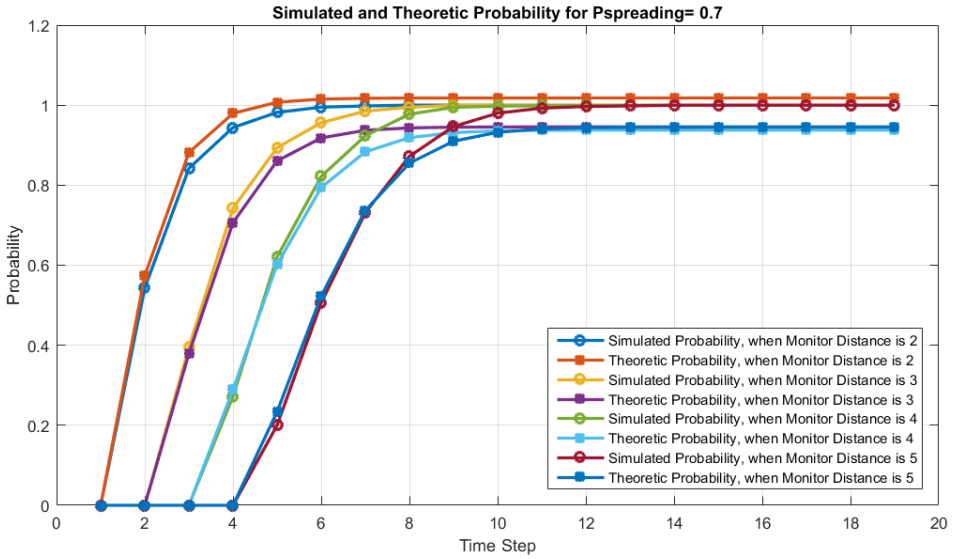


Figure 25: Simulated and Theoretic Rumor Probabilities for Pspreading = 0.5

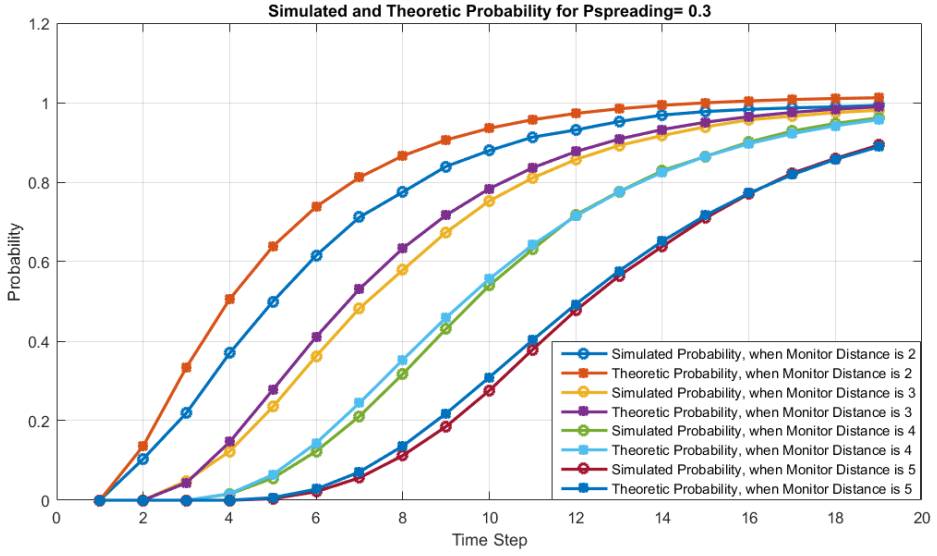


Figure 26: Simulated and Theoretic Rumor Probabilities for Pspreading = 0.3

Evaluation of Mathematical Approximations: Initial Solution

As derived in Chapter 2 of this report, the probability that a node will have the rumor at time step k is equal to:

$$Q_d(k) = \sum_{t=d}^{k} (\mu \times \kappa)^d \times (1 - \mu \times \kappa)^{t-d} \times \frac{2^t}{\sqrt{2\pi(t-1)}} \times e^{\left(-\frac{(t-2d+1)^2}{2(t-1)}\right)}$$

The formulation above has been obtained using Stirling's and Taylor approximations for the simplification of the expression for the number of paths. Nevertheless, without any mathematical simplifications, the probability that a node will have the rumor at time step k is equal to:

$$Q_d(k) = \sum_{t=d}^{k} (\mu \times \kappa)^d \times (1 - \mu \times \kappa)^{t-d} \times \binom{t-1}{d-1}$$

The plots below show a comparison between the theoretic probability with and without mathematical approximations. Firstly, we notice that the approximated formulation does not saturate at a value of $P = 1$ in all cases, while the theoretical probability formula with no simplifications converges to a maximum of $P = 1$, which is the expected limit of a cumulative distribution function. Hence, while the non-approximated theoretical probability gives more accurate results, the approximated probability is highly accurate, converging to the former one.

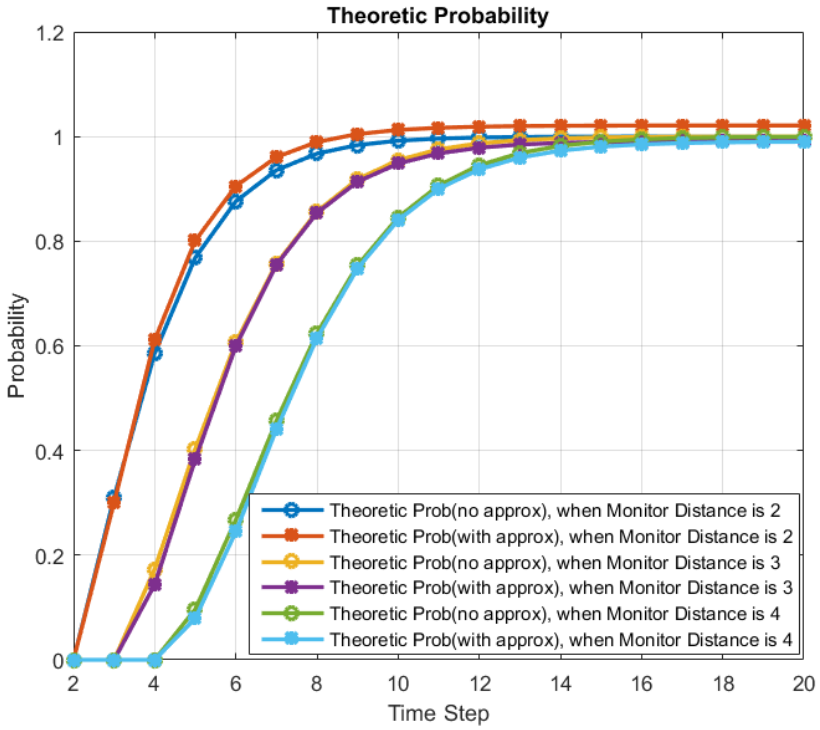


Figure 27: Theoretical Probabilities of Rumor Infection, with and without Mathematical Approximations

The plots below show a comparison between the non-approximated theoretical probability and the simulated one (left), and between the approximated theoretical probability and the simulated one (right). The simulated probability has been obtained by averaging a number of $R = 100$ rumor experiments. We can see that in both cases, the theoretic probability converges to the experimental one. Nevertheless, the non-approximated formula gives more accurate results for smaller sensor distances, and ensures a saturation at $P = 1$.

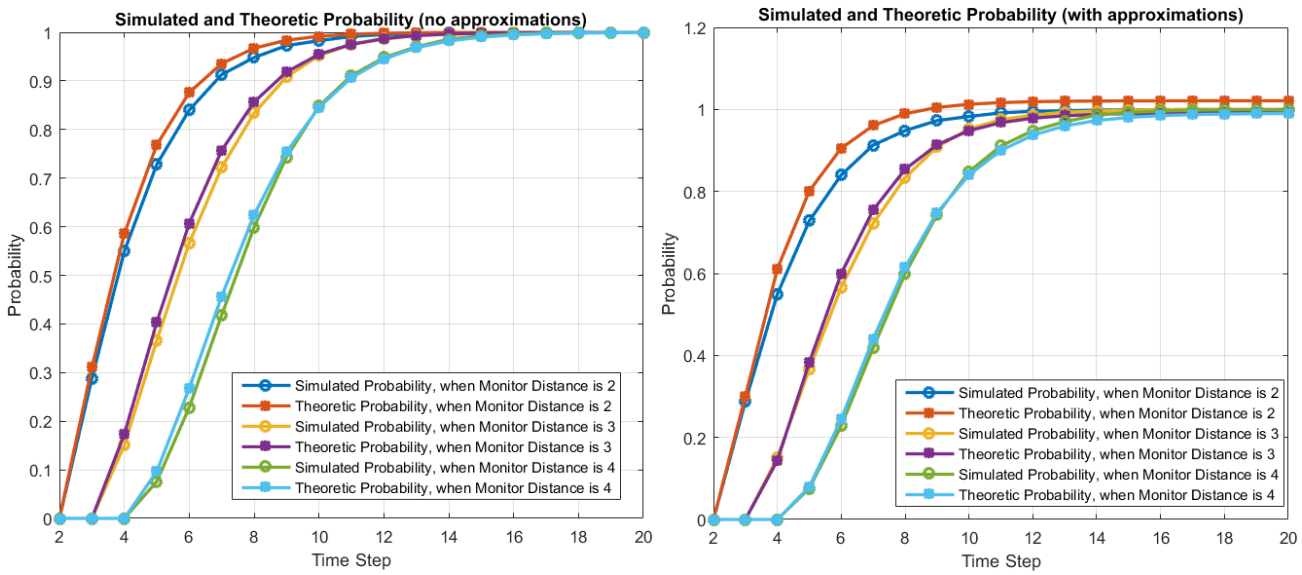


Figure 28: Simulated and Theoretical Probability with no Mathematical Approximations (left), and with Approximations (right)

In summary, the closed-form expression for the theoretical probability, obtained through mathematical simplifications, converges to the non-approximated formulation and will therefore be used throughout this report, as part of the source detection algorithm.

Evaluation of Theoretical Probability Formula: A Robust Solution

The plots below show a comparison the simulated and theoretic probabilities, obtained using the robust formula for the calculation of the probability of rumor infection.

The simulated probability was obtained using the results from a spreading of rumors in a small-world network of $N = 200$ nodes, average degree $V = 4$, and rewiring probability $P = 0.2$. The spreading model assumes spreading to exactly 1 neighbour with probability $P_s = 1$. This assumption is made as a result of the fact that the theoretic probability formula does not consider any multiplication of rumors and in addition, it assumes that the rumor is guaranteed to follow one of the $A, B, or\ C - type$ segments at any time step. Moreover, it is assumed that $pA = pB = pC = 0.33$, values which are used for the calculation of the theoretic probabilities.

The plots below show the simulated and theoretic probabilities. The latter is obtained using the following formula, which does not take into account the reflected paths:

$$Q_d(k) = \sum_{t=d}^{k} \sum_{d \le A \le k} \{ \binom{k-1}{A-1} \binom{k-A}{k+d-2A+1} \} {p_A}^A {p_B}^{k+d-2A} {p_C}^{A-d}$$

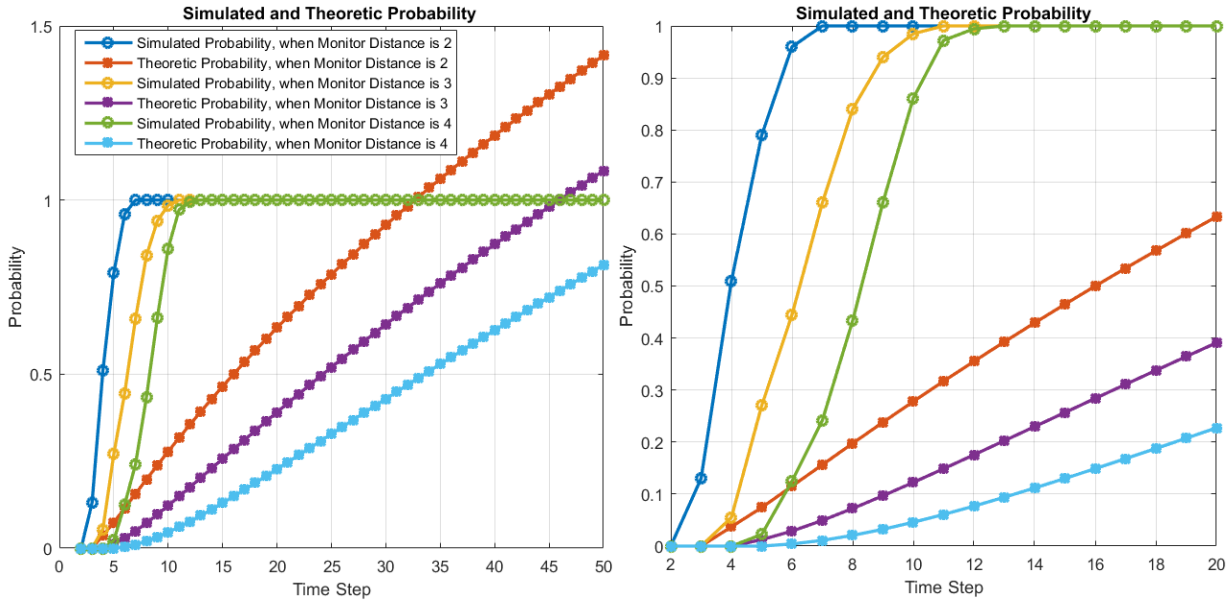


Figure 29: Simulated and Theoretic Probabilities assuming no Reflected Paths, for 50 Time Steps (left) and 20 Time Steps (right)

In the figure below, the left subplot shows the theoretic probability calculated using the following formulation:

$$Q_d(k) = \sum_{t=d}^{k}\{\sum_{d\le A\le k}(\binom{k-1}{A-1}\binom{k-A}{k+d-2A+1}{p_A}^{A}{p_B}^{k+d-2A}{p_C}^{A-d} - \sum_{y=2}^{k-d}[\sum_{A_P=1}^{floor\left(\frac{y}{2}\right)}\binom{y-1}{A_P-1}\binom{y-A_P}{A_P-1}\binom{k-y}{A-d-A_P}\binom{k-y-(A-d-A_P)}{A-A_P}]{p_A}^{A-d}{p_B}^{k-d-2(A-d)}{p_C}^{A}])\}$$

The right subplot shows the theoretic probability calculated as follows:

$$Q_d(k) = \sum_{t=d}^{k}\sum_{d\le A\le k}\{\binom{k-1}{A-1}\binom{k-A}{k+d-2A+1} - \sum_{d\le A\le k}\binom{k-1}{A-1}\binom{k-A}{k-d-2A+1}\}{p_A}^{A}{p_B}^{k+d-2A}{p_C}^{A-d}$$

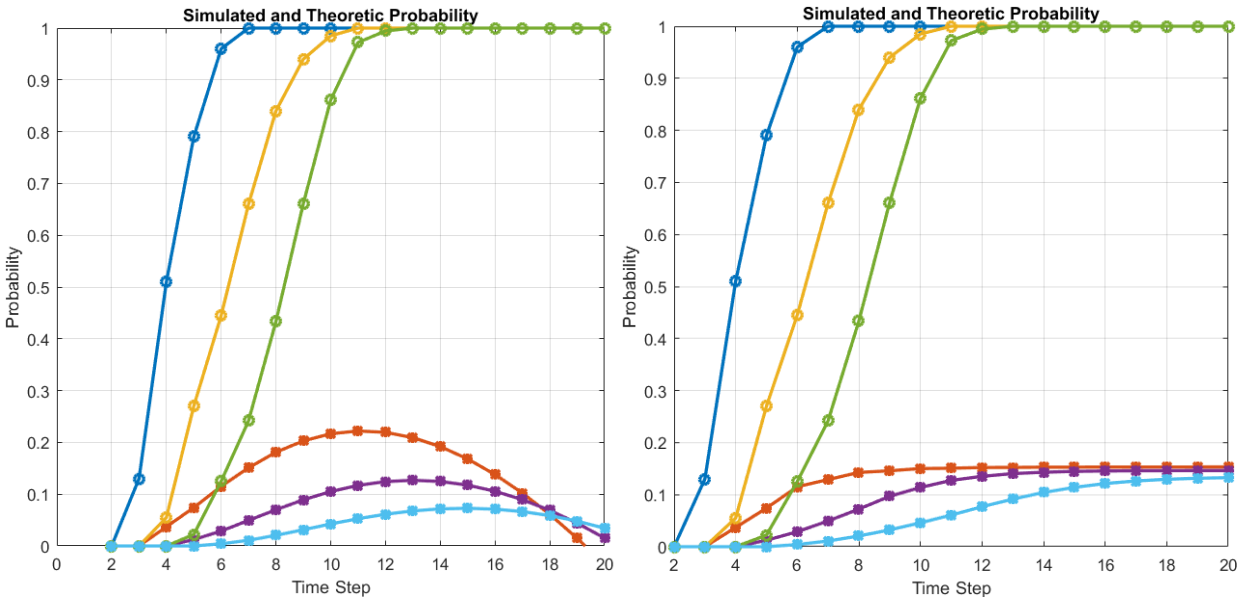


Figure 30: Simulated and Theoretic Probabilities with Calculation of Reflected Paths using Two Different Methods

As we can see from the subplots above, the theoretic probability does not converge to the probability obtained through simulations. This might be a result of the approximation that the probabilities of advancing through the network are assumed constant and equal, i.e. $pA = pB = pC = 0.33$. This could also be a result of the calculation of the reflected paths used for the derivation of the theoretic probability, and from the result above we can deduce that the number of reflected paths might be overestimated.

Furthermore, the first method of calculating the reflected paths (left) gives more accurate results at small time steps. However, the accuracy degrades significantly for large time steps compared to the second formulation (right).

