SCIENTIFIC REPORTS

OPEN

Fast and accurate detection of spread source in large complex networks

Received: 23 May 2017
Accepted: 15 January 2018
Published online: 06 February 2018

Robert Paluch[1], Xiaoyan Lu[2], Krzysztof Suchecki[1], Bolesław K. Szymański[2,3] & Janusz A. Hołyst[1,4]

Spread over complex networks is a ubiquitous process with increasingly wide applications. Locating spread sources is often important, e.g. finding the patient one in epidemics, or source of rumor spreading in social network. Pinto, Thiran and Vetterli introduced an algorithm (PTVA) to solve the important case of this problem in which a limited set of nodes act as observers and report times at which the spread reached them. PTVA uses all observers to find a solution. Here we propose a new approach in which observers with low quality information (i.e. with large spread encounter times) are ignored and potential sources are selected based on the likelihood gradient from high quality observers. The original complexity of PTVA is $O(N^{\alpha})$, where $\alpha \in (3,4)$ depends on the network topology and number of observers (N denotes the number of nodes in the network). Our Gradient Maximum Likelihood Algorithm (GMLA) reduces this complexity to $O(N^2\log(N))$. Extensive numerical tests performed on synthetic networks and real Gnutella network with limitation that id's of spreaders are unknown to observers demonstrate that for scale-free networks with such limitation GMLA yields higher quality localization results than PTVA does.

We live in the networked society. Every second we interact with many networks from which we collect, process and transmit a huge amount of information, which increases exponentially each year[1–4]. Increasing interconnectivity of the world exposes us to world-wide range of pathogens, viruses both physical and virtual, misinformation and rumors with often grievous consequences[5–8]. A good example is a fake tweet about explosion in White House in 2013, which caused $130 billion loss on the stock market[9]. Another example is the United States presidential election of 2016 when many rumors or fake news became viral on Facebook or Twitter and might have affected elections[10]. Many papers seek finding best conditions for spreading[11–16] or sets of optimal spreaders[17–20] but here we investigate an inverse problem. It became clear that one of the major challenges facing network and data scientists is to develop effective methods for detecting and suppressing spread of dangerous viruses, pathogens, misinformation or gossips. The basic component of such a system is undoubtedly a fast algorithm finding a source of such spread. The first widely discussed research on this subject has been done by Shah and Zaman[21] and Pinto, Thiran and Vetterli[22]. In social networks, Shah and Zaman introduced *rumor centrality* of a node as the number of distinct ways a rumor can spread in the network starting from that node. They showed that the node with maximum *rumor centrality* is the Maximum Likelihood Estimator of the rumor source if the underlying graph is a regular tree. They studied also the detection performance for irregular geometric trees, small-word networks and scale-free networks. This method assumes that we know all the connections between nodes and additionally the infection states of all nodes. Pinto *et al.* relaxed some of these constraints since their algorithm requires information about state of not every node, but only about some fraction of nodes called *observers*. A further description of this algorithm is given in the next section and in Supplementary Information. After these two publications, the topic of the source detection became popular and many other variants of this problem have been studied. We can distinguish two main approaches to this issue: the snapshot-based[21,23–25] and the detector-based[22,26,27] source detection. The first one requires the snapshot of an entire network at a certain time instance, the second needs to monitor only a small subset of nodes but all the time. Regardless of the above division, researchers considered also

[1]Center of Excellence for Complex Systems Research, Faculty of Physics, Warsaw University of Technology, Koszykowa 75, 00662, Warsaw, Poland. [2]Social Cognitive Networks Academic Research Center, Rensselaer Polytechnic Institute, 110 8th Street, Troy, NY, 12180-3590, USA. [3]The ENGINE Centre, Wroclaw University of Science and Technology, Wyb. Wyspianskiego 27, 50-370, Wroclaw, Poland. [4]ITMO University, 49 Kronverkskiy av., 197101, Saint Petersburg, Russia. Correspondence and requests for materials should be addressed to R.P. (email: paluch@if.pw.edu.pl)

different epidemic models[25,28], spreading at weighted or time-varying graphs[29–32] and multi-source detection problems[33,34]. In 2014 Jiang *et al.* described state-of-the-art and conducted comparative studies[35]. One of their conclusions is that current methods are too computationally expensive and they can not be use for a quick identification of the propagation source. The main goal of our research was finding the method which executes in reasonable time on large complex networks and delivers high quality of localization results at the same time.

Results

Before demonstrating our main results, we present a brief description of Pinto-Thiran-Vetterli algorithm (PTVA)[22]. Then, we introduce our approach in which observers with low quality information (i.e. with large spread encounter times) are ignored and potential sources are selected based on the likelihood gradient from high quality observers. In order to measure the performance of the algorithms we use three different quality of localization measures: the accuracy, the rank and the distance error. The accuracy is the empirical probability that a source found by the algorithm is the true source. The rank is the true source position on the nodes list, which is sorted in descending order by likelihood of being the source. The distance error is the shortest path distance between the real source and the source found by the algorithm. Details on these measures can be found in the section Methods.

Pinto-Thiran-Vetterli Algorithm. Pinto, Thiran and Vetterli[22] proposed a general framework for the localization of the spread source in which some of the nodes in network act as observers and report from which neighbor and at what time it received the information. However, in real life the identity of the neighbor that sent the message to the observer is not always available (like in the case of gossip spreading on the public square). For this reason, and for the sake of greater generality and applicability of our studies, we do not require data received by observers to contain the identities of nodes from which the spread came. We refer to tests in which PTVA is applied to such data as Pinto-Thiran-Vetterli Algorithm executed on data with Limited Information (PTVA-LI). This lowering of the requirements on input data increases applicability of the methods but reduces detection accuracy, and yet it does not affect the algorithm's complexity or speed. Thus, PTVA-LI tests for the speed and complexity are valid also for PTVA.

PTVA calculates the likelihood of each node to be the source (which we call the score, see Eq. 1 in Sup. Inf. Section S.1) using the reported times (observed delays) from all available observers. For this purpose, PTVA assumes information spreads through the network along the shortest paths and therefore uses breadth-first search (BFS) tree in place of the actual but unknown propagation tree. The method also assumes that the propagation times θ_i for each edge are i.i.d Gaussian random variables, for which the mean μ and the variance σ^2 are known. The algorithm's complexity for arbitrary graphs is $O\ (N(K^3+N^2))$, where N is the size of the network and K is the number of observers. If $K \sim N^{\gamma}$, PTVA complexity ranges from $O(N^3)$ when $\gamma \leq 2/3$ to $O(N^4)$ when $\gamma = 1$. For more details on PTVA see Sup. Inf. Section S.1.

Algorithm 1. Gradient Maximum Likelihood Algorithm.

```
Require: $\mathscr{G}$, $\mu$, $\sigma^2$, $\{o_k\}$, $\{t_k\}$, $K_0$
Ensure: the list of suspected nodes sorted in descending order by the score (the source likelihood)
sort all observers in ascending order of the observed delays $t_k$
leave only first $K_0$ observers from the sorted list - these are the nearest observers
select one observer as reference and label its arrival time $t_r$
compute the vector of observed delays relative to $t_r$: $[\mathbf{d}]_k = t_k - t_r$
initialize the empty list $S$ of pairs (node, score)
initialize $v \leftarrow (o_1, 0)$        ▷ $o_1$ is the first observer from the sorted list, 0 is its initial score $\phi(o_1)$
initialize $\phi_{max} \leftarrow 0$        ▷ $\phi_{max}$ is the variable that store the highest score
while $\phi(v) \geqslant \phi_{max}$ do
    initialize the empty list $T_v$ of pairs (node, score)
    for each neighbor $n$ of $v$ do
        if $n \notin S$ then
            build the diffusion tree $\mathscr{T}_{\mathrm{bfs},n}$ from the shortest paths between $n$ and the nearest observers
            compute $\mu_n$ and $\Lambda_n$ with respect to tree $\mathscr{T}_{\mathrm{bfs},n}$
            compute the score $\phi(n) = \exp(-\frac{1}{2}(\mathbf{d} - \mu_n)^{\mathrm{T}}\Lambda_n^{-1}(\mathbf{d} - \mu_n))/|\Lambda_n|^{1/2}$
            add $(n, \phi(n))$ to $T_v$
        end if
    end for
    if $T_v$ is not empty then
        $v \leftarrow$ the neighbor $n$ of $v$ with the highest score $\phi(n)$
        add nodes from $T_v$ to $S$
    else
        break the while loop
    end if
end while
sort $S$ in descending order by the score
```