Finally, in the figure below we plot the simulated and theoretic probabilities, calculated by assuming that the first segment in the path of the rumor is not necessarily an $A - type$ segment. This could be justified by the fact that the path of the rumor may be a $B - type$ segment, in the case the source does not spread the rumor to any of its neighbours for the first time step.

In this sense, the left subplot shows the theoretic probability calculated using the following formulation:

$$
\begin{aligned}
&Q_d(k) \\
&= \sum_{t=d}^{k} \{ \sum_{d \le A \le k} (\binom{k}{A} \binom{k-A}{k+d-2A} {p_A}^{A} {p_B}^{k+d-2A} {p_C}^{A-d} \\
&- \sum_{y=2}^{k-d} [\sum_{A_P=1}^{floor\left(\frac{y}{2}\right)} \binom{y}{A_P} \binom{y-A_P}{A_P} \binom{k-y}{A-d-A_P} \binom{k-y-(A-d-A_P)}{A-A_P}] {p_A}^{A-d} {p_B}^{k-d-2(A-d)} {p_C}^{A}]) \}
\end{aligned}
$$

The right subplot shows the theoretic probability calculated as follows:

$$
Q_d(k) = \sum_{t=d}^{k} \sum_{d \le A \le k} \{ \binom{k}{A} \binom{k-A}{k+d-2A} - \sum_{d \le A \le k} \binom{k}{A} \binom{k-A}{k-d-2A} \} {p_A}^{A} {p_B}^{k+d-2A} {p_C}^{A-d}
$$

The results show an improvement in the theoretic probability formulation, particularly for the second calculation method.

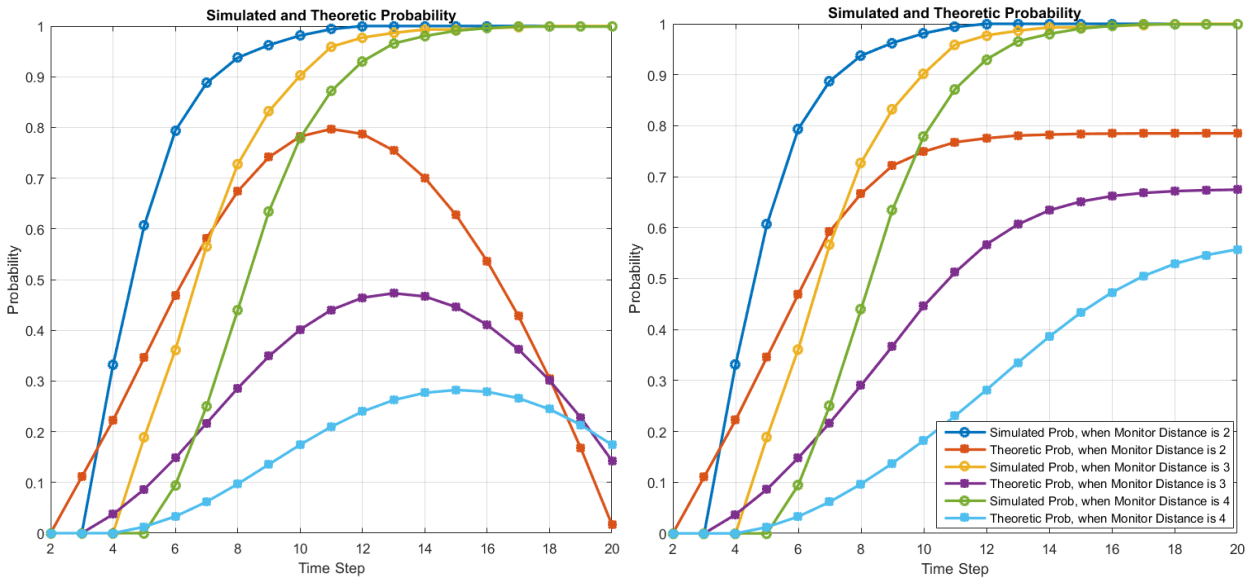


Figure 31: Simulated and Theoretic Probabilities with Calculation of Reflected Paths using Two Different Methods, with no Constraints on the First Path Segment

Overall, the formulation of the probability of infection which does not account for the illegal paths gives an upper bound on the real spreading probability. On the other hand, the formulation which includes the elimination of the illegal paths using the reflection method leads to a lower bound on the real probability of infection. Nevertheless, further improvement of the theoretic formulation will remain as a part of future development.

Evaluation of the Algorithm for Estimation of Shortest Paths (Accuracy)

The plots below show the frequency of the error of estimation of the shortest paths between the sensor nodes and the source, as well as the number of error hops against the number of monitors observed. The error hops represent the difference between the estimated distance and the actual distance between the sensor and the source. The tests are performed in a small-world network of size $N = 200$ nodes, using 100 repeated experiments in order to compute an average error of estimation.

In the plots below, the number of rumors used for the simulation of a spreading of rumors, which in turn is used to determine the optimal parameters in the theoretical probability formula, is $R = 40$.

We can see that the accuracy of estimation increases as the number of rumors used for the actual rumor simulation increases, as the probability of error decreases, while the number of error hops decreases as well. Furthermore, as the monitor distance increases, the error hops become larger.

We can also observe that the error probability is very small for monitor distance lower than $d = 3$. Nevertheless, there is a small error even for $d \leq 3$ and an explanation for this result might be the fact that even if the theoretical probability represents a good approximation of the average simulated probability (obtained using a very large number of rumors), the actual probabilities obtained from the rumor spreading (using a lower number of rumors) might not deviate from the average simulated probability and hence, the theoretical one. In order to account for this, the estimation algorithm should include the elimination of those monitor nodes for which the individual simulated probability significantly deviates from the average simulated probability.

Moreover, we can notice that for the cases when the distance is wrongly estimated, the number of error hops, calculated as the absolute value of the difference between the estimated distance and the actual distance between the monitor and the node, is typically $d_e = 1\ hop$.

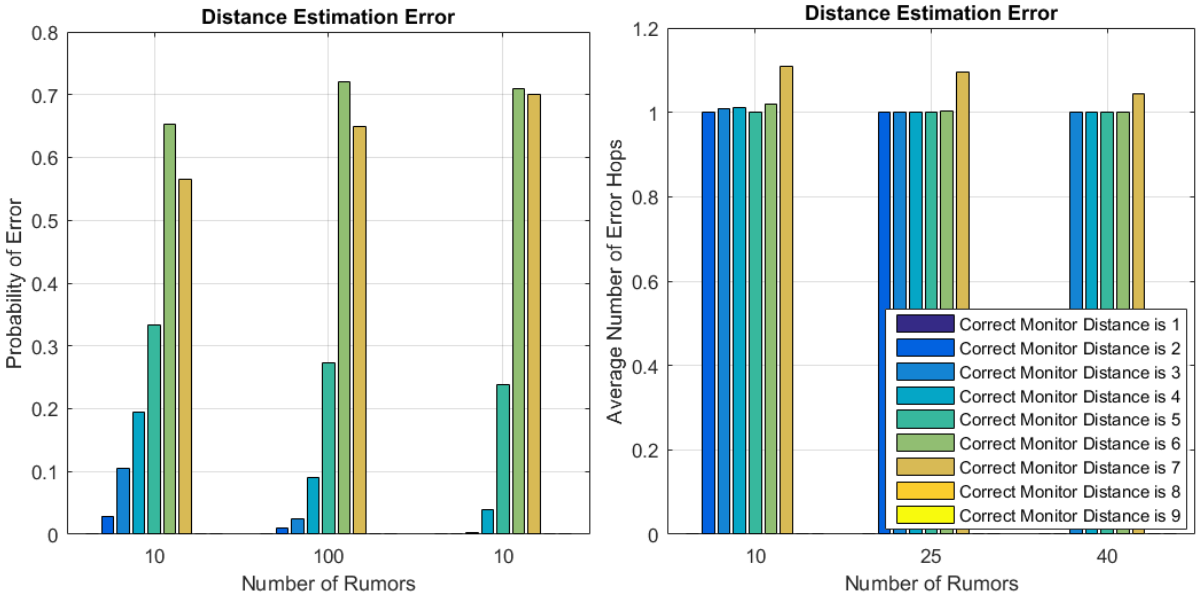


Figure 32: Distance Estimation Error for Different Numbers of Rumors, in a Small-world Network of 200 Nodes, with Increased Accuracy of Theoretical Probability Parameters

The plots below show the probability of estimated distance error and the average number of error hops between the monitor and the source, as the number of monitors varies from $M = 10$ to $M = 50$ monitors, for a number of rumors $R = 10$. We can notice that the number of monitors contributes to the accuracy of detection, as generally as a larger number of monitor nodes leads to an increased error probability, particularly at large monitor distances.

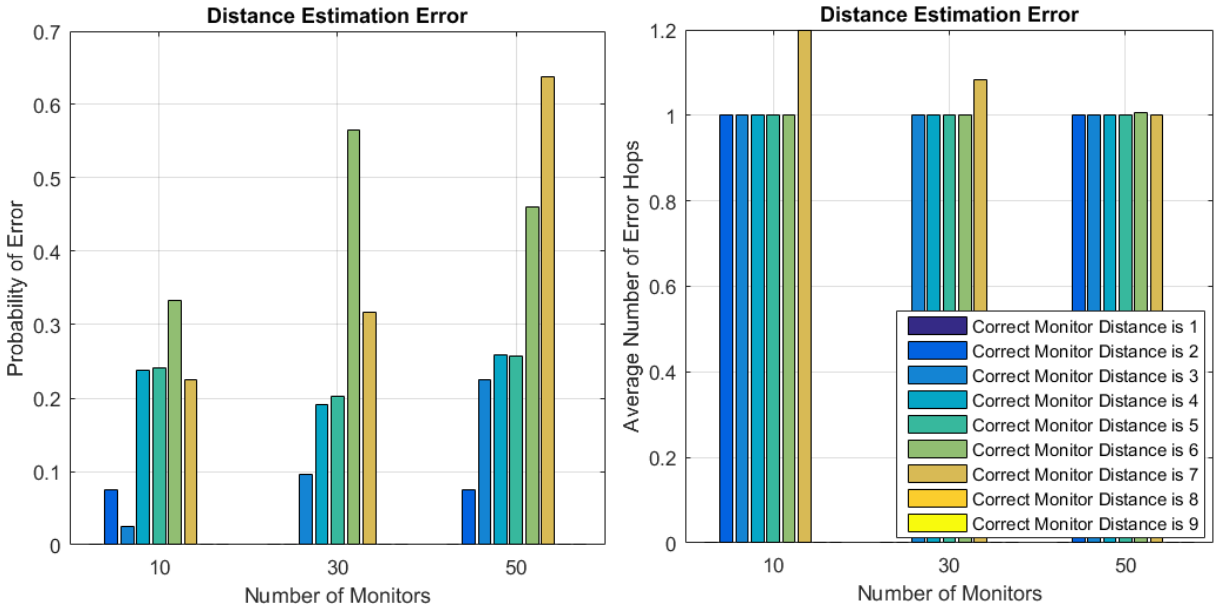


Figure 33: Distance Estimation Error for Different Numbers of Monitors, in a Small-world Network of 200 Nodes, with Increased Accuracy of Sensor Measurements

In the plots below, the number of rumors used for the simulation of a spreading of rumors, which in turn is used to determine the optimal parameters in the theoretical probability formula, is $R = 5$.

We can see that the estimation of shortest paths is less accurate when the theoretical probability is determined from a simulation of a lower number of rumors, particularly at lower monitor distances. As we can observe, the error probability is higher, and the average number of error hops is larger as well. For example, for a monitor distance $d = 3$ and using 10 monitor nodes, the probability of error is $P \cong 0.025$ for a larger number of rumors ($R = 10\ above$) used to derive the parameters of the theoretic probability, while it increases to $P \cong 0.1$ when using less rumors ($R = 5\ below$).

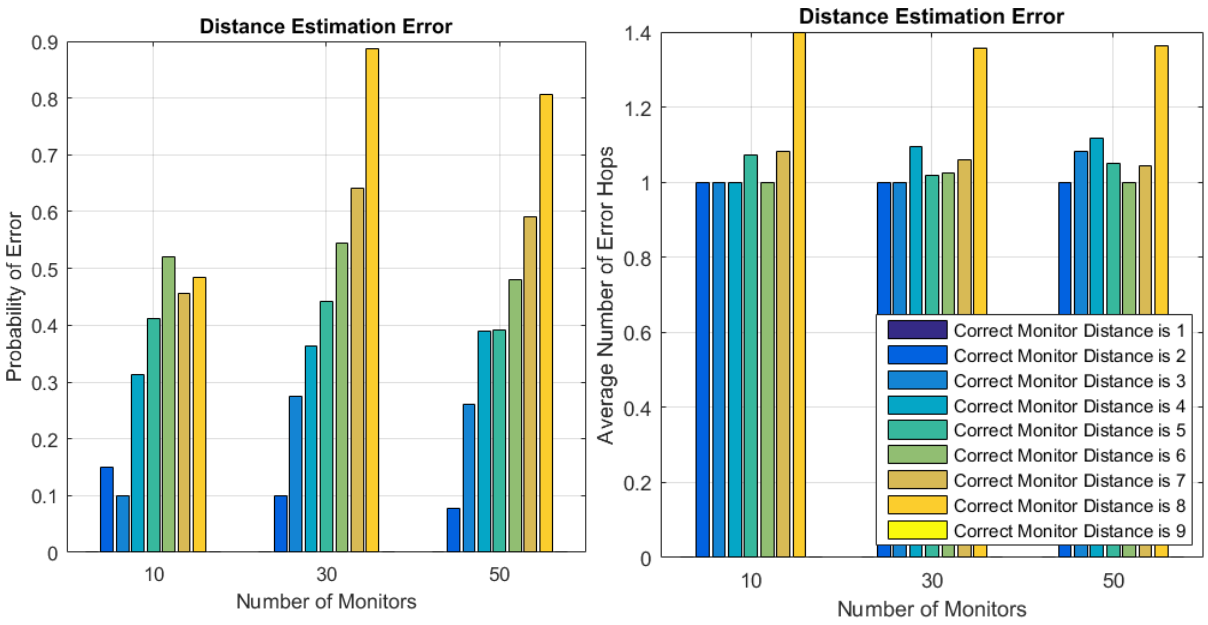


Figure 34: Distance Estimation Error for Different Numbers of Monitors, in a Small-world Network of 200 Nodes, with Low Accuracy of Sensor Measurements

In the plots below, the number of rumors used for the simulation of a spreading of rumors, which in turn is used to determine the optimal parameters in the theoretical probability formula, is $R = 100$. We can notice a significant improvement in the estimation of the shortest distances, particularly when the monitor distance is low. For example, in the plots below we can see that the probability of error for distances $d \leq 3$ is $P = 0$, while for $d = 4$ the probability is very small.

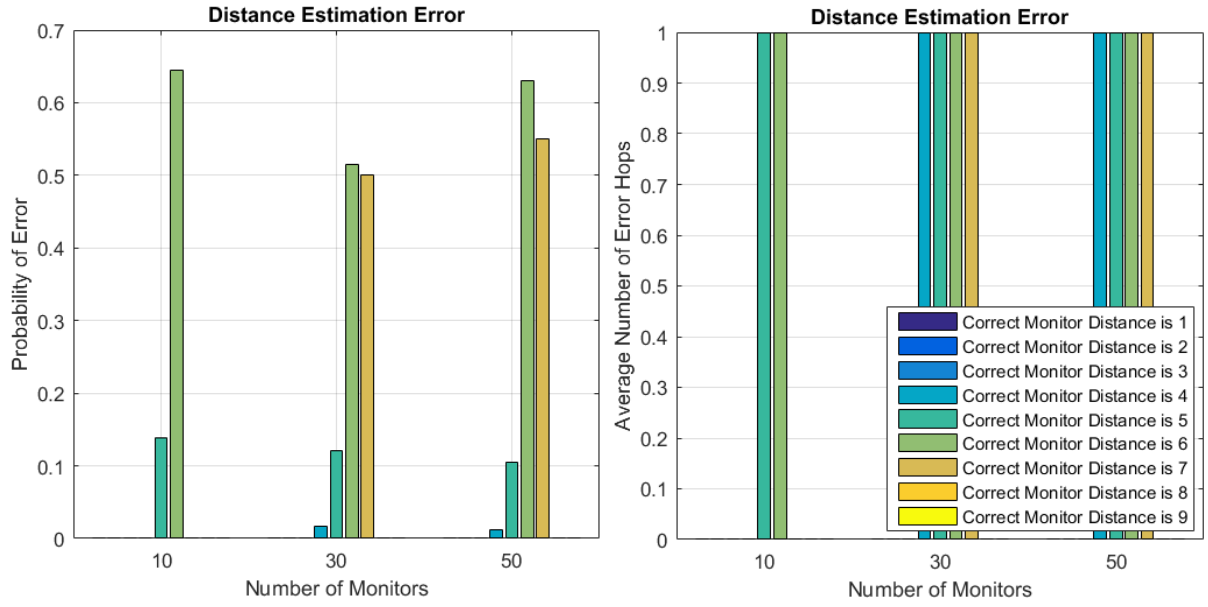


Figure 35: Distance Estimation Error for Different Numbers of Monitors, in a Small-world Network of 200 Nodes, with Highest Accuracy of Sensor Measurements

One of the main reasons for the existence of a distance estimation error, even though the theoretic probability converges to the averaged simulated probability could be the deviations in the individual sensor measurements from the expected value. This is more likely to happen when a small number of rumors is available for the actual rumor spreading process observed.

For example, the results below show that noise in the sensor measurements. Specifically, we show the deviation of probability of rumor from the average value. This leads to an incorrect shortest path between the monitor and the source being computed. In this case, the estimate distance is $d = 7$, while the actual ground truth distance is $d = 6$. As it can be seen from the data below, the actual sensor measurements approach the values of the theoretic probability for $d = 7$. In addition, it is typically expected that the probability of a node located at distance $d = 6$, measured at time step $k = 7$ should be positive, while the sensor measurement is in this case 0.