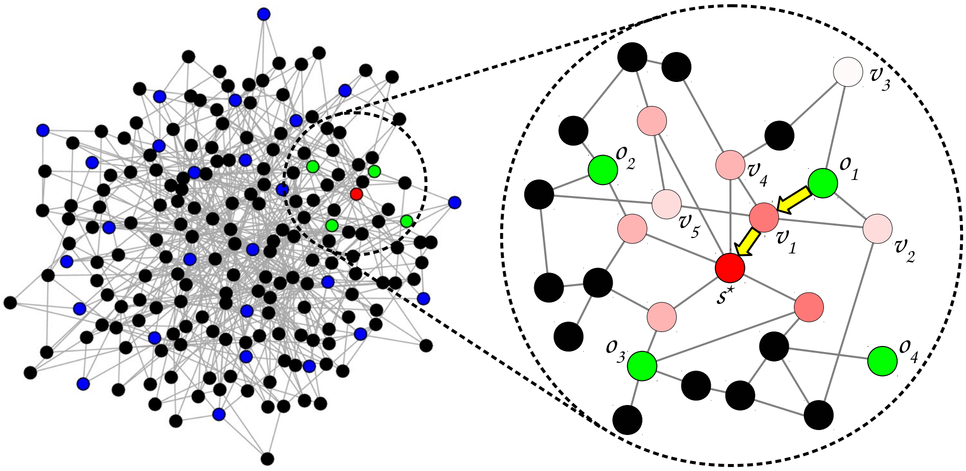

Figure 1. The visualization of GMLA. The left picture presents the whole graph. The red node is the true source. Green nodes mark the K_0 nearest observers with the smallest time delays (in this plot $K_0 = 4$). The rest of the observers in the network are highlighted in blue. The picture on the right is a zoom of a small area around the nearest observers. In this picture the color corresponds to the score (the likelihood of being the source) of the node (except for the observers which are green). The higher the score of the node is, the darker red is its color. At the beginning the algorithm computes the scores for the neighbors of the nearest observer (in this plot *the observer one* is o_1 and its neighbors are v_1, v_2, and v_3). Afterwards GMLA selects the neighbor with the highest score (v_1 in this case) and starts computing the scores for its neighbors (o_1, v_4, v_5 and s^*). During this step there is no need for estimating the likelihood for the node v_2 since it was done in the previous step. All the scores which are computed are stored in the list. Since s^* has the highest score among the neighbors of v_1, in the next step GMLA will compute the scores for it neighbors. None of the neighbors of s^* has higher score than s^*, therefore the algorithm stops here. The node s^* is the source according to GMLA because it has the highest score from all tested (suspected) nodes. The nodes not visited by the algorithm are black since their scores are undefined (their scores are not computed).

Gradient Maximum Likelihood Algorithm. *Description.* Compared to the framework introduced by PTVA-LI we propose two improvements: a limited number of observers, and a gradient-like selection of suspected nodes. The first idea takes advantage of the fact that observers which are very far from the spread source make very small contribution to the score in comparison to the nearest observers (Fig. S4 in Sup. Inf. Section S.2). On the other hand, those distant observers increase greatly the cost of information processing. Since a distance between any observer o_k and the true source should increase (in average) with the arrival time t_k, we can use only a small number $K_0 \ll K$ of the nearest observers and drastically shorten the time needed for computing the score. The limited number of observers was used in earlier work[22,36] where the search algorithm was run before all K observers get infected in order to limit the outbreak. In contrast, here we focus on the optimization of the algorithm's complexity for large complex networks.

The second idea introduces a procedure of the nodes selection for the score calculation. It is very likely that the spread source is in close proximity to the observer which has the smallest time at which the spread was observed (the *observer one*). The procedure starts by calculating scores of the nearest neighbors of the *observer one* and then selects a neighbor with the highest score. Next, the algorithm jumps into this node and calculates scores for its nearest neighbors in order to find the one which has a score greater than or equal to the current maximum. The process is gradient-like and it is continued until all neighbors have a score lower than the current maximum (see Fig. 1). Each calculated score is remembered (along with the node) which allows the algorithm to avoid double-calculation and to prepare a ranking of nodes suspected to be the source. The number of suspected nodes $N_0 = |V_s|$ depends primarily on the size of the network and the average degree $\langle k \rangle$. The empirical studies shows that $N_0 \sim \langle k \rangle \log(N)$ (Fig. S6 in Sup. Inf. S.2). It is worth noting that the algorithm does not guarantee that the true source s^* will be selected for score calculation, i.e. $\mathrm{P}(s^* \in V_s) < 1$ (see Fig. S9a and S10a in Sup. Inf. S.2).

The Gradient Maximum Likelihood Algorithm (GMLA) is summarized in Algorithm 2. $\mathcal{G}$ denotes the underlaying graph, μ and σ^2 denote the mean and the variance of the random propagation delay associated with one edge, $\{o_k\}$ is the set of observers and $\{t_k\}$ are the times at which they observed the spread. The score of a node is the likelihood that this node is the true source. We denote the score of a node v as $\phi(v)$. The formulas for $\phi(v)$, μ_v and Λ_v are given by equations (1,3,4) in Sup. Inf.

Complexity. Using the symbols K_0 and N_0 we reformulate the time complexity of GMLA as $O(N_0(K_0^3 + N^2))$ in the worst case. Assuming $N_0 \sim \log(N)$ and $K_0 \ll N$, which is true for our method, the complexity can be further simplified into $O(\log(N)N^2)$.

Fine-tuning and performance. The number of the nearest observers K_0 is a crucial parameter of GMLA and should be carefully selected. If K_0 is too small, the accuracy of the algorithm decreases. On the other hand, large

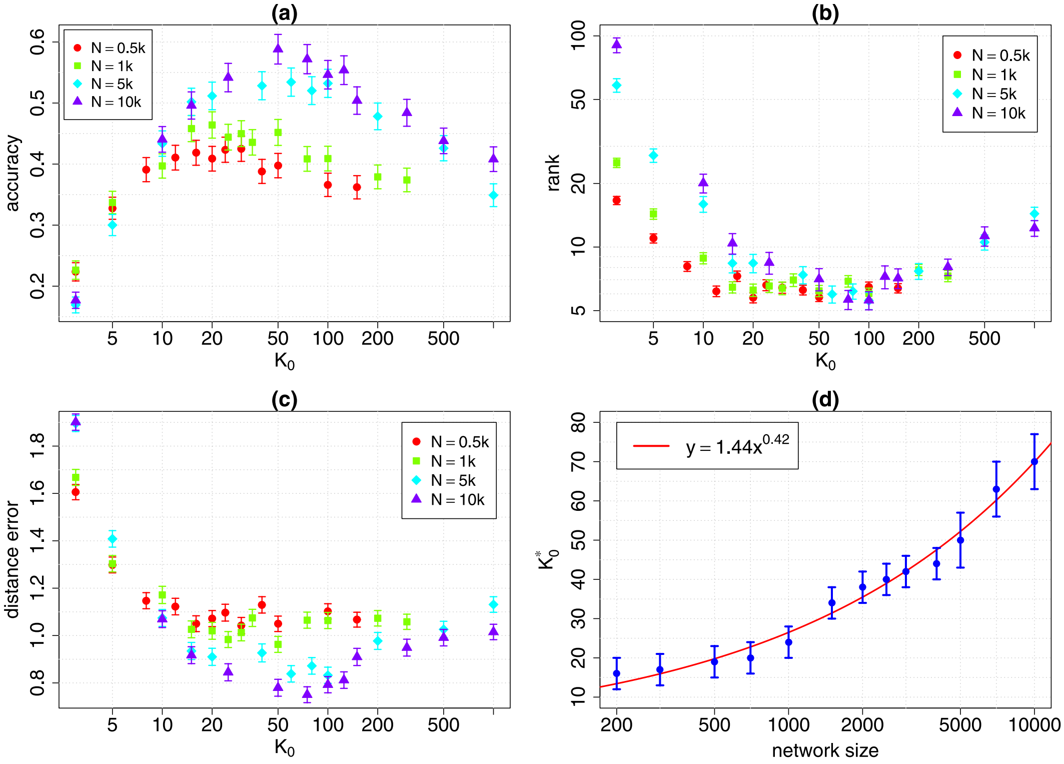

Figure 2. Performance of GMLA versus number of the nearest observers K_0 for various sizes of BA network ($m = 3$). (**a**) The maximum of accuracy is not very sharp but is clearly visible. (**b**,**c**) The minima of rank and distance error are not always compatible with the maximum of accuracy which increases the uncertainty of finding K_0^*. (**d**) The optimal number of the nearest observers K_0^* for various sizes of BA network. The solid line is a nonlinear least squares model $y = bx^a$ (Gauss-Newton algorithm), where $a = 0.42 \pm 0.04$ and $b = 1.44 \pm 0.51$ (95% confidence interval).

K_0 increases the time of computation. The optimal number of the nearest observers K_0^* is the minimal number of the nearest observers K_0 needed to achieve maximal quality of the spread source localization. We test how K_0^* depends on the network size, the average degree and the propagation ratio for Erdös-Rényi (ER) and Barabási-Albert (BA) networks[37] (see Sup. Inf. Section S.3). No substantial relationship was found between K_0^* and the average degree of the network or the propagation ratio (Figs S15–S18 in Sup. Inf. S.3). Figure 2 presents how the number of the nearest observers affects the performance of GMLA for various sizes of BA network with the minimum degree $m = 3$ (m is the initial degree of each attached node, thus $\langle k\rangle = 2m = 6$). It is easy to see a peak of the accuracy and the valleys of the rank and distance error. Figure 2d shows the estimates of K_0^* for different sizes of BA network. In the case of Erdös-Rényi network, no peak of the accuracy is observed, but the saturation point is clearly visible (Fig. S13c in Sup. Inf. S.3). This also applies to the rank of the true source and the distance error (Fig. S13d,e). The fact that we can observe the peak of the accuracy for BA networks (not only the saturation point like for ER graphs) has substantial consequences, because it means that taking only $K_0 \ll K$ nearest observers not only shortens the computation time, but it may also improve the quality of the source localization under certain circumstances. As we show further in Discussion, such a circumstance is the occurrence of the hubs in BA network. In the next paragraphs we present a numerical estimation of the complexity of GMLA as well as its performance in terms of the quality of results in comparison to PTVA-LI.

Tests on synthetic networks. We tested GMLA and PTVA-LI for various sizes of Erdös-Rényi (ER) random graphs and Barabási-Albert (BA) networks. We used Susceptible-Infected model (see details in section Methods) for the spread with the infection rate $\beta = 0.5$ ($\lambda = \sqrt{2}$). The observers were distributed randomly over a whole network with the density $\rho = 0.2$. In order to maintain a high efficiency of GMLA, we set the number of the nearest observers as a function of the network size $K_0 = 0.5\sqrt{N}$ (see Fig. 2d and Fig. S12 in Sup. Inf. S.3) For comparative purposes, we introduce also a baseline method. The baseline method is very naive and according to it, the true source is always the *observer one* (with smallest delay t_k). Details on the baseline method are given in section Methods.