Time Step	1	2	3	4	5	6	7	8	9	10	11	12	13
Individual Sensor Measurement	0	0	0	0	0	0	0	0.1	0.1	0.3	0.4	0.7	0.8
Theoretic Probability Approximate Values, $d = 7$	0	0	0	0	0	0	0.03	0.10	0.22	0.36	0.52	0.65	0.76
Theoretic Probability Approximate Values, $d = 6$	0	0	0	0	0	0.04	0.14	0.29	0.46	0.6	0.74	0.83	0.89

Table 5: Illustration of Noise in Sensor Measurements

In summary, using these observations, we can improve the performance of the detection algorithm, as follows: by considering a larger number of rumors for the simulation used to derive the optimal theoretical probability parameters, by assuming a large number of rumors for the actual spreading probability, and by considering measurements obtained from monitors for which the estimated distance is small. The assumption of a large number of rumors is justified by the motivation of this project, to estimate the source of information dissemination in a network where we assume numerous *attacks* from the same source.

Evaluation of the Algorithm for Estimation of Shortest Paths (Robustness)

In order to evaluate the robustness of the estimation of the shortest distances between the monitor nodes and the source, based on the derived theoretical probability for rumor infection, the following tests were performed. Firstly, the average probability of rumor infection was derived by simulating a spreading of multiple rumors in a small-world network. The optimal *connectivity index* in the theoretical probability of rumor dissemination was found through the minimum mean-square error method, based on the average simulated probability. The shortest paths between the monitors and the source were then estimated using various values of the connectivity index and the results were compared with the optimal case.

The plots below show the distance estimation errors, using 30 monitors and 100 rumors, for different values of the connectivity index. A large number of rumors was chosen in order to ensure that the sensor measurements approach their expected value, such that they do not impact the distance estimation.

The optimal value of this parameters was found to be $k = 1.215$. We can see that around the optimal value, the probability of errors achieves a minimum compared to the other cases when the monitor distance is smaller than $d = 5$. In addition, the average number of error hops is also small in this case, with $d_e = 1\ hop$.

Moreover, we can see an increase in the distance estimation error when the value of the connectivity index is much larger/smaller compared to the optimal one. Nevertheless, the increase in the estimation error is not

significant, particularly for small monitor distances, and provided the deviation of the connectivity index from its optimal value is not too big.

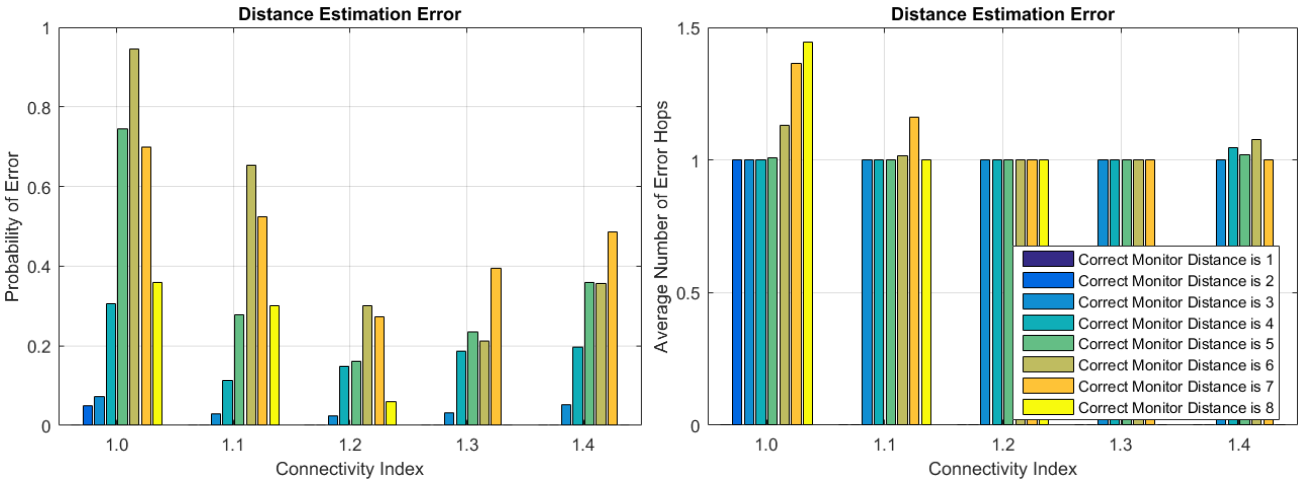


Figure 36: Distance Estimation Error in a Small-world Network, using Different Values of the Connectivity Index with Maximum Deviation from the Optimal Value $\Delta k = 0.2$

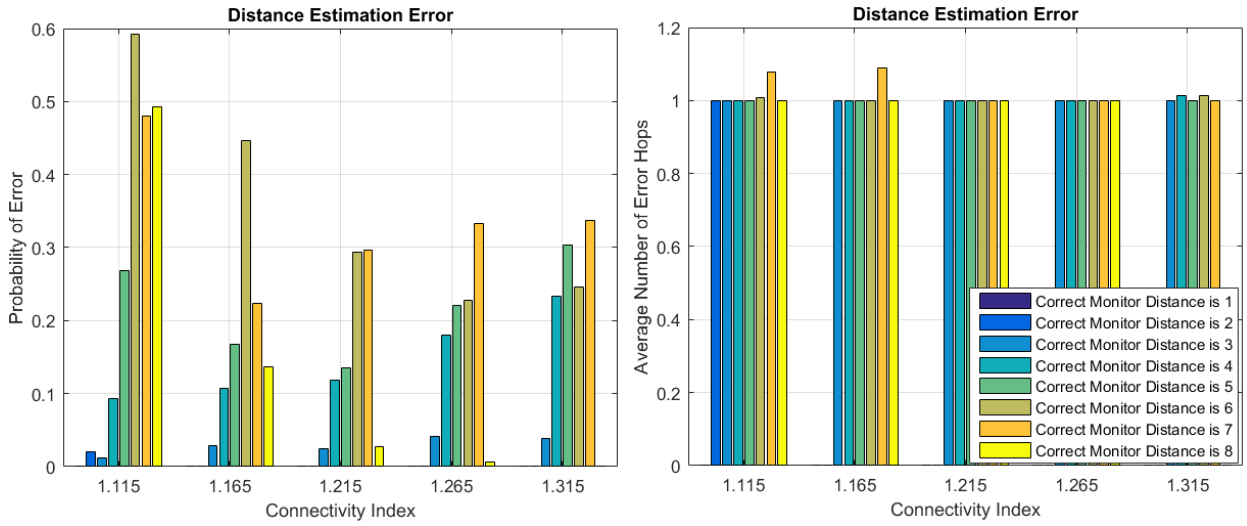


Figure 37: Distance Estimation Error in a Small-world Network, using Different Values of the Connectivity Index with Maximum Deviation from the Optimal Value $\Delta k = 0.1$

In summary, the distance estimation algorithm is robust to small variations in the parameter of the theoretical probability formula around its optimal value, i.e. the connectivity index. As a result, the estimation of the optimal connectivity index in a network of given topology could be a good approximation for the connectivity index of networks of similar topologic characteristics, even if the networks are not identical.

Evaluation of the Source Detection Algorithm

This section describes the evaluation methods used to assess the performance of the source detection algorithm. In particular we evaluate the following enhancements of the algorithm:

1. Sensor Confidence Level and Adaptive Connectivity Index
2. Source Rumor Centrality

For each of these, the performance of the estimation algorithm is measured through the number of sources estimated for different cardinalities of the monitor set, as well as through the probability of correct detection. Tests are performed on all four network topologies considered: tree graph, random geometric graph, small-world network, and scale-free network. Moreover, other parameters are varied when performing the tests, such as the maximum cardinality of the set of potential sources (the maximum number of sources we can estimate). In addition, the evaluation considers different number of rumors used in the spreading simulation, for the derivation of the simulated probabilities of infection.

Enhancement 1. Sensor Confidence Levels and Adaptive Connectivity Index

The following section describes the evaluation of the source detection algorithm, based on the assignment of a confidence level to each of the monitor nodes available, and ranking them according to this. Measurements from one sensor at a time are considered, provided the number of estimated sources is within the desired limits (the minimum cardinality of the set of estimated sources is set by the user). The enhancement is also using the optimal connectivity index, derived in order to increase the accuracy of the theoretic probability, for the specific network topology and parameters given.

In the following we present the results obtained using various metrics to calculate the confidence levels of the sensors.

Enhancement 1.1: Initial Method for the Calculation of Sensor Confidence Levels

The sensor confidence levels are initially evaluated based on the following criteria:

a. *Criterion 1*: The estimated distance should be lower than $d_{estimated}^{MAX}$.
b. *Criterion 2*: The minimum mean-square error of monitor node i should be lower than the maximum error ε_S.
c. *Criterion3*: The error between the theoretical and sensor measured probabilities should be lower than the error between the theoretical and the average sensor measured probabilities, i.e. $\epsilon_1 = P_{theoretic} - V_{individual}$, $\epsilon_2 = P_{theoretic} - V_{average}$ and $\epsilon_1 \leq \epsilon_2$.

The results obtained are described below.

Small-world Network

The graph below shows the cardinality of the set of estimated sources, against the number of monitor nodes available, for a small-world network of size $N = 200$ nodes and a spreading of 20 rumors. Each subplot illustrates the detection results, when using only monitor nodes at a certain maximum distance. In the figure below, this maximum distance ranges from $d = 1$ to $d = 5$, while the maximum distance of a monitor is $d = 11$. The colouring represents the detection probability, ranging from 0 (red) to 1 (green).

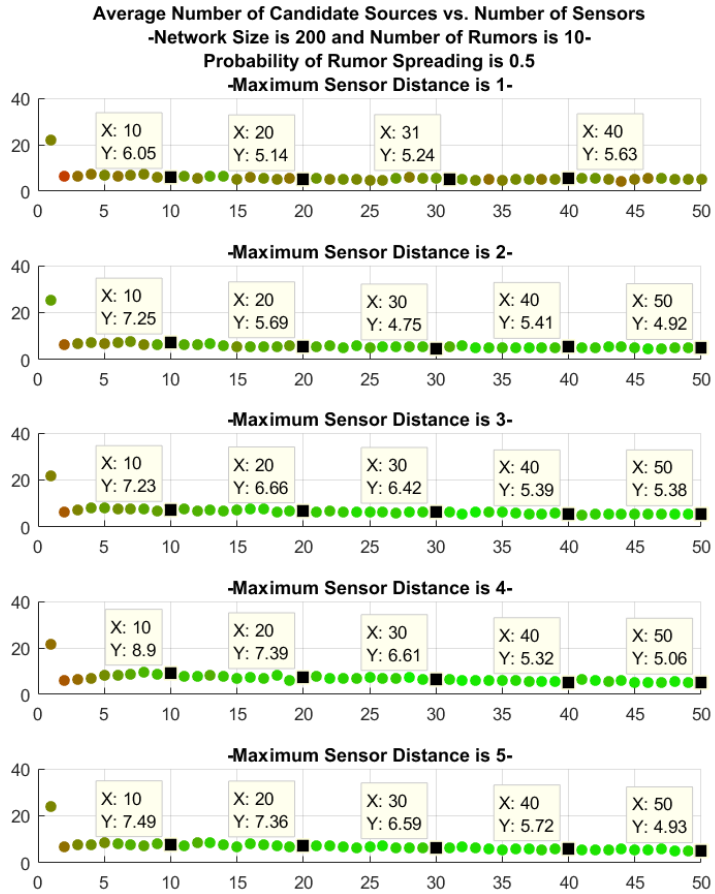


Figure 38: Average Number of Estimated Sources against Number of Available Monitor Nodes, Small-world Network with N=200

Taking into account the fact that the sensors for which the estimated distance is small provide more accurate results, while on the other hand, their number is limited and this might cause erroneous estimation of the source, the following approach is used to improve the performance of the algorithm, as already described in the *Implementation* chapter above. In this sense, for each values of d_{sensor}^{MAX} (maximum sensor estimated distance, used in *Criterion 1* when assigning the sensor confidence levels), a set of estimated sources is computed. The union of all the sets corresponding to each maximum sensor distance is then found, representing the final set of potential sources.

As we can see from the results below, the detection accuracy significantly improves when using the union of all the sets of estimated sources. For example, when the maximum sensor distance is d_{sensor}^{MAX} =3, the probability of correct estimation is $P = 0.71$ using 10 monitors (5% of the network size), while the probability increases to $P = 0.83$ when using the union of all the sets of candidate sources, for the same number of monitors.

We should also note that as expected, the cardinality of the set of potential sources is higher when the union of all the sets is taken. This represents a trade-off with the higher probability of correct detection.

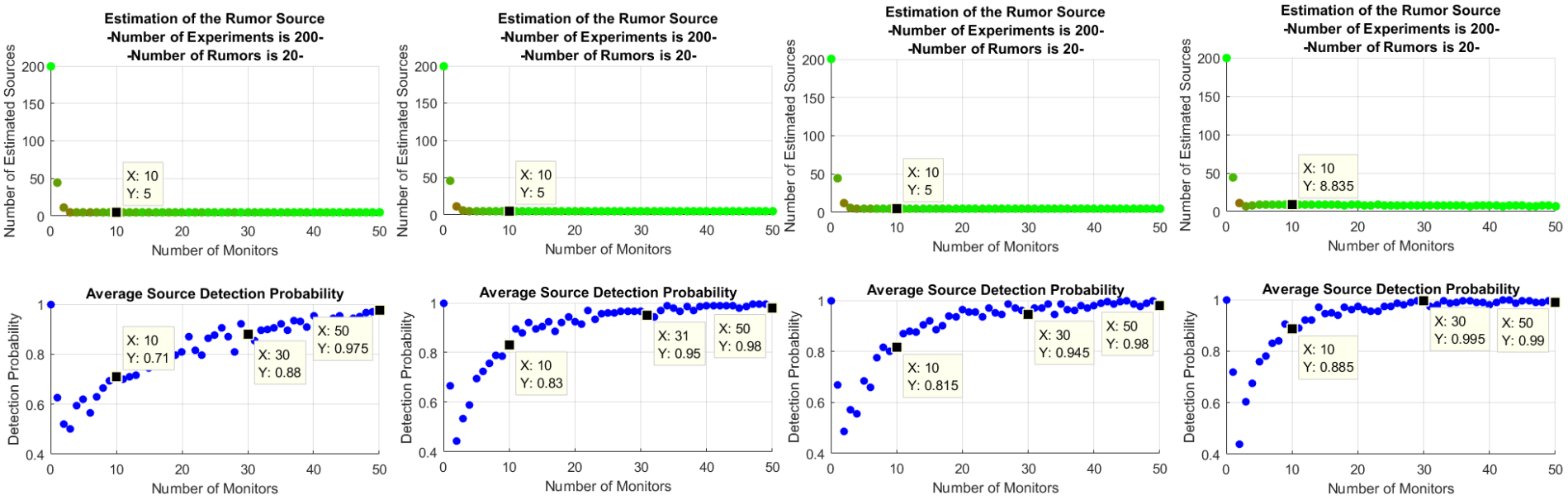


Figure 39: Best Detection Probability (from left to right), for Sensor Maximum Distance d=3, d=5, d=9, and using the Union of the Set of Estimated Sources

Enhancement 1.2: Refined Method for the Calculation of Sensor Confidence Levels

The sensor confidence levels are calculated based on the following criteria, and a detailed calculation of the confidence level is given in the *Implementation* chapter of this report.

a. *Criterion 1*: The estimated distance should be at most $d_{estimated}^{MAX}$.
b. *Criterion 2*: The number of occurrences of *A* should be large, where *A* is defined as the event when the sensor measurement at time k corresponding to a monitor at estimated distance d should be larger than the theoretic probability at time k and distance $d+1$ and lower than the theoretic probability at time k and distance $d-1$. In other words, $V_{sensor}(k,d) < P_{theoretic}(k,d-1)$ and $V_{sensor}(k,d) > P_{theoretic}(k,d+1)$.

The experiment below was performed using 20 rumors to simulate an information dissemination in a small-world network of $N = 200$ nodes. As already discussed, the union method increases the detection accuracy, from $P = 0.77$ when $d_{sensor}^{MAX} = 3$ to $P = 0.91$, in the case where only 10 monitor nodes are available (5% of the network size). In the case of 30 monitors available (15% of network), the probability of correct detection increases from $P = 0.95$ to $P = 0.99$. Nevertheless, the union of the sets of candidate sources has a bigger cardinality as expected, and this represents a trade-off with the higher correct detection probability. In the graphs below we can also notice the reduced detection accuracy when having a very large $d_{sensor}^{MAX} = 9$. These results agree to the ones obtained from the evaluation of the algorithm for estimation of the shortest paths in the section above. There we have seen that the probability of error increases as the monitor's distance from the source becomes larger.