The most important feature of GMLA is a remarkable reduction of the computation time. Figures 3d and 4d show that the empirical complexity decreases from $O(N^{3.46})$ to $O(N^{1.15})$ for ER graph and from $O(N^{3.49})$ to $O(N^{1.32})$ for BA network. Furthermore, one can observe an initial difference between GMLA and PTVA-LI computing times for the networks of size 200, which is a factor 4.4 for ER graph and 3.6 for BA network.

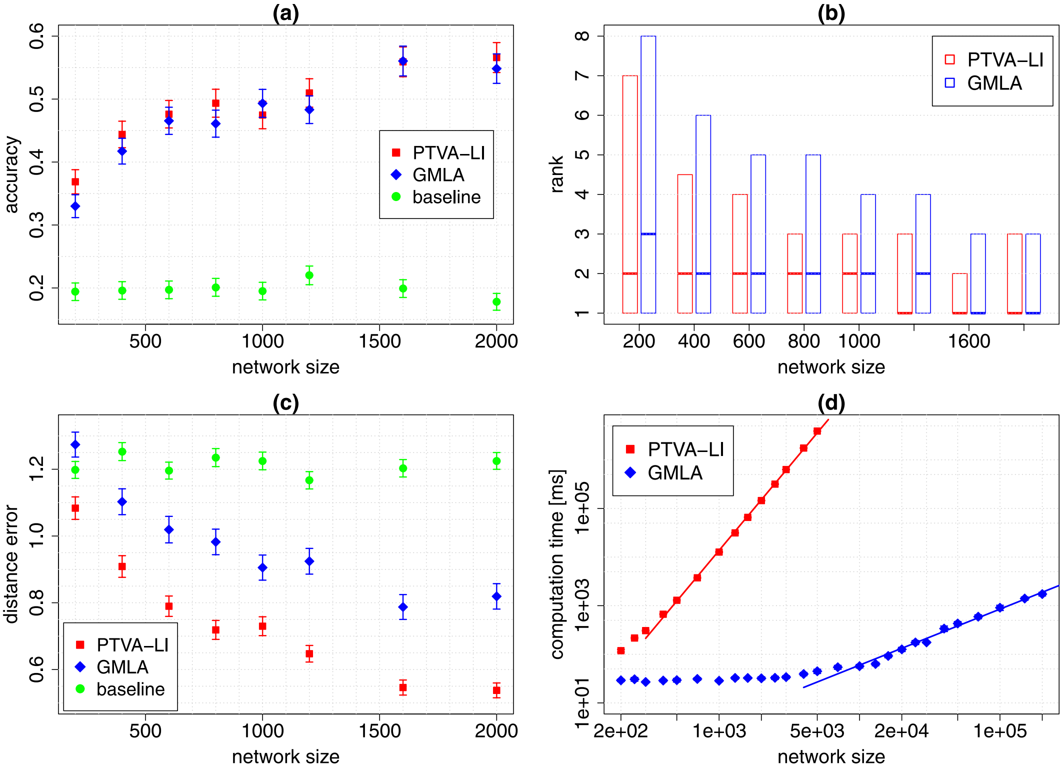

Figure 3. GMLA (blue) versus PTVA-LI (red) and the baseline method (green) on ER network. (**a**) The accuracy of both algorithms is almost the same and it increases with the network size. (**b**) The band inside the box shows the median of the true source rank. The bottom and top of the box show the values of the first and third quartiles. (**c**) The mean distance error for PTVA-LI and GMLA decreases with the network size. (**d**) The mean computation time of a single realization of GMLA is substantially shorter than of PTVA-LI. The solid lines are linear models $ln(\text{time}) = a\ \ln(\text{size}) + b$, where $a = 3.46 \pm 0.07$ for PTVA-LI and $a = 1.15 \pm 0.09$ for GMLA (95% confidence interval). In the figures (**a–c**) each point is the result of 1000 realizations. In the figure (**d**) results for PTVA-LI and GMLA are averaged over 26 realizations. Parameters: $\langle k\rangle = 6$, $\lambda = \sqrt{2}$, $\rho = 0.2$, $K_0 = 0.5\sqrt{N}$ (details in the text).

The quality of the source localization clearly depends on the network topology. In general, both algorithms achieve better results for ER graphs than BA networks. In the case of ER graphs, the accuracy of both algorithms is almost the same (Fig. 3a), but PTVA-LI is characterized by lower rank and distance error (Fig. 3b,c). On the other hand, for BA networks which are larger than 300 nodes GMLA outperforms PTVA-LI in every test of quality of the results (Fig. 4a–c). Moreover, the advantage of GMLA increases with the size of BA network and is especially high for large networks, for which the computation of PTVA-LI takes too long to collect a large enough statistics.