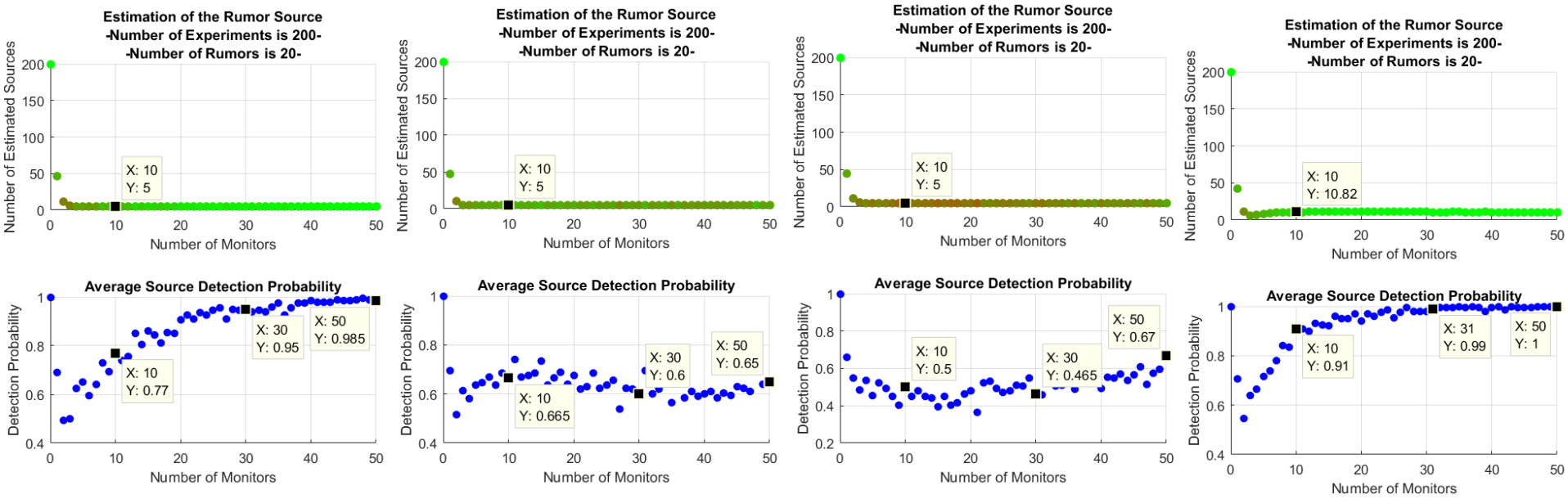


Figure 40: Best Detection Probability (from top left to bottom right), for Sensor Maximum Distance d=3, d=5, d=9, and using the Union of the Set of Estimated Sources

Enhancement 2. Source Rumor Centrality

The following section describes the evaluation of the source detection algorithm, based on assigning a rumor centrality level to each the sources in the final set of candidate sources, and ranking them accordingly. The detection probability will be computed for different numbers of monitor nodes available, by reducing the cardinality of the set of candidate sources to a fixed value, and selecting the sources according to their rumor centrality level.

The improved algorithm also includes *Enhancement 1,* assigning confidence levels to sensor nodes and ranking them accordingly, as well as computing an optimal *connectivity index* to ensure high accuracy of the theoretic probability.

The different enhancements described below differ through the method of calculating the rumor centrality assigned to each potential source.

Enhancement 2.1: Initial Method for Rumor Centrality Calculation

This algorithm calculates the rumor centrality of a candidate source based on the infection of each of the monitor nodes available. In this sense, the rumor centrality will be the sum of the estimated distances between the monitor nodes and the source, provided that the average infection at time step $k = d + 1$ is positive, where d is the estimated shortest path between the sensor and the source. This approach is motivated by the fact that a potential source is more likely to have started the rumor if the distance to the monitor nodes which become infected after a short period of time is small on average. Therefore, the rumor centrality is calculated as follows:

$RC = \sum_{i,V_i(d_i+1)>0.5} d_i$, where d_i is the estimated shortest path between the sensor and the source and V_i is the measured probability of node i having the rumor.

Small-world Network

The plots below show the detection probability and detected number of sources, for a spreading of 20 rumors, in a small-world network, of size $\mathrm{N} = 200$ nodes.

In the first set of subplots, we can notice the reduced performance of the detection algorithm, as a result of the reduced set of candidate sources. Nevertheless, we observe that for a number of candidate sources fixed to $\mathrm{N_S} = 1$, the probability of correct detection has a value $\mathrm{P} > 0.9$, as the number of monitors increases above $\mathrm{M} = 30$ nodes (which represents 15% of the network size).

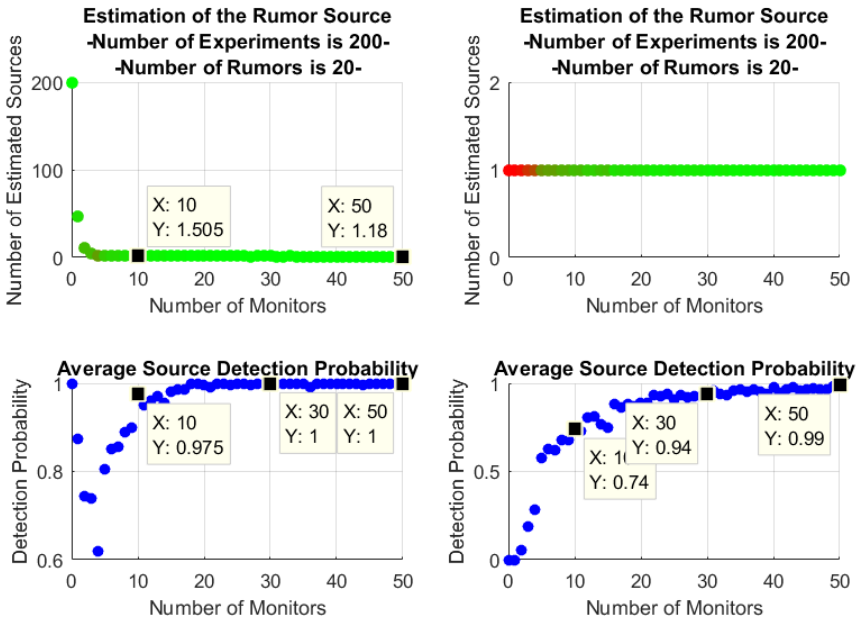


Figure 41: Best Detection Probability and Number of Estimated Sources for Set Cardinality of Minimum 1 Source and Constant Source Set Cardinality equal to 1

In the subplots below, we can see that as the minimum cardinality of the set of potential sources increases (before rumor centrality algorithm is applied), the detection probability decreases. This is a result of the rumor centrality calculation, which may not rank the candidate sources correctly, if the number of nodes to be ranked is larger than 2.

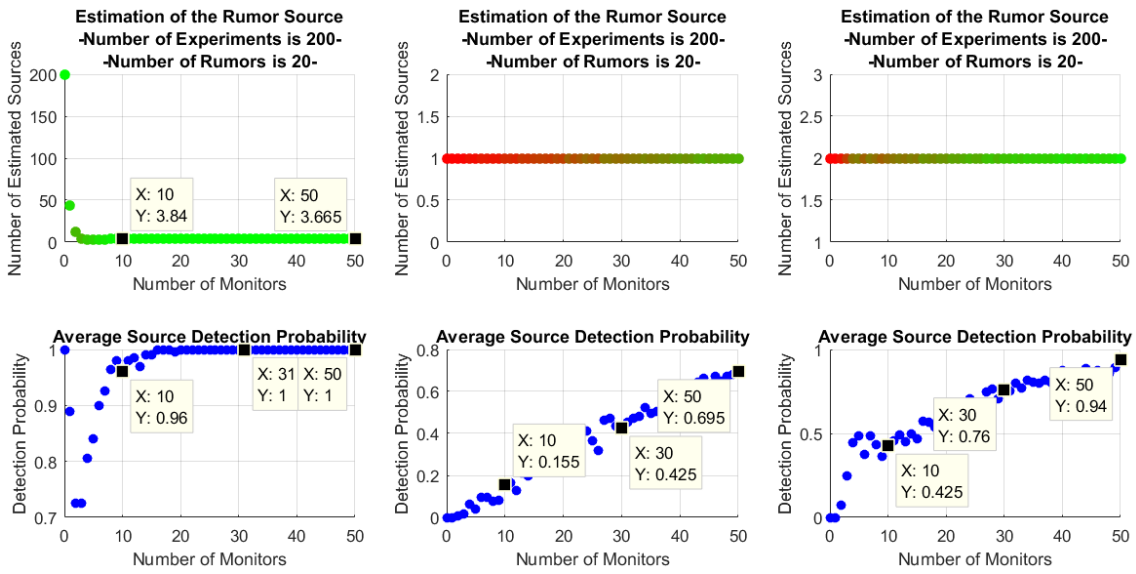


Figure 42: Best Detection Probability and Number of Estimated Sources for Set Cardinality of Minimum 2 Sources and Constant Source Set Cardinality equal to 1, and2

Therefore, we can conclude the following results. The rumor centrality method could accurately rank the nodes in the set of candidate sources, in order to obtain a small number of potential sources, which are very likely to have started the rumor. The detection accuracy of the algorithm including the source rumor centrality enhancement is higher if a set of minimum $N_S = 1$ source is found using the algorithm with *Enhancement 1* (before rumor centrality ranking is applied), followed by a reduction of this set to exactly $N_S = 1$ using *Enhancement 2.*

We should also note that the calculation of the rumor centrality could be modified as follows:

$RC = \sum_{i,V_i(d+1)>x} d_i$, where the parameter x could be adjusted to ensure a higher probability of correct detection.

Enhancement 2.2 Modification of Estimation of the Set of Candidate Sources

This algorithm involves a change to the method of creating the set of candidate sources, based on measurements from the sensors ranked according to their confidence levels. The strategy of creating the set of potential sources is described in the *Implementation* chapter as *Strategy 2,* and summarized below.

For each sensor node i, we compute how many nodes will be eliminated from the set of candidate sources, based on measurements from this sensor. As a result, we are able to determine the number of remaining potential sources, if we were to consider the measurements provided by monitor i. If this number is greater than the minimum cardinality set by the user, then we perform the estimation based on measurements from node i. Otherwise, we discard these measurements and the set of potential sources remains as before. For example, suppose the minimum cardinality of the set of potential sources is $C = 10$. There are currently $N_S = 15$ sources in this set. However, if we consider the measurements from node i, there would be another $N_S = 9$ sources eliminated, which would bring the set of candidate sources to $N_S = 6$. This is lower than $C = 10$ and hence, the measurements from node i will be disregarded and the number of potential sources remains at $N_S = 15$ sources.This is repeated for all the monitor nodes, by considering more confident nodes first.

As we can see from the results below, there is no significant improvement in the detection probability, compared to the previous algorithm. For example, as seen in the subplots below, with a minimum cardinality of the set of sources equal to $N_s = 1$ (cardinality before the enhancement is applied), and with exactly $N_s = 1$ source estimated using the rumor centrality method, the previous algorithm would give a detection probability of $P = 0.74$ using 10 monitors, compared to $P = 0.69$ given by the current algorithm for the same number of monitors. Using 30 monitors, the previous detection probability is $P = 0.94$, compared to of $P = 0.945$ from the current algorithm.

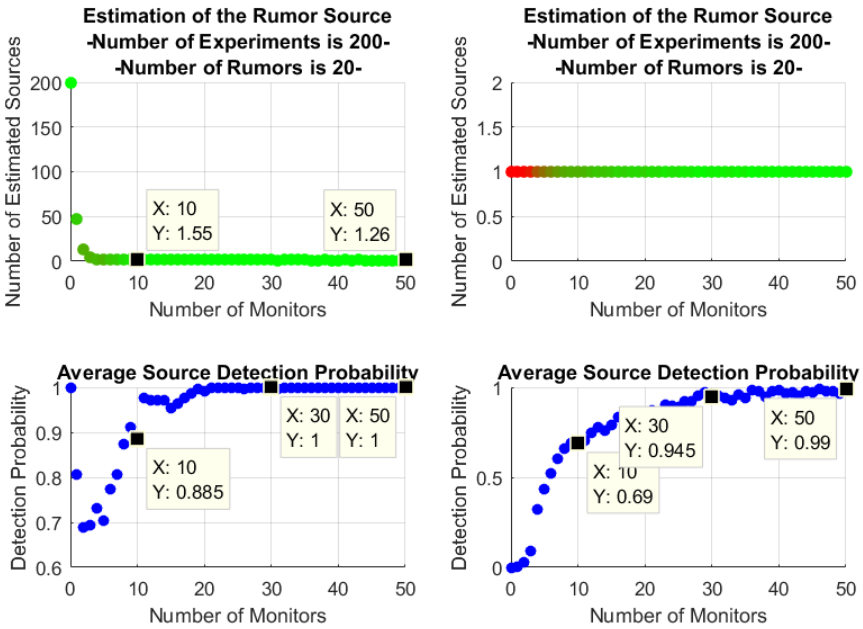


Figure 43: Best Detection Probability and Number of Estimated Sources for Set Cardinality of Minimum 1 source and Constant Source Set Cardinality equal to 1

From the subplots below we can see that the detection accuracy decreases as the minimum cardinality of the set of sources increases, as a result of the rumor centrality calculation.

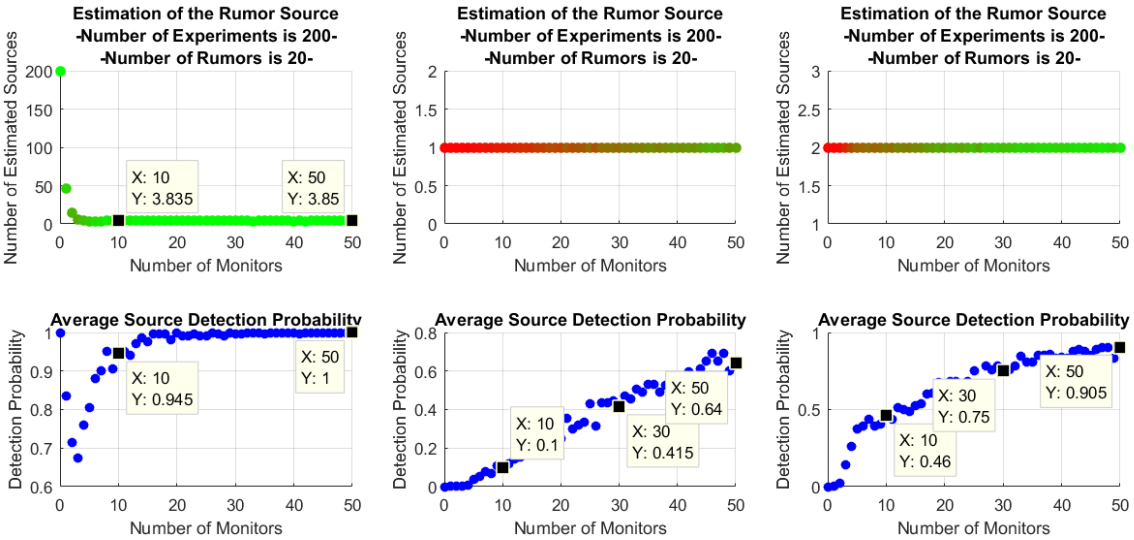


Figure 44: Best Detection Probability and Number of Estimated Sources for Set Cardinality of Minimum 2 sources and Constant Source Set Cardinality equal to 1 and 2

Enhancement 2.3: Refined Method for Rumor Centrality Calculation

As before, the algorithm calculates the rumor centrality of a candidate source based on the infection of each of the monitor nodes available, using the same strategy of creating the source of potential source (see *Strategy 2* in *Implementation* chapter). In this case however, the calculation for the rumor centrality will be different. This is described in the *Implementation* sections and summarized below:

$$RC = \sum_{i \epsilon Set1} d_i + \sum_{i \epsilon Set2} Inf(\infty) + \sum_{i \epsilon Set3} 1000 + \sum_{i \epsilon Set4} 2000 + \sum_{i \epsilon Set5} 3000$$

In the above summation, the individual sets of each individual sum are:

$Set\ 1 = \{node\ i\ |V_i(d_i + 1) > 0, V_i(d_i) = 0\}$
$Set\ 2 = \{node\ i\ |V_i(d_i) \neq 0\}$
$Set\ 3 = \{node\ i\ |V_i(d_i + 1) = 0, V_i(d_i + 2) \neq 0\}$
$Set\ 4 = \{node\ i\ |V_i(d_i + 1) = 0, V_i(d_i + 2) = 0, V_i(d_i + 3) \neq 0\}$
$Set\ 5 = \{node\ i\ |V_i(d_i + 1) = 0, V_i(d_i + 2) = 0, V_i(d_i + 3) = 0\}$

The results below were obtained by simulating a spreading of rumors in a small-world network of $N = 200$ nodes, with average vertex degree $V = 6$ and rewiring probability $\beta = 0.2$.

In the first set of subplots, the minimum cardinality of the set of sources (before *Enhancement 2.3* is applied) is $C = 1$, in the second set $C = 2$, while in the third set $C = 5$. We can notice a clear improvement in the detection accuracy compared to the previous enhancement. For example, using the previous algorithm, the probability of correct detection was $P \cong 0.7$,when using a number of monitors equal to 10 and when the final set of candidate sources is $C = 1$. Nevertheless, the current enhancement ensures a high probability of detection in this case, of $P \cong 0.92$. Furthermore, as seen in the second set of subplots below, when using 10 monitor nodes (5% of network size), the probability of correct detection is $P = 0.975$ when the final set of candidate sources contains $N_S = 2$ nodes, increasing to $P = 1$ using 30 monitors (15% of network size).

Finally, we should note that the index of the source node is randomly chosen and that several experiments were performed for various choices of the source node. Therefore, the location of the source does not have an impact on the performance of the detection algorithm.

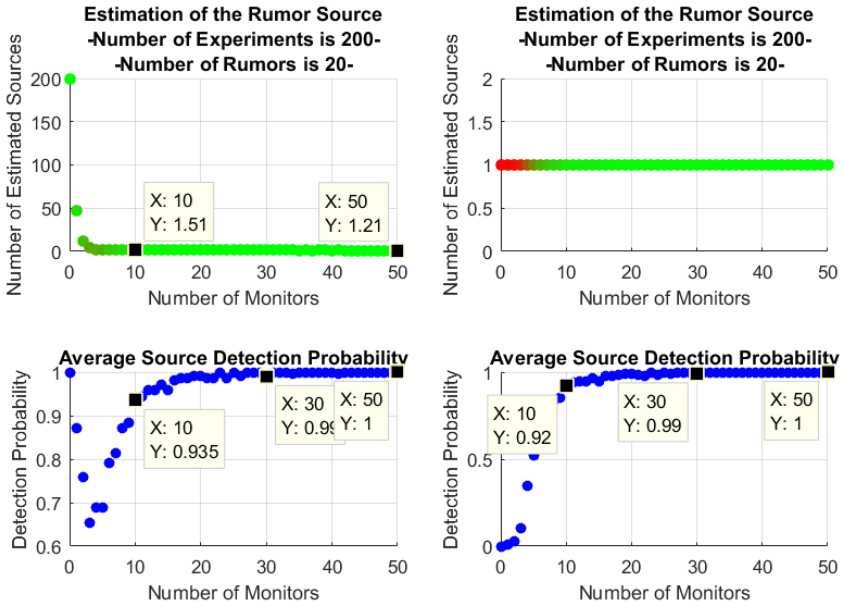


Figure 45: Best Detection Probability using Enhancement 2.3, for a Cardinality of the Candidate Sources of Minimum 1 (left) and Exactly 1 (right)

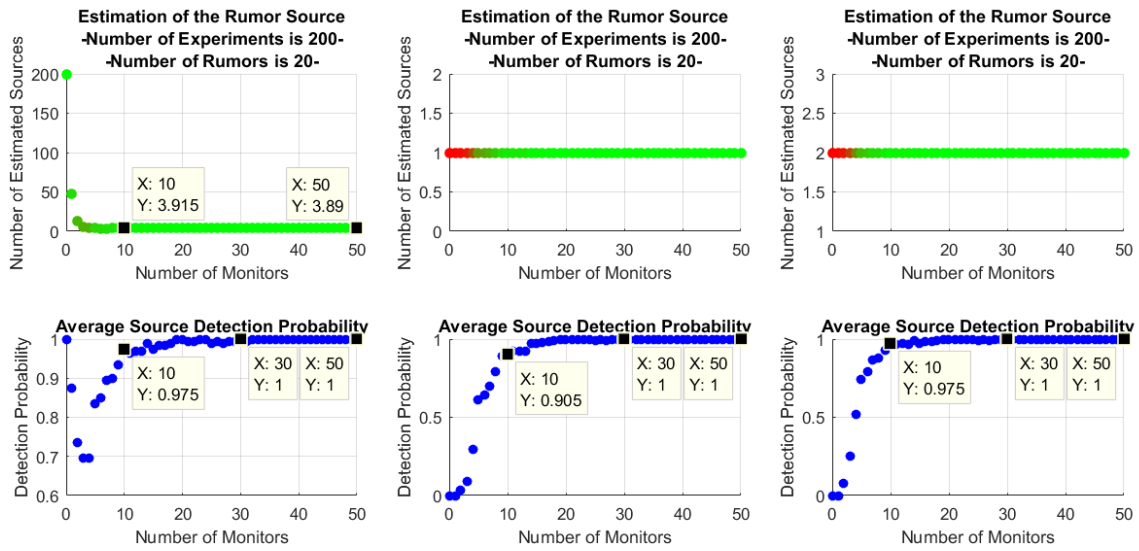


Figure 46: Best Detection Probability using Enhancement 2.3, for a Cardinality of the Candidate Sources of Minimum 2 (left) and Exactly 1 (middle), and 2 (right)

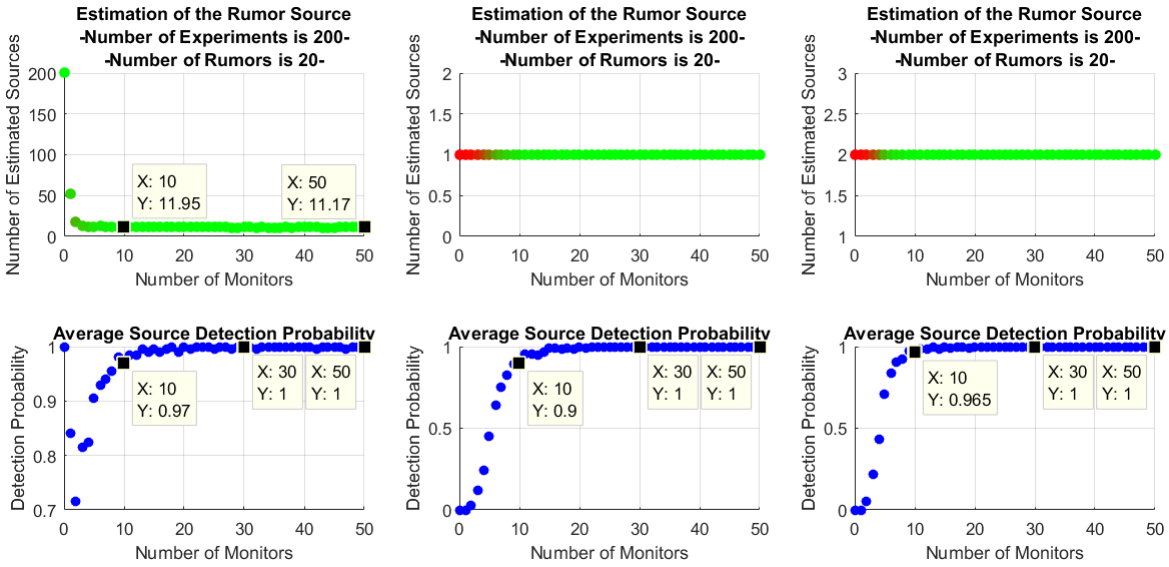


Figure 47: Best Detection Probability using Enhancement 2.3, for a Cardinality of the Candidate Sources of Minimum 5 (left) and Exactly 1 (middle), and 2 (right)

Enhancement 2.4: Further Modification of the Estimation of the Set of Candidate Sources

As already explained in the *Implementation* chapter, In order to account for the errors that could occur as a result of using less confident sensors (as in Strategy 1 implemented in Enhancement 2.3 above), the following method can be used to create the set of candidate sources.

For each sensor node i, we compute how many nodes will be eliminated from the set of candidate sources, based on measurements from this sensor. As a result, we are able to determine the number of remaining potential sources, if we were to consider the measurements provided by monitor i. If this number is greater than the minimum cardinality set by the user, then we consider the estimation based on measurements from node i. Otherwise, we discard these measurements and the set of potential sources remains as before.
Once we discard a sensor from the set of ranked monitors, we will not consider any other sensors which are less confident than the sensor we have discarded.
This method will ensure more accurate detection. Nevertheless, it leads to a large cardinality of the set of candidate sources, before the rumor centrality method is applied, which could lead to big computational complexity of the rumor centrality algorithm.

The results below were obtained by simulating a rumor spreading in a small-world network, of size $N = 200$ nodes, where the minimum cardinality of the set of candidate sources is $C = 1$ (first set of subplots), and $C = 2$ (second set of subplots).
Firstly, we can notice that the cardinality of the set of sources using *Strategy 2* is much larger compared to the case when using *Strategy 1.* As seen in the second set of subplots below, when the minimum cardinality is set to $C = 2$, the number of estimated sources becomes very large and increases with larger number of monitors. This is because, as the number of monitors increases, it is more likely that we observe the actual rumor source and that would imply eliminating all the other nodes besides itself, leading to a set of candidate sources which has a single source and hence, which has a cardinality lower than the minimum set as $C = 2$.
Furthermore, the detection accuracy does not improve for a larger cardinality of the set of potential sources and therefore, it is not the preferred strategy for the detection algorithm, due to its increased complexity.

Finally, we should note that the index of the source node is randomly chosen and that several experiments were performed for various choices of the source node. Therefore, the location of the source does not have an impact on the performance of the detection algorithm.

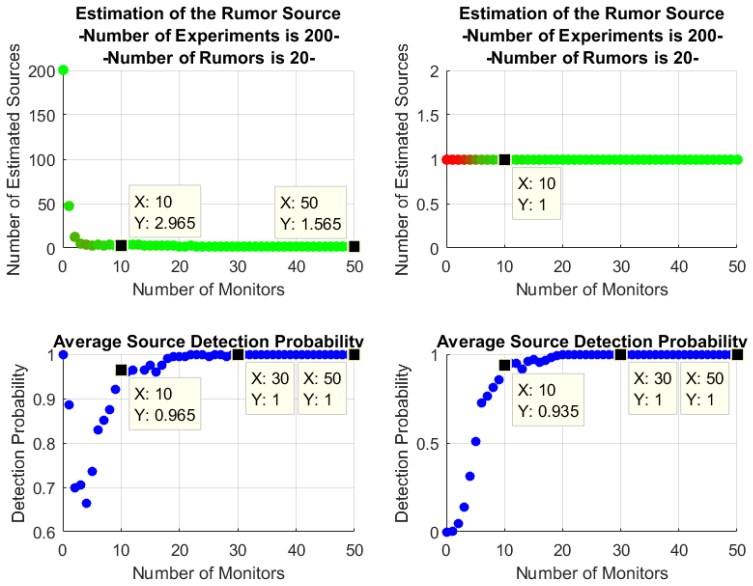


Figure 48: Best Detection Probability in a Small-world Network, using Enhancement 2.4, for a Cardinality of the Candidate Sources of Minimum 1 (left) and Exactly 1 (right)

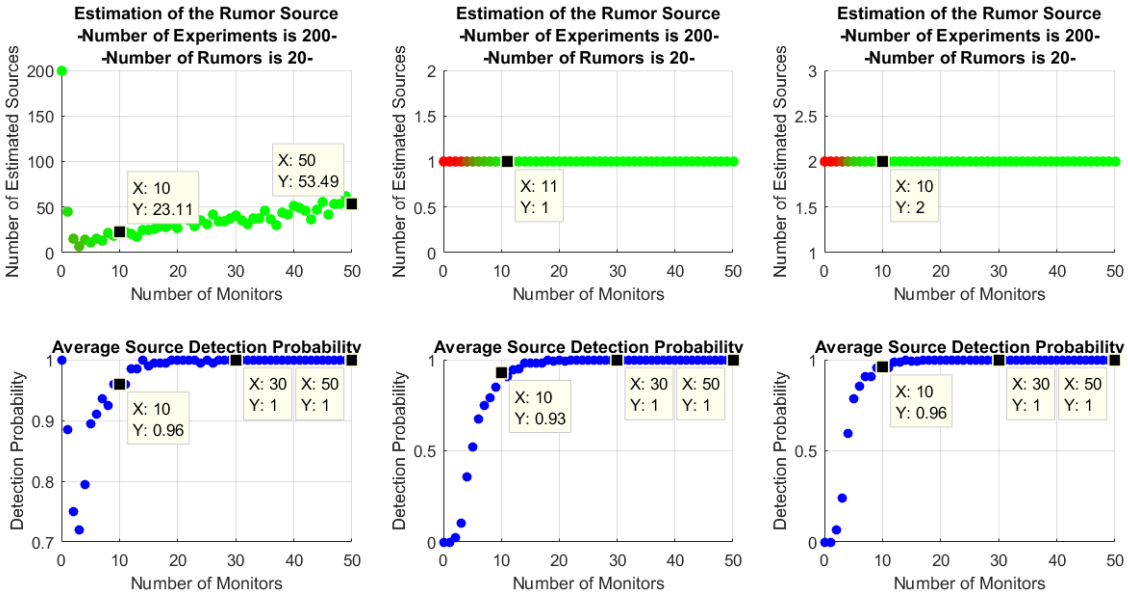


Figure 49: Best Detection Probability using Enhancement 2.4, for a Cardinality of the Candidate Sources of Minimum 2 (left) and Exactly 1 (middle), and 2 (right)

Enhancement 2.4: Illustration of the Set of Detected Sources

The plots below show the vertices and edges in a small-world network of $N = 200$ nodes. In all subplots, the source of rumors is highlighted through a bigger circle.

The left subplot shows the probability of detecting any of the network nodes, using *Enhancement 2.4* of the estimation algorithm. As we can see, the actual source is the most likely node to be detected, using 10 monitors, with a probability of detection $P > 0.8$.

The middle subplot shows the probability of detecting each of the network nodes, in the case when the real source node is not correctly detected, and the right subplot is a more detailed representation of the middle subplot. As we can see, the most likely nodes to be detected in the case of wrong estimation are located 1 hop away from the real rumor source. The next likely nodes are in a neighbourhood of the real source, and the probability of detection of a node decreases as its distance from the source increases.

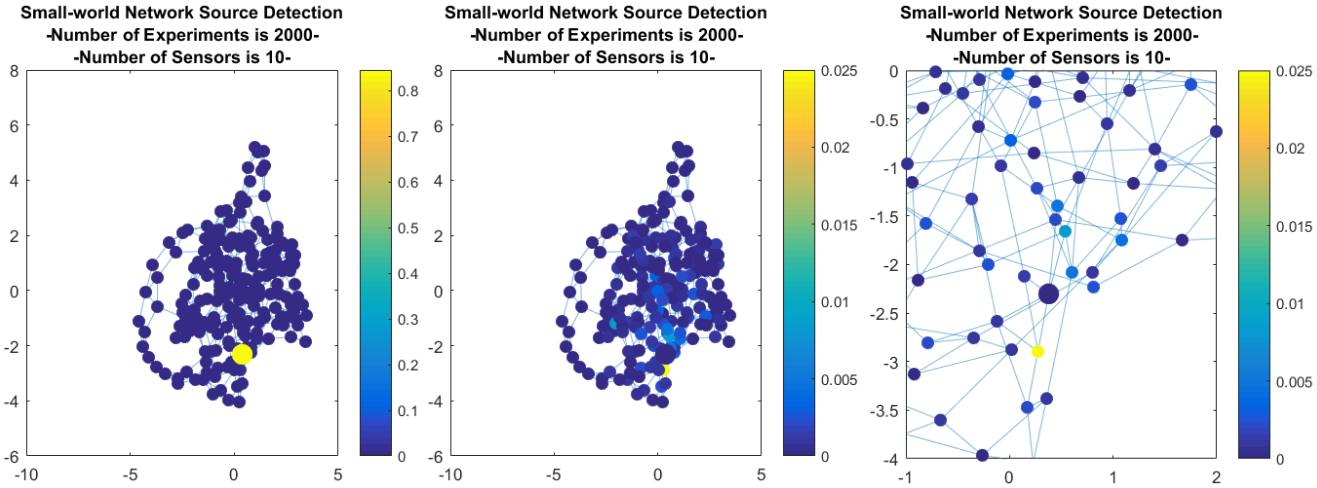


Figure 50: Illustration of Probability of Detection of all the Nodes in a Small-world Network of size N=200, using 10 Monitors

The same results have been plotted, by observing only 5 monitor nodes. Although the probability of correct detection is lower compared to the example above, if the correct source is not correctly estimated, the most likely nodes to be considered potential sources are located 1 hop away from the real source.

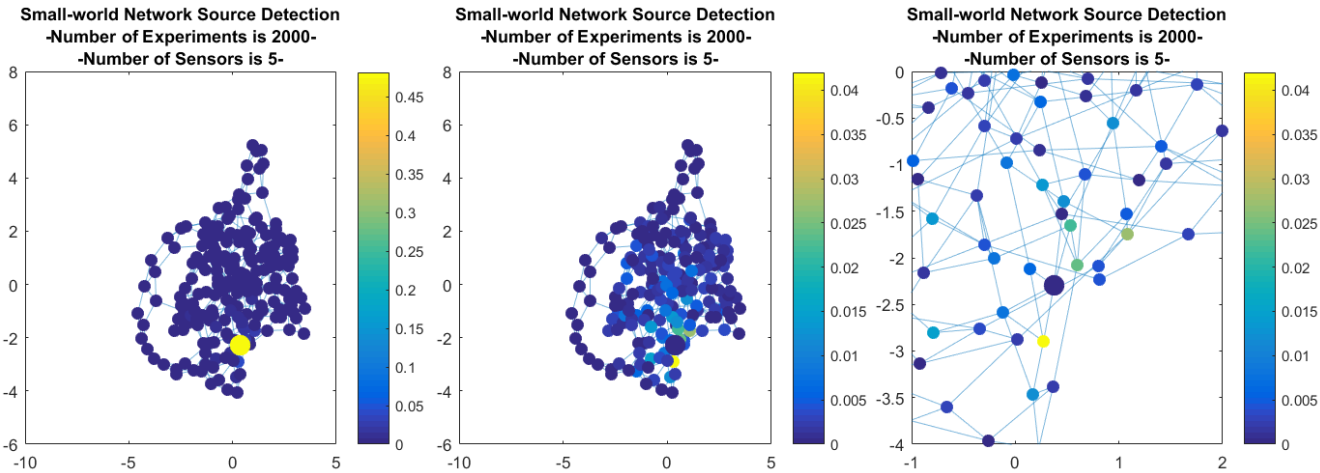


Figure 51: Illustration of Probability of Detection of all the Nodes in a Small-world Network of size N=200, using 5 Monitors

Enhancement 2.4: Different Network Characteristics

The plots below show the detection probability of the rumor source in a small-world network of size $N = 200$ nodes, for an average vertex degree $V = 4$, and a number of rumors $R = 50$ rumors. We can see that compared to the case when only $R = 20$ rumors are available (Figure 48), the detection accuracy improves, particularly for the case when the set of candidate sources is reduced to $N_S = 1$ source (right subplot). The reason for this could be the improvement in the sensor measurements when the average infection probability is derived from a larger number of rumors. As we can see in the left subplot, the set of candidate sources is lower compared to the case when 20 rumors are available (Figure 48, left subplot). As a result, the source rumor centrality method will provide a more accurate ranking of the potential sources.