Tests on real social network. Another test was performed on Gnutella, a real peer-to-peer network. This kind of network is used for direct exchange of data via Internet between users and therefore can be used to spread the malware. The graph obtained from SNAP Datasets[38–40] contains $N = 6299$ nodes and has the average degree $\langle k\rangle = 6.6$ (more details on data are in the section Methods). We examine the algorithms for different densities of the observers, but we keep a constant number of the nearest observers in GMLA ($K_0 = 30$). During tests we use simple SI model to simulate spreading. The results are shown in Fig. 5. For the density of the observers below 10% the outcomes of both methods are very similar – GMLA has slightly better accuracy but visibly worse rank than PTVA-LI. The situation changes when the density of the observers is equal or greater than 10% – GMLA performs better according to all efficiency measures. However, the main difference between these algorithms lies in the computation time (Fig. 5b). Initially, for $\rho = 2.5\%$ the computation time differs by a factor 61.5, but it increases with the density of observers since the computation time for PTVA-LI increases with ρ (see Fig. S2d in Sup. Inf. Section S.1).

Discussion

We introduce a new algorithm (GMLA) for the spread source localization in the well-known Pinto-Thiran-Vetterli limited observers formulation. The main drawback of the Pinto-Thiran-Vetterli Algorithm (PTVA) is its time complexity. For large networks with many observers the complexity of PTVA is defined by the complexity of matrix operations, which is $O(K^3)$ per node in the worst case (where K denotes the number of observers). We avoid this drawback in out algorithm by reducing the number of the observers used to determine the score (the likelihood of being the source) and by limiting the number of suspected nodes. The latter is performed by the

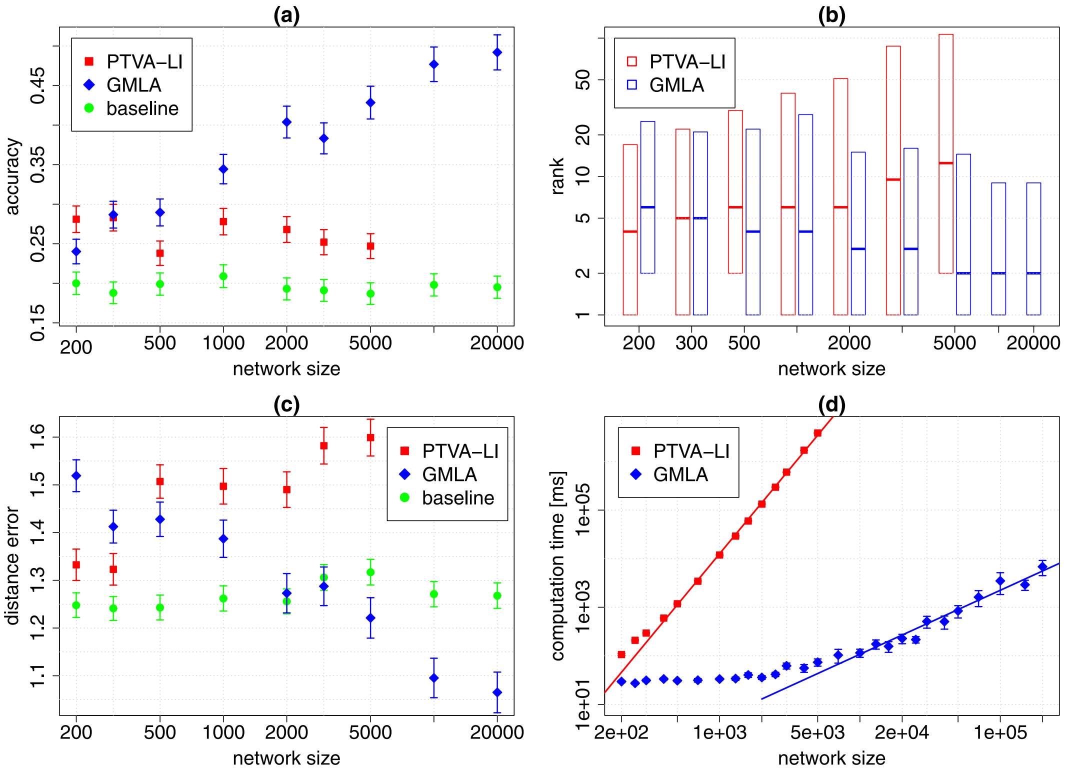

Figure 4. GMLA (blue) versus PTVA-LI (red) and the baseline method (green) on BA network ($m = 3$). (**a**) The accuracy of GMLA increases with the network size and it is much higher than accuracy of PTVA-LI. (**b**) The band inside the box shows the median of the true source rank. The bottom and top of the box show the values of the first and third quartiles. GMLA has much lower rank than PTVA-LI. (**c**) The mean distance error for PTVA-LI increases with the network size in contrast to GMLA, which initially is the highest in the plot, but becomes the lowest for the networks larger that 3000 nodes. (**d**) The mean computation time of a single realization of GMLA is substantially shorter than of PTVA-LI. The solid lines are linear models $\ln(\text{time}) = a\ \ln(\text{size}) + b$, where $a = 3.49 \pm 0.08$ for PTVA-LI and $a = 1.32 \pm 0.08$ for GMLA (95% confidence interval). In the figures (**a**)–(**c**) each point is the result of 1000 realizations. In the figure (**d**) results for PTVA-LI and GMLA are averaged over 26 realizations. Parameters: $\langle k \rangle = 6$, $\lambda = \sqrt{2}$, $\rho = 0.2$, $K_0 = 0.5\sqrt{N}$ (details in the text).

selection procedure which starts from the neighbors of the first observer and follows the gradient of the score. As a result of the selection, we get a limited number of the suspected nodes $N_0 = |V_s| \sim \log N$ in contrast to PTVA where each node is checked ($V_s = V$). Thanks to this approach, the complexity of Gradient Maximum Likelihood Algorithm (GMLA) is $O(\log(N)N^2)$ in the worst case and as far as we know this is the fastest algorithm for the spread source detection in generic networks with incomplete observations.