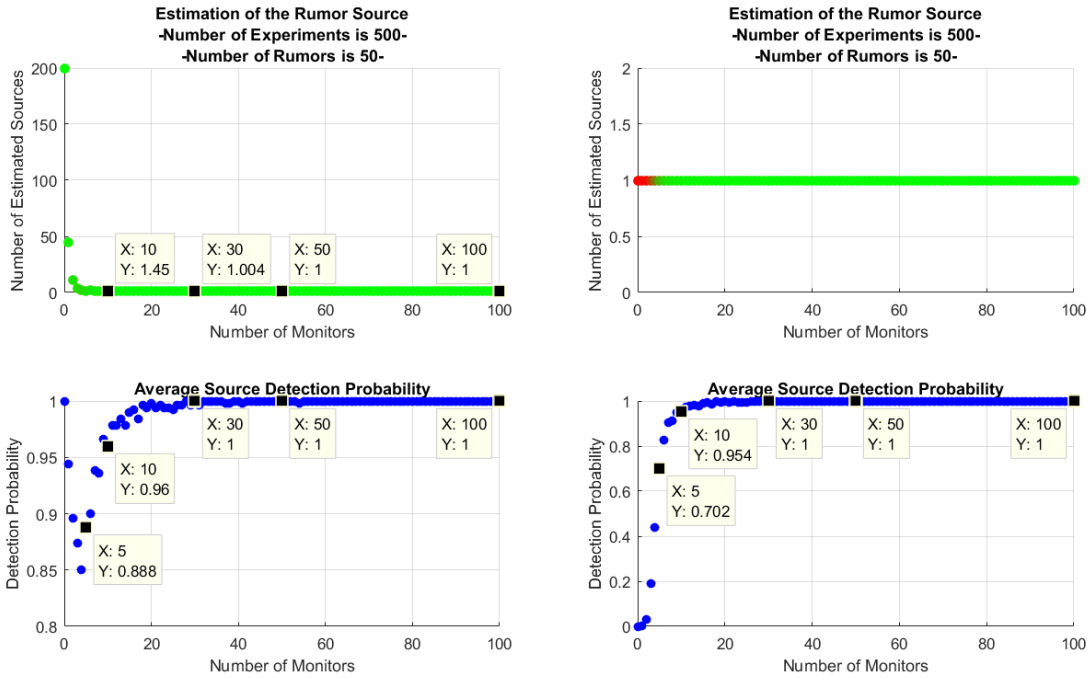


Figure 52: Best Detection Probability in a Small-world Network with Average Vertex Degree V=4, using Enhancement 2.4, for a Cardinality of the Candidate Sources of Minimum 1 (left) and Exactly 1 (right)

The plot below shows the probability of correct detection of the rumor source, in a small-world network of $N = 200$ nodes and average vertex degree $V = 10$. We can see that the vertex degree does not significantly impact the estimation algorithm. Only for the case when only 10 monitor nodes are available (5% of the network), there is a larger decrease in the detection probability. This result could be due to the fact that in a small-sized network with a high vertex degree, most nodes will be located very close to the source and therefore, the rumor infection happens with a bigger intensity, compared to the model predicted by the theoretical probability. Nevertheless, while in social networks there may be nodes with very large degree (hubs), there are also many nodes with few connections and therefore, the results obtained for a large average vertex degree and small network radius might not be representative for a real-life application.

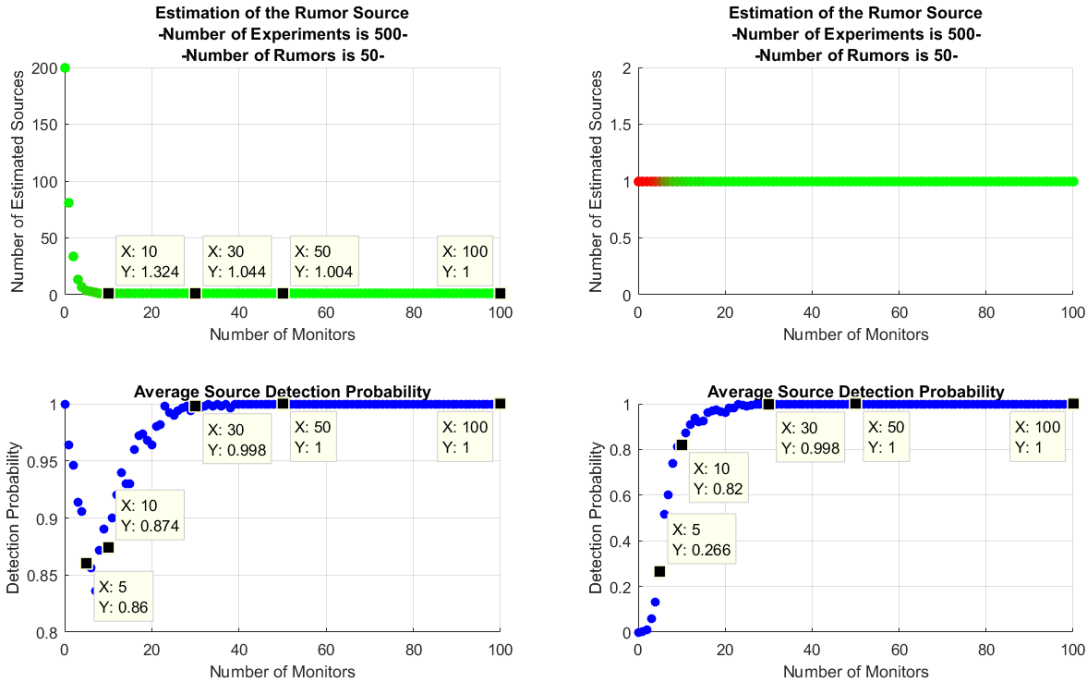


Figure 53: Best Detection Probability in a Small-world Network with Average Vertex Degree V=10, using Enhancement 2.4, for a Cardinality of the Candidate Sources of Minimum 1 (left) and Exactly 1 (right)

Evaluation of Final Algorithm on All Network Topologies

Tree Graph

The results below reflect the source detection performance in a tree graph of $N = 156$ nodes, with $C = 5$ number of children for each node, and $D = 4$ depth. The index of the source is chosen as $i = 156$, in order to best evaluate the rumor centrality algorithm.

From the results below we can see that the probability of correct detection (before applying the rumor centrality algorithm) is $P = 1$, for a number of monitors above $M = 30$ (15%). Nevertheless, in this case the set of candidate sources has a cardinality in the interval $[1,2]$. The probability of correctly detecting $N_S = 1$ source using the rumor centrality method is lower compared to the one obtained on other network topologies. This is a result of the fact that the main criterion used in the calculation of rumor centrality is mainly accounting for errors in the calculation of the shortest distances, while in the case of a tree topology, the accuracy of detection of the shortest paths is higher compared to other topologies, and the erroneous results might be due to other types of noise in the sensor measurements.

The best detection probability is obtained using Enhancement 2.2, as this would ensure the lowest cardinality of the set of potential sources, before the rumor centrality method is applied.

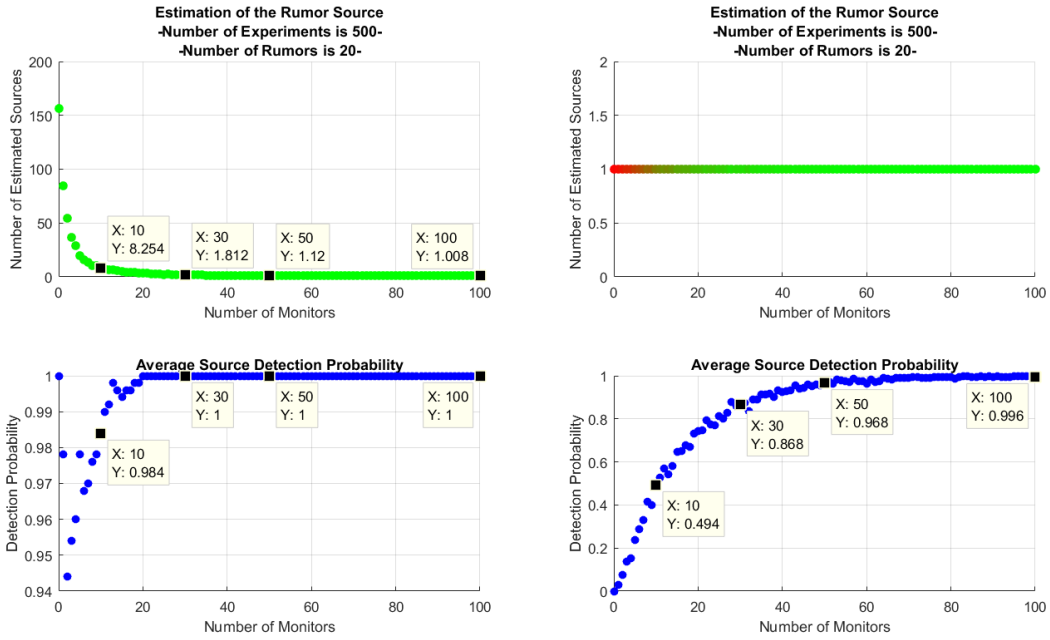


Figure 54: Probability of Correct Detection in a Tree Graph with N=, C=, D=, using Enhancement 2.3 and Cardinality of the Set of Potential Sources with Minimum 1 (left) and Exactly 1 Source (right)

The probability of distance error is shown in the subplots below, for different number of monitoring nodes.

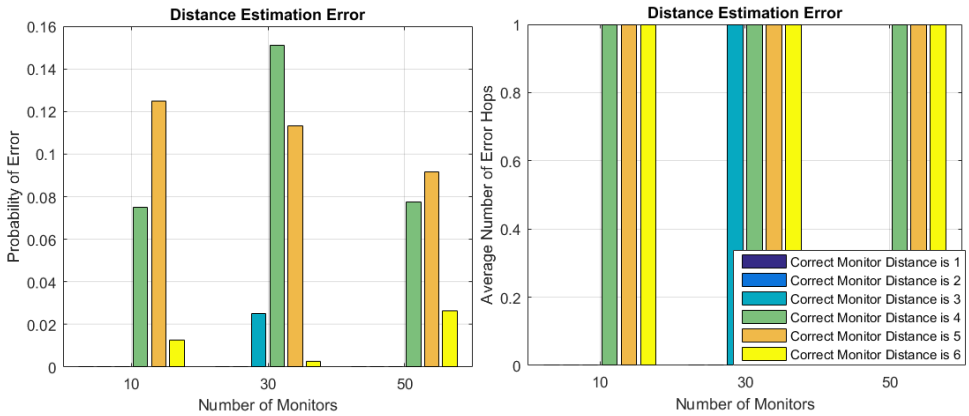


Figure 55: Distance Estimation Error in a Tree Graph of N=156 Nodes

Random Geometric Graph

The results below reflect the source detection performance in a random geometric graph of $N = 200$ nodes, with connectivity radius $R = 0.1$, and a grid dimension equal to $D = 1$. The index of the source is chosen as $i = 200$, in order to best evaluate the rumor centrality algorithm.

The best detection probability is obtained using Enhancement 2.4 and we can see that the accuracy is mostly limited by the shortest path estimation and sensor confidence assignment methods. On the other hand, the rumor centrality algorithm is able to reduce the number of candidate sources to $N_S = 1$, without degrading the detection performance.

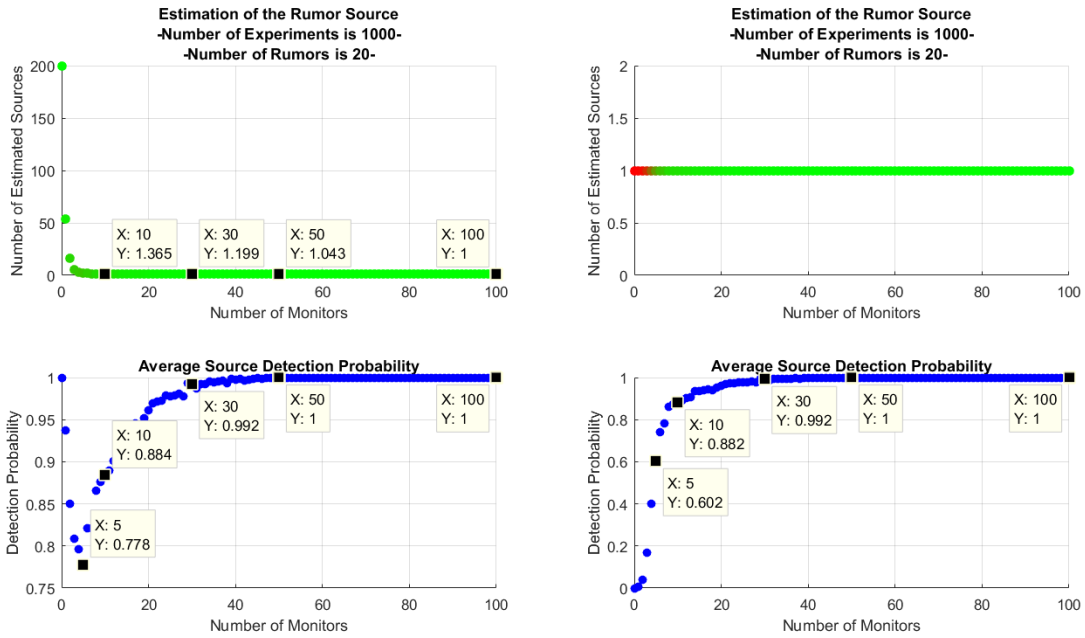


Figure 56: Probability of Correct Detection in a Random Geometric Graph with N=200 and R = 0.2, using Enhancement 2.4, and Cardinality of the Set of Potential Sources with Minimum 1 (left) and Exactly 1 Source (right)

Small-world Network

The results below reflect the source detection performance in a small-world network of $N = 200$ nodes, rewiring probability $\beta = 0.2$, and average vertex degree is $V = 4$. The index of the source is chosen as $i = 200$, in order to best evaluate the rumor centrality algorithm.

As in the case of a random geometric graph, the best detection probability is obtained using Enhancement 2.4 and the accuracy is mostly limited by the shortest path estimation and sensor confidence level assignment methods. On the other hand, the rumor centrality algorithm is able to reduce the number of candidate sources to $N_S = 1$, without degrading the detection performance.

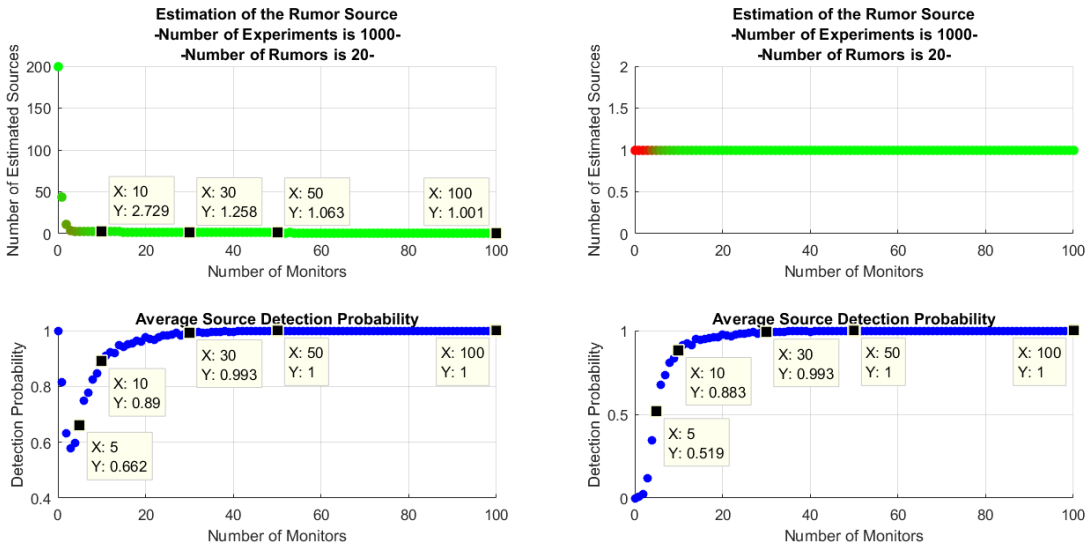


Figure 57: Probability of Correct Detection in a Small-world Network with N=200 and beta = 0.2, using Enhancement 2.4, and Cardinality of the Set of Potential Sources with Minimum 1 (left) and Exactly 1 Source (right)

Random Network

A random graph has been obtained using the Watts-Strogatz algorithm, for $N = 200$ nodes, and with rewiring probability $\beta = 1$. The results below reflect the source detection performance using Enhancement 2.4, in a random graph where the index of the source is chosen as $i = 200$, in order to best evaluate the rumor centrality algorithm.

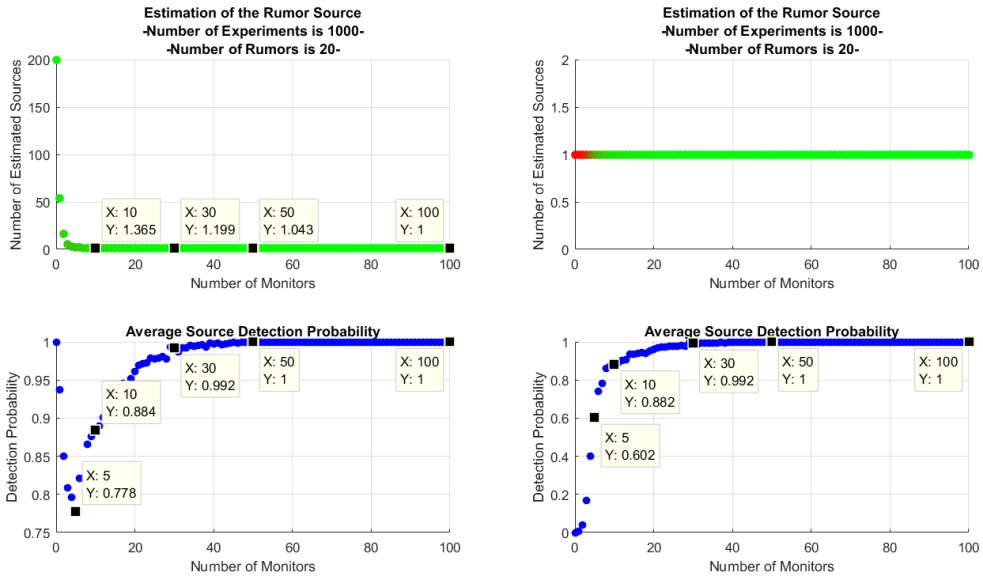


Figure 58: Probability of Correct Detection in a Random Network with N=200, using Enhancement 2.4, and Cardinality of the Set of Potential Sources with Minimum 1 (left) and Exactly 1 Source (right)

Scale-free Network

A scale-free network has been obtained using Method I presented in the *Implementation* chapter of this report. The results below reflect the source detection performance using Enhancement 2.4, in a scale-free network of $N = 200$ nodes, where the index of the source is chosen as $i = 200$, in order to best evaluate the rumor centrality algorithm.