We test GMLA and PTVA-LI on Erdös-Rényi, Barabási-Albert and Gnutella networks and compare performance of these algorithms using three measures: the accuracy, the rank of true source, and the distance error. Both algorithms work noticeably better for ER graphs than BA networks. For ER graphs, the quality of source localization by both algorithms is similar (with a minimal advantage of PTVA-LI), but for BA networks GMLA achieves much better results. The additional tests performed on Regular Random Graphs (Fig. S19 in Sup. Inf. Section S.4), Exponential Random Graph (Figs S20, S21 in Sup. Inf. S.4) and Configuration Model with the degree distribution which follows a power-law (Fig. S22 in Sup. Inf. S.4) confirm that GMLA outperforms PTVA-LI for scale-free networks. As is well known, the essential property of scale-free network is existence of the hubs - the nodes with a very high degree (here we consider nodes with $k \geqslant \sqrt{N}$ to be the hubs). The hubs are usually responsible for a very rapid spread in the network, but can their presence hinder detection of the source? Fig. 6a shows the accuracy of PTVA-LI for 4 special sets of observers in BA network. All sets are equipotent (15 nodes) and contain only the observers which are the second order neighbors of the true source. In addition, the first set (black triangles) consists solely of the observers which are "behind" the hubs. We say the observer is "behind" the hub (or is noisy) if the shortest path between this observer and the true source passes through any hub. This also applies to the observers which are the hubs. The second set (gold triangles) is the opposite of the first set - it contains only non-noisy observers which are not "behind" any hub. The third set (dark red squares) is a random mixture of the first two. The last set (purple diamonds) consists of the observers which have the smallest times at which the spread reached them (the quickest observers). This is the same criterion for the selection of observers as that which GMLA uses. As Fig. 6a shows, using the observers "behind" the hubs substantially worsens the accuracy of PTVA-LI. It means that information is degraded after passing through the hub. This is the main reason why PTVA-LI and GMLA are less effective for scale-free networks. The highest accuracy of PTVA-LI is