In this case, the detection accuracy is lower compared to the previous network topologies. The reason for this may be the fact that scale-free networks have a highly heterogeneous degree distribution and thus contain many high-degree nodes, as well as nodes with very few link connections. Therefore, the measurements at monitoring nodes might significantly vary between nodes with very high and nodes with very low degree. Large deviations of these measurements from their expected value may lead to more erroneous results and a wrong estimation of the source.

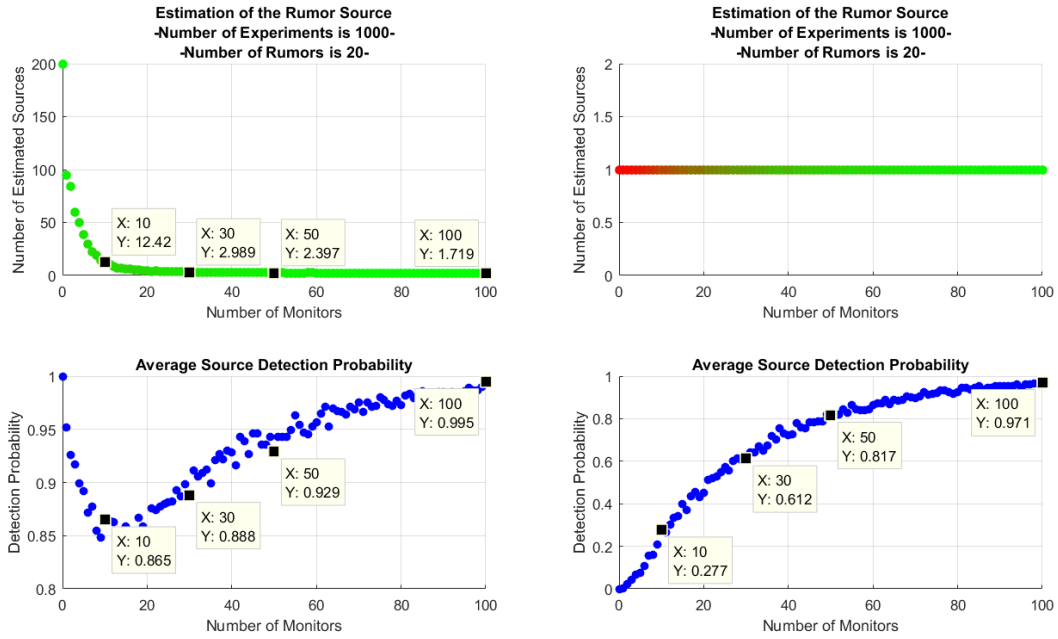


Figure 59: Probability of Correct Detection in a Scale-free Network with N=200, using Enhancement 2.4, and Cardinality of the Set of Potential Sources with Minimum 1 (left) and Exactly 1 Source (right)

Algorithm Complexity

The complexity of the algorithm is dominated by the Dijkstra algorithm, which has the following running time. In a graph with E edges and V vertices, the simplest implemention of Dijkstra's algorithm gives the running time $O(\mathrm{E} + \mathrm{V}^2) = O(\mathrm{V}^2)$. However, for sparse graphs, more efficient algorithms could give a time complexity of $O(\mathrm{E} + \mathrm{V}\log\mathrm{V})$.

Other computationally expensive parts of the algorithm might be the following.

Firstly, the estimation of the *connectivity index* used in the theoretical formulation for the probability of infection requires the simulation of a spreading of rumors on the network. Nevertheless, even a small number of rumors used in this simulation leads to accurate results. Furthermore, the calculation of the optimal index requires the employment of the minimum mean square error estimation method, based on samples of the theoretic and simulated probability at different time steps and for different distances. This could hence lead to high complexity for a large number of time steps or a wide range of distances. Nevertheless, as the range of distances considered is typically narrow (as the radius of a social network is typically small) and the number of time steps is small (as the probability of infection quickly saturates to its maximum after a short delay from the rumor initiation), the time complexity of this part of the algorithm is reduced. In addition, for a network which does not evolve in time, the parameters of the theoretical formula need to be estimated only once as they will remain constant over time.

Secondly, the estimation of the shortest distances from each monitor to the source requires the implementation of the minimum mean-square error method. Nevertheless, as the range of possible distances is reduced (due to small radius of a social network), and since we are observing only a small fraction of the network nodes, this method should not significantly reduce the speed of the algorithm.

Chapter 6

Summary of Results

This chapter will firstly summarize the state-of-the-art solutions to the problem of estimating a single rumor source. The results obtained using the algorithm presented in this report will then be discussed and compared to the state-of-the-art.

State-of-the-Art

Some of the state-of-the-art main results obtained for random geometric graphs and trees are the following.

In [5] the authors propose a rumor centrality method used to rank the potential rumor sources. The evaluation is performed on a tree graph and the results show that the detection probability of the rumor source estimator is approximately $P = 0.9$, for a network size of $N < 100$ nodes, decreasing to $P \cong 0.2$ for a size of $N = 400$ nodes. In both cases, the parameter of the regular tree is $\alpha = 0$. When the parameter $\alpha = 1,2,3, or\ 4$, the probability of correct detection is $P \in [0.9,1)$. In addition, the frequency of an estimator error equal to $e = 1\ hop$ is approximately 80%.

In [11] the authors propose a maximum *a posteriori* estimator to identify the rumor source using a single observation of all the nodes in the network. The evaluation of the method is performed on a regular tree graph of 1000 nodes, assuming there is access to a set of suspects. The results show that when the set of suspects has cardinality $k = 2$, the detection probability is $P \cong 0.55$ for a node degree of $\delta = 3$, increasing to $P \cong 0.95$, for a larger node degree of $\delta = 20$.

In [12] a pseudo-likelihood function for the source is used to estimate the source of rumors, and the evaluation of the method shows that if we observe 5% of the network size, the probability that the source is within the first top 10 ranked (ranking based on the pseudo-likelihood function) is $P \cong 0.5$, increasing to $P \cong 0.82$ if we observe 30% of the network, and a similar value if we observe the entire network. The tests are performed on networks of size $N = 100$ nodes, assuming a constant spreading probability within the network.

Some of the main results obtained for random, small-world and scale-free networks are the following.

In [1] the authors describe an ML detector used to detect the source of rumors, assuming a susceptible-infected spreading model and using multiple observations of the entire network. The results obtained show that for a scale-free network, the probability of correct detection is $P < 0.1$ when using a single observation, increasing to $P \cong 0.9$ when having access to five observations. For a small-world network, the probability of correct detection is $P < 0.2$ when having access to five observations of the entire network. The tests are performed on networks of size $N = 10000$ nodes.

In [10] the authors describe a rumor centrality and node selection method as a solution to the rumor source detection problem. Evaluation of the method on a random directed graph of 30146 nodes shows that the real source is within the top 10 ranked nodes using the rumor centrality method, when using more than 650 monitors (2.15% of the network size). In addition, the rank of the actual source is around 5 when using 5120 monitoring nodes (16.98% of the network size).

New Approach

We summarize below the main results obtained when evaluating the source detection algorithm used to estimate the source of rumors in a network.

In terms of accuracy of the theoretical formulation for the probability of infection, this converges to the probability obtained by simulating a spreading of a large number of rumors. The theoretical probability will be compared against the simulated probability from a dissemination of multiple rumors (the number of rumors in a real-world scenario is assumed to be $R \geq 20$). The accuracy of the theoretical probability leads to a good estimation of the shortest path between the monitors and the potential source. With most of the errors occurring as a result of the deviation of the individual sensor measurements from the expected value, the distance error probability is very low, particularly for smaller monitor distances. In addition, the average distance error is $|d_e| \cong 1\ hop$.

This observation becomes important when designing the source detection algorithm, in order to account for the cases when the noisy sensor measurements lead to a distance error of $1\ hop$. This is achieved by assigning a confidence level to each sensor, based on its estimated distance to the source, as well as the measurements obtained from the monitors at different time steps. The main criterion of calculating the confidence levels is based on the measurements of the sensor at time equal to d, when the estimated shortest distance to the source is d. If the real distance would be d, then the measurements at time d must be zero. Hence, if these measurements are not zero, it means the estimated distance should be smaller. Considering that the distance error hop is typically $|d_e| \cong 1\ hop$, it is very likely that in this case the distance is $d - 1$ instead of d as initially estimated. Nevertheless, the criterion includes checks at time steps equal to $d - 1$, as well as $d - 2$, to account for potential distance errors of $2\ hops$ and $3\ hops$ respectively.
Based on the sensor measurements ranked according to their confidence levels, a set of candidate sources is obtained. This set is further reduced to a single candidate source, by assigning a rumor centrality level to each potential source and ranking all the sources accordingly. The main criterion used in the calculation of the rumor centrality is based on the sensor measurements at a time equal to d, where d is the shortest distance between the candidate source and the monitor. If these measurements are positive, it means this source could not have started the rumor, as the rumor could only get to the monitor at time $d + 1$.
While the sensor confidence level accounts for the case when the distance error is $d_e = -1\ hop$, the rumor centrality method accounts for the case when the distance error is $d_e = 1\ hop$.

In summary, using the source detection algorithm based on estimation of shortest distance using the theoretical probability formula, sensor confidence level assignment and source rumor centrality ranking, the following results are obtained. For a scale-free network, the probability of correct detection is $P > 0.8$ when observing more than 5% of the network nodes, increasing to $P = 0.92$ when observing 25% of the network nodes, with a set of minimum 1 and maximum 3 candidate sources. For all other network topologies considered, the probability of correct detection is $P > 0.8$ when observing 5% of the network nodes, increasing to $P > 0.99$ when observing 15% of the network, and $P = 1$ when observing 25% of the network nodes, with a set of exactly 1 candidate source.

The table below summarizes the results obtained in the *Evaluation* section, when simulating a rumor spreading in a small-world network of size $N = 200$ nodes. We should note that the values below are only an approximation of the typical results that could be obtained in a small-world network of different characteristics, different source nodes etc. Moreover, the results are obtained using 200 repeated experiments to obtain an average performance of the detection algorithm.

Small-World Network, of size $N = 200$ nodes			Cardinality of the Set of Detected Sources				
			$C = 5$	$C = 2$	$C = 1$	$C > 5$	Notes
Enhancement 1.1	Number of Monitors	$M = 10$ 5%	$P = 0.83$	n/a	n/a	$P = 0.885$	The cardinality of the set of candidate sources when $C > 5$ is in the interval $[8,10]$, for the particular network size.
		$M = 30$ 15%	$P = 0.95$	n/a	n/a	$P = 0.995$	
		$M = 50$ 25%	$P = 0.98$	n/a	n/a	$P = 99$	
Enhancement 1.2	Number of Monitors	$M = 10$ 5%	$P = 0.77$	n/a	n/a	$P = 0.91$	The cardinality of the set of candidate sources when $C > 5$ is in the interval $[10,15]$, for the particular network size. There is an improvement in the detection probability, when $C > 5$, for a slight increase in the cardinality of potential sources.
		$M = 30$ 15%	$P = 0.95$	n/a	n/a	$P = 0.99$	
		$M = 50$ 25%	$P = 0.985$	n/a	n/a	$P = 1$	
Enhancement 2.1	Number of Monitors	$M = 10$ 5%	n/a	n/a	$P = 0.74$	n/a	The cardinality of the set of candidate sources before the enhancement is applied is in the interval $[1,2]$, for the particular network size. This is further reduced to exactly 1 candidate source using Enhancement 2.1. There is a great improvement in the detection probability, in particular when $C = 1$.
		$M = 30$ 15%	n/a	n/a	$P = 0.94$	n/a	
		$M = 50$ 25%	n/a	n/a	$P = 0.99$	n/a	
Enhancement 2.2	Number of Monitors	$M = 10$ 5%	n/a	n/a	$P = 0.69$	n/a	The cardinality of the set of candidate sources before the enhancement is applied is in the interval $[1,2]$. This is further reduced to exactly 1 candidate source using Enhancement 2.2. There is no significant improvement in the detection accuracy compared to the algorithm above.
		$M = 30$ 15%	n/a	n/a	$P = 0.945$	n/a	
		$M = 50$ 25%	n/a	n/a	$P = 0.99$	n/a	
Enhancement 2.3	Number of Monitors	$M = 10$ 5%	n/a	$P = 0.975$	$P = 0.905$	n/a	There is an improvement in the detection probability, particularly for the case when the final set of candidate sources contains only one node. At the same time, there is an increase in the cardinality of the set of sources before the rumor centrality is applied, i.e. $C \in [3,4]$, however this should not significantly impact the complexity of the algorithm.
		$M = 30$ 15%	n/a	$P = 1$	$P = 1$	n/a	
		$M = 50$ 25%	n/a	$P = 1$	$P = 1$	n/a	
Enhancement 2.4	Number of Monitors	$M = 10$ 5%	n/a	n/a	$P = 0.935$	n/a	The cardinality of the set of candidate sources before the enhancement is applied is in the interval $[1,3]$. The detection accuracy improves compared to the previous algorithm, particularly when observing a lower number of monitor nodes.
		$M = 30$ 15%	n/a	n/a	$P = 1$	n/a	
		$M = 50$	n/a	n/a	$P = 1$	n/a	

Table 6: Summary of Detection Probability for Different Algorithm Enhancements

Chapter 7

Conclusions

Future Directions

One future direction would be to further improve the theoretical probability of rumor dissemination, in order to ensure a more precise formula and which could better model various network topologies and parameters.

In addition, it would be interesting to research the problem of spreading of multiple sources, how a multiple source rumor could be modelled and how it impacts the theoretical probability of rumor spreading.

Moreover, some state-of-the-art approaches examine the problem of rumor spreading by assuming varying infection probabilities at each node in the network. This would be a more realistic assumption than the one where the probability of spreading is constant throughout the network. Nevertheless, this model has increased complexity and the derivation of an exact probability formula for the rumor dissemination process may be challenging.

Furthermore, the current development assumes a susceptible-infected model. Nevertheless, in a more realistic scenario, some nodes could recover from the rumor information in time (for example, some persons might forget the information or might realise the rumor is false and not spread it further). Therefore, analysing the susceptible-infected-recovered model could be useful for real-world applications.

In what the sensor measurements are concerned, future development should also consider the problem of rumor source detection, through observations in a fixed time window, at some unknown time after the initial spreading. This would be more suitable for some real-life applications, where the start time of the spreading might be unknown.

In terms of evaluation of the algorithm, simulations and tests have only been carried out on synthetic data. Hence, future work should include testing on real networks, to best assess the performance of the detection algorithm. Moreover, various noise scenarios should be developed to ensure the algorithm is robust to false information obtained from some of the sensor nodes, loss of information etc.

Concluding Remarks

The goal of this project was to successfully infer the source responsible for spreading of data within a social network, based on multiple observations at some of the nodes in the network. The state-of-the-art solutions to this problem are based on the ideal assumption that there is access to time snapshots of the entire network. Moreover, the evaluation of these solutions is mostly performed on simple topologies such as trees or random geometric graphs, and the results show that the probability of correct detection is generally smaller than one.

This project addresses the problem of localizing a single diffusion source in a social network, based on several time measurements at some randomly selected nodes, and assuming multiple rumor *attacks* from the same source. The result of this project provides a source detection algorithm based on a novel approach of estimating the theoretical probabilities of rumor infection of a node, as a function of its distance to the rumor source. The theoretical rumor dissemination probabilities are compared against measurements at some randomly selected monitoring nodes in the network, which are obtained by simulating a spreading of multiple rumors. The results are used to obtain an estimation of the shortest paths between the monitors and the rumor source, based on the minimum mean-square error method. The source detection algorithm relies on the estimation of the shortest distances between the monitors and the source, as well as sensor confidence level assignment and source rumor centrality ranking.

The evaluation of this algorithm is performed on simple topologies such as tree graph, and random geometric graph, as well as topologies that accurately model the characteristics of a social network such as small-world and scale-free network. The results obtained are the following. For a scale-free network, using a simulation of 20 rumors the probability of correct detection is $P > 0.8$ when observing more than 5% of the network nodes, increasing to $P = 0.92$ when observing 25% of the network nodes, with a set of minimum 1 and maximum 3 candidate sources. For all other network topologies considered, the probability of correct detection is $P > 0.8$ when observing 5% of the network nodes, increasing to $P > 0.99$ when observing 15% of the network, and $P = 1$ when observing 25% of the network nodes, with a set of exactly 1 candidate source. Moreover, the probability of correct detection increases as the number of rumors becomes larger. Furthermore, in the case of wrong estimation the detected source is typically located 1 hop away from the real rumor source, using any number of monitoring nodes.

Future research directions would further increase the performance of the algorithm and enable it to be more robust to noisy sensor measurements. As a result, the accuracy of the source detection algorithm could increase, particularly when this is employed in a scale-free network. Last but not least, the current probability formulation and source detection algorithm could be further developed in order to provide a solution for the estimation of multiple diffusion sources in a network.

Bibliography

[1] Z. Wang, W. Dong, W. Zhang and C. W. Tan, “Rooting our Rumor Sources in Online Social Networks: The Value of Diversity From Multiple Observations,” *Selected Topics in Signal Processing, IEEE Journal,* vol. 9(4), no. 663-77.

[2] S. Kwon, M. Cha, K. Jung, W. Chen and Y. Wang, “Prominent Features of Rumor Propagation in Online Social Media,” *Data Mining (ICDM), 2013 IEEE 13th International Conference ,* no. 1103-1108, 2013.

[3] J. Murray-Bruce and P.-L. Dragotti, “Estimating Localized Sources of Diffusion Fields Using Spatiotemporal Sensor Measurements,” *IEEE Transactions,* vol. 63(12), no. 3018-31, 2015.

[4] R. Alexandru, “Final Year Project Interim Report”.

[5] D. Shah and T. R. Zaman, “Rumors in a Network: Who's the Culprit,” *Information Theory, IEEE Transactions,* vol. 57(8), no. 5163-5181.

[6] T. Z. D. Shah, “Rumor Centrality: A universal source detector,” *ACM SIGMETRICS Performance Evaluation Review,* vol. 40, no. 1, pp. 199-210, 2012.

[7] B. A. Prakash, J. Vrekeen and C. Faloutsos, “Spotting Culprits in Epidemics: How many and Which ones?,” *Data Mining (ICDM), 2012 IEEE 12th International Conference ,* pp. 11-20, 2012.

[8] N. Karamchandani and M. Franceschetti, “Rumor Source Detection under Probabilistic Sampling,” *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium,* pp. 2184-2188, 2013.

[9] A. Y.Lokhov, M. Mezard, H. Ohta and L. Zdebovora, “Inferring the origin of an epidemic with a dynamic message-passing algorithm,” *Physical Review E,* vol. 90(1), no. 012801, 2014.

[10] E. Seo, P. Mohaptra and T. Abdelzaher, “Identifying Rumors and Their Sources in Social Networks,” *SPIE defense, security, and sensing. International Society for Optics and Photonics,* pp. 83891I-83891I, 2012.

[11] W. Dong, W. Zhang and C. W. Tan, “Rooting out the Rumor Culprit from Suspects,” *Information Theory Proceedings (ISIT),* pp. 2671-2675, 2013.

[12] A. Agaskar and Y. M. Lu, “A fast Monte Carlo algorithm for source localziation on graphs,” *SPIE Optical Engineering+ Applications,* pp. 88581N-88581N, 2013.

[13] N. Fedwa, E. Krause and A. Sisson, “Spread of A Rumor,” *Society for Industrial and Applied Mathematics. Central Michigan University,* no. 25, 2013.

[14] T. Zhu, Z. Fu and B. Wang, “Epidemic dynamics on complex networks,” *Progress in Natural Science,* vol. 16(5), no. 452-7, 2006.

[15] A.-L. Barabási, Network Science: The Scale-Free Property.

[16] S. H. Reuven Cohen, “Scale-Free Networks Are Ultrasmall,” *Physical review letters,* vol. 90(5), no. 058701, 2003.

[17] T. Konstantopoulos, “Markov Chains and Random Walks. Lecture notes,” 2009.

[18] C. Ling, “Probability and Stochastic Processes. Lecture Notes,” 2016.

[19] P. Snell and P. Doyle, “Random walks and electric networks,” *Free Software Foundation,* 2000.

[20] A.-L. Barabasi, E. Ravasz and T. Vicsek, “Deterministic scale-free networks,” *Physica A: Statistical Mechanics and its Applications,* vol. 299(3), no. 559–64, 2001.

[21] D. Easley and J. Kleinberg, Networks, crowds, and markets: Reasoning about a highly connected world, Cambridge University Press, 2010 .

[22] T. Taahashi and N. Igata, “Rumor detection on twitter,” *Soft Computing and Intelligent Systems (SCIS) and 13th International Symposium on Advanced Intelligent Systems (ISIS), 2012 Joint 6th International Conference,* pp. 452-457, 2012.

[23] R. Toivonen, J.-P. Onnela and K. Kaski, “A model for social networks,” *Physica A: Statistical Mechanics and its Applications,* vol. 371(2), no. 851-60, 2006.

[24] D. Wang and M. T. Amin, “Using Humans as Sensors: An Estimation-theoretic Perspective,” *Proceedings of the 13th international symposium on information processing in sensor networks,* pp. 35-46, 2014.

[25] Z. Wang and C. W. Tan, “On Inferring Rumor Source for SIS Model under Multiple Observations,” *Digital Signal Processing (DSP), 2015 IEEE International Conference,* pp. 755-759, 2015.

[26] Z. Wang, W. Dong, W. Zhang and C. W. Tan, “Rumor Source Detection with Multiple Observations: Fundamental Limits and Algorithms,” *ACM SIGMETRICS Performance Evaluation Review,* vol. 42, no. 1, 2014.

Appendices

Appendix A. Matlab Environment

Generation of a small-world network

```
%Author: Roxana Irina Alexandru (ria12)
%Email: rialexandru01@gmail.com
%University: Imperial College London
%Description: Generation of a Small-world network
%Input parameters:
%1.Number of Nodes : N
%2.Average vertex degree (2K) : K
%3.Rewiring probability: beta
%Outputs:
%1. Adjancency matrix M
%2. Matrix of shortest distances D

ccc;
%The number of nodes is N
N=200;

%Node degree is 2K
K=2;

%Rewiring probability
beta=0.2;

%Select the index of the source node
src=1;

%Create a matrix describing the graph structure

%h=WattsStrogatz(N,K, beta);
M=WattsStrogatz(N,K,beta);

D=zeros(N,N);
%Find all shortest paths between any two nodes
parfor i=1:N
    for j=1:N
        D(i,j)=dijkstra(M,i,j);
    end
end
```

Watts-Strogatz Algorithm

```
%Author: Roxana Irina Alexandru (ria12)
%Email: rialexandru01@gmail.com
%University: Imperial College London
%Description:
% H = WattsStrogatz(N,K,beta) returns a Watts-Strogatz model graph with N
% nodes, N*K edges, mean node degree 2*K, and rewiring probability beta.
% beta = 0 means a ring lattice, and beta = 1 means a random graph.
function [ M ] = WattsStrogatz( N,K,beta )


%Adjacency Matrix
M=zeros(N,N);

% Connect each node to its K next and previous neighbors. This constructs
% indices for a ring lattice.
%Create matrix of K repeated rows
%On each row
s = repelem((1:N)',1,K);
t = s + repmat(1:K,N,1);
t = mod(t-1,N)+1;

% Rewire the target node of each edge with probability beta
%The source is the current node we are looking at
for source=1:N
    switchEdge = rand(K, 1) < beta;

    newTargets = rand(N, 1);
    %Avoid self-loops
    newTargets(source) = 0;
    %Look at the previous K neighbors of source and set the connection to 0
    %i.e. get the indices for which t=source
    newTargets(s(t==source)) = 0;
    %Find the connections to the next K neighbors and set some of them to
    %0, according to probability Beta
    newTargets(t(source, ~switchEdge)) = 0;
    %The last 2K nodes in the ind vector are the next and previous neighbors
    %Which do not need to be re-wired
    [~, ind] = sort(newTargets, 'descend');
    %nnz returns the number of non-zero matrix elements
    %For those neighbors for which switchEdge=1, i.e. which need to be
    %rewired, create a new edge to other 2 random nodes
    t(source, switchEdge) = ind(1:nnz(switchEdge));
end

for i = 1:N
    for k = 1:K
    j=t(i,k);
    M(i,j)=1;
    M(j,i)=1;
    end
end
```

Generation of a tree graph

```
%Author: Roxana Irina Alexandru (ria12)
%Email: rialexandru01@gmail.com
%University: Imperial College London
%Description: Function which creates a tree network
%Inputs:
%1. Number of children of each node: C
%2. Depth of tree: D
%Outputs: Adjancency matrix adjMatrix
function [N, adjMatrix ] = treeNetwork( D, C )

N=1;
X=1;
%Number of nodes
for i=1:(D-1)
    X=X*C;
    N=N+X;
end

M=zeros(N,N);

%Select the index of the source node
src=1;

%Connect the source node 1
for c=2:(C+1)
    M(1,c)=1;
    M(c,1)=1;
end

for k=2:(D-1)
    lower=((C^(k-1)-1)/(C-1))+1;
    upper=(C^(k)-1)/(C-1);
    const=C^(k-1);
    for i=lower:upper
        for j=1:C
        e=lower+const+(i-lower)*C+j-1;
        M(i,e)=1;
        M(e,i)=1;
        end
    end
end


adjMatrix=M;
end
```

Generation of a scale-free network using Barabasi algorithm

```
%Author: Roxana Irina Alexandru (ria12)
%Email: rialexandru01@gmail.com
%University: Imperial College London
%Description: Function which generates a scale-free network using the
%Barabasi algorithm, with fixed degree distribution
%Inputs: Number of nodes (must be a power of 3)
%Outputs: Adjancency network M
function [ M ] = SFBarabasiModel( N )
M = zeros(N,N);

%Pick a random initial node, the root of the graph
root = 1;

%Number of iterations
L = (log(N)/log(3)) ;

%Initial unit has no connections and a single node
A = zeros(1,1);

for l = 1:L
    P = A;
    sprev = 3^(l-1)+1;
    snew = 3^l;

    %Update the unit
    A = zeros(snew,snew);
    for i = 1:sprev-1
        for j = 1:i;
            A(i,j) = P(i,j);
            %ensure symmetric
            A(j,i) = A(i,j);
        end
    end

    for i = sprev : 2*3^(l-1)
        for j = sprev:i
            A(i,j) = P(i-sprev+1,j-sprev+1);
            A(j,i) = A(i,j);
        end
        %Last 2^(k-1) new nodes should be connected to the root
        if i>= (2*3^(l-1)-2^(l-1)) && i<=(2*3^(l-1))
        A(i,root) = 1;
        A(root,i) = 1;
        end
    end

    for i = 2*3^(l-1)+1:snew
        for j = 2*3^(l-1)+1:i
            A(i,j) = P(i-2*3^(l-1),j-2*3^(l-1));
            A(j,i) = A(i,j);
        end
        if i>= (snew-2^(l-1)) && i<=snew
        A(i,root) = 1;
        A(root,i) = 1;
        end
    end
end
M = A;
End
```

Generation of a general scale-free network

```
%Author: Roxana Irina Alexandru (ria12)
%Email: rialexandru01@gmail.com
%University: Imperial College London
%Description: Function which generates a scale-free network
%Inputs:
%1. Number of nodes N
%2. Average degree of seed networks: d
%3. Size of seed network: mo
%Outputs: Adjancency network M
function [ M ] = SFgraph( d, mo, N)
%The average degree of the initial network is d
%Create initial graph of m0 nodes
M = zeros(N,N);
for i = 1:mo
    neighbours = randi(d);
    for n = 1:neighbours
        j = randi(i);
        if j~=i
            M(j,i) = 1;
            M(i,j) = 1;
        else
            M(j,i) = 0;
            M(i,j) = 0;
        end
    end
end
for i = mo+1:N
    degsum = 0;
    deg = zeros(1,i-1);
    p = zeros(1,i-1);
    %Look at all pre-existing nodes
    for j = 1: i-1
        %Calculate degree of node j
        deg(j) = calcNodeDegree(N, M,j);
        degsum = degsum + deg(j);
    end
    for j = 1: i-1
        %Calculate probability of connection to each node j
        p(j) = deg(j)/degsum;
        %Generate a connection with probability p(j) between node j and
        %node i

        %Probability of connection
        prob2 = p(j);

        %Probability of not being infected by rumor
        prob1 = 1-p(j);

        %Probability vector
        prob=[prob1, prob2];

        %Vector with N random values between 0 and 1
        r=rand(1,1);
%         C = cumsum([0, prob]);
%         s = sum( r>=cumsum([0,prob]) );
        randVector=sum( r>=cumsum([0,prob]) )-1;
        M(i,j) = randVector;
        M(j,i) = randVector;
    end
end
end
```

Calculation of the theoretic probability of infection

```
%Author: Roxana Irina Alexandru (ria12)
%Email: rialexandru01@gmail.com
%University: Imperial College London
%Description: This function calculates the theoretical probability of rumor
%infection
%Inputs:
%1. The node distances for which the theoretic probability is calculated:
% from 1 to dmax
%2. The optimal connectivity index: connectIndex
%3. The number of nodes in the network N
%4. The number of time steps K
%5. The probability of rumor spreading Ps
%Output: the theoretic probability of rumor infection Ptheoretic

function [ Ptheoretic ] = calcTheoreticProb( N,K, Ps,dmax, connectIndex)

P = zeros(N, K);
Ptheoretic = zeros(K, dmax);
 %-----------------------THEORETIC PROBABILITIES-----------------------
   %For each distance
   for d = 1 : dmax
       %For each time step
      for nsum = 1 : K-1
          %Calculate the probability of getting first infected at time k
            for k = d : nsum
                %If the distance is not 1, subtract 1 from the time step
                %and distance, as the first step in the path is always a
                %step further away from source (A-type)
                if d~=1
                    n2=k-1;
                    k2=d-1;
                else
                    n2=k;
                    k2=d;
                end ;

                pk=Ps*connectIndex;
                P(k,d) =pk^(k2+1)*((1-pk)^(n2-k2))*(2^(n2+1))/sqrt(2*pi*(n2))*exp(-((n2-2*k2)^2/(2*n2)));

                Ptheoretic(nsum,d)= Ptheoretic(nsum,d)+P(k,d);
            end
            %Truncate the values above 1 to 1
%             if Ptheoretic(nsum,d)>1
%                 Ptheoretic(nsum,d)=1;
%             end
      end
   end

end
```

Calculation of the simulated rumor probability

```
%Author: Roxana Irina Alexandru (ria12)
%Email: rialexandru01@gmail.com
%University: Imperial College London
function [ Psimulated ] = calcSimProb( N, K, dmax, Vtest, src, D, spreadingModel)
Psimulated=zeros(N,dmax);

%-----------------------SIMULATED PROBABILITIES-----------------------

% if strcmp(spreadingModel,'SI')
    %For each distance from the source
    %This calculated the average probability of a node located at that
    %distance to get the rumor after a number of steps
    for d=1:dmax
        no=0;
        %For each node in the network
        for j=1:N
            %Select those nodes which are at distance d from the source
            if D(src,j)== d
                no=no+1;
                %For each step
                for n=1:K
                    %Calculate the average probability obtained through
                    %simulation, for each distance d
                    Psimulated(n,d)=(Psimulated(n,d)*(no-1)+Vtest(n,j))/no;
                end

            end

        end
    end

end
```

Simulation of a Spreading of Rumors

```
%Author: Roxana Irina Alexandru (ria12)
%Email: rialexandru01@gmail.com
%University: Imperial College London

%The number of repeated experiments of spreading of rumors is L
L=5;
%The number of steps is K
K=3;

%The average of all Us is the matrix V
V=zeros(K,N);
Error=zeros(K,N);
%Probability of spreading of rumors
Pspreading=0.5;
Pspreading= 1-Pspreading;


%Select the index of the source node
%src=1;

%Choose the number of sources and
%Choose random indices which will be the sources

findSrc = 0;
noSources = 1;
% indices=randi(N,1,noSources);
for l=1:L
    %Define a matrix which will hold all the vectors corresponding to all the
    %nodes, at each step taken of one iteration
    %Initially the matrix has 0 elements
    U=zeros(K,N);
    %Initialize multiple sources
    u=setMultipleSources(noSources,indices,N);

    %Count number of non-zero elements in u
    count=0;
    for i=1:N
        if u(i)==1
            count=count+1;
        end
    end

    %Put this as the vector corresponding to first step in matrix U
    U(1,:)=u;

    %Spread the rumors
    for k=1:K-1
        u=stepSpreadingNonSelfAvoiding(u, M, Pspreading);
        U(k+1,:)=u;

    end

    %Update the average matrix V
    for v1=1:N
        for v2=1:K
            V(v2,v1)=(V(v2,v1)*(l-1)+U(v2,v1))/l;
            Error(v2,v1)=V(v2,v1)-U(v2,v1);
        end
    end

end
```

Plot of the network nodes and edges

```
%Author: Roxana Irina Alexandru (ria12)
%Email: rialexandru01@gmail.com
%University: Imperial College London

function [  ] = plotStepGraphs( M, V, K )

  figure
  for k = 1:size(V)
      if K >=3
        dim=floor(K/3);
        if mod(K,3) ~=0
            dim1=dim(1)+1;
        else dim1 = dim(1);
        end
        subplot(dim1,3,k)
        graphBuild(M, V(k,:)');

%         title('Average Spreading of Rumors SW')
         if k == 2
             title('Average Spreading of Rumors Small-World')
         end
        ylabel(['Spreading at step ', num2str(k)])
        grid on
      else
        graphBuild(M, V(k,:)');
%         title('Average Spreading of Rumors SW')
        ylabel(['Spreading at step ', num2str(k)])

      end
  end


end
```

Generation of a random vector

```
%Author: Roxana Irina Alexandru (ria12)
%Email: rialexandru01@gmail.com
%University: Imperial College London
%Function which generates a random vector of 0 and 1, given its input size
function [ randVector ] = randVec( N, Pspreading)

%Probability of not being infected by rumor
prob1 = Pspreading;
%Probability of being infected by rumor
prob2 = 1-prob1;

%Probability vector
prob=[prob1, prob2];

%Vector with N random values between 0 and 1
r=rand(1,N);

%Initially, the random vector generated has only 0 elements
randVector=zeros(1,N);

for i=1:N
    randVector(i)=sum(r(:,i)>=cumsum([0,prob]))-1;
end

end
```

Initialization of the rumor source

```
%Author: Roxana Irina Alexandru (ria12)
%Email: rialexandru01@gmail.com
%University: Imperial College London
%Function takes as input the index of the nodes which are sources
%Function sets the sources for spreading of rumours
%and creates a new vector illustrating the initial state of nodes
function [ tinitial ] = setMultipleSources( noSources,indices, N )

tinitial=zeros(1, N);
for i=1:noSources
    index=indices(i);
    tinitial(index)=1;
end


end
```