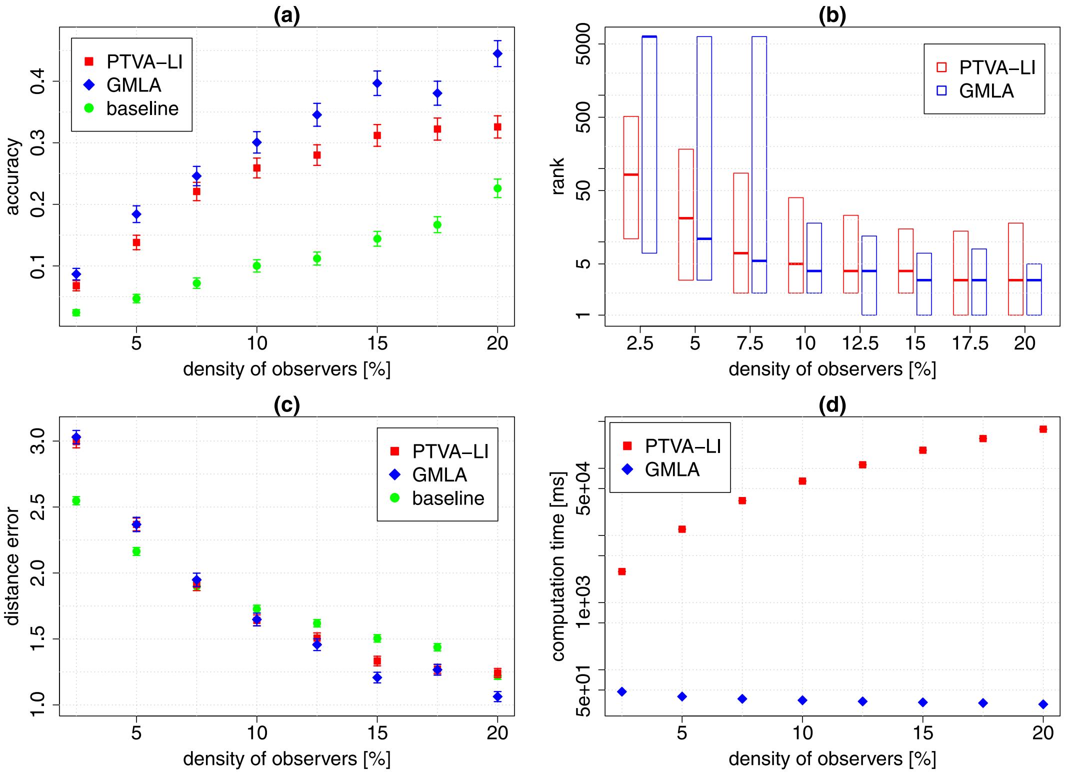

Figure 5. GMLA (blue) versus PTVA-LI (red) and the baseline method (green) on Gnutella network. (**a**) GMLA achieves higher accuracy than PTVA-LI and the baseline method. (**b**) The band inside the box shows the median of the true source rank. The bottom and top of the box show the values of the first and third quartiles. In a single realization, GMLA can have rank equal to $N = 6299$ when the score for the true source is undefined (see Fig. 1). (**c**) The mean distance error. For $\rho > 10\%$ GMLA has the lowest distance error. (**d**) The mean computation time of a single realization of GMLA is substantially shorter that PTVA-LI and, in contrast to PTVA-LI, is independent of the density of observers. Each point in the figures is the result of 1000 realizations. Parameters: $\lambda = \sqrt{2}$, $K_0 = 30$ (details in the text).

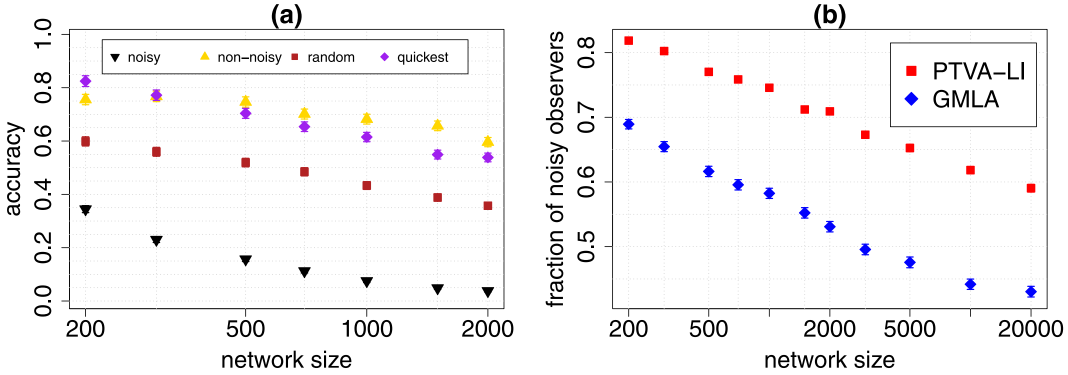

Figure 6. Effect of hubs in BA network ($m = 3$ and $\lambda = \sqrt{2}$ in all simulations). (**a**) Accuracy of the source localization vs network size for 4 sets of observers. Each set contains 15 observers which are 2 hops from the true source. Black triangles is the set containing only observers which are "behind" the hubs (noisy observers). Gold triangles is the set with only non-noisy observers. Dark red squares is the set with randomly selected observers. Purple diamonds is the set with the observers with the smallest time delays. These observers are the quickest receivers. Each point is the result of 2000 realizations. (**b**) The fraction of the observers which are "behind" the hubs (noisy observers) for PTVA-LI and GMLA. The analysis was done on BA network ($m = 3$) with constant density of observers $\rho = 0.2$ and the number of the nearest observers $K_0 = 0.5\sqrt{N}$. The results are averaged over 1000 realizations.

achieved when using only non-noisy observers. However, the quality of the source localization of the algorithm with the quickest observers is only slightly lower. Since GMLA uses the quickest observers, it achieves better results than PTVA-LI in scale-free networks with hubs, because the nearest observers infrequently are "behind" the hubs for sufficiently large networks, as is confirmed by Fig. 6b. Moreover, this conclusion is supported by the results obtained for Gnutella network, which also contains some hubs (0.4% of nodes has degree $k \geqslant \sqrt{N}$).

Although GMLA does not use information from all observers, as PTVA-LI does, it achieves better results for scale-free networks in quality of localization tests based on three measures: the accuracy, the rank of true source, and the distance error. This is because GMLA acts like a filter and rejects low quality information from distant observers which are often "behind" the hubs.

In summary, we proposed a new method for fast and accurate detection of spread source with incomplete observations which is capable to process timely large networks consisting of tens of thousands of nodes. Our algorithm is much faster and provides higher quality of localization results than Pinto-Thiran-Vetterli algorithm for scale-free networks. The key to this success is limiting the information sources to the most important observers, while ignoring excessive and noisy information from far observers, as well as use of likelihood gradient for selection of potential spread sources. The phrase "less is more" once again turned out to be truth here.

Methods

Propagation ratio.

For spreading process we define the propagation ratio λ as the ratio between the mean μ and the standard deviation σ of time delay associated with an edge in the network.

Susceptible-Infected (SI) model.

We simulate the spread through the network using discrete Susceptible-Infected (SI) model[41]. In this model each node can be in one of two states: susceptible or infected. At $t = 0$ only one random node is infected. We called this node the true source. At each subsequent time step each infected node has a chance to pass the information to its neighbor. The number of chances per time step is equal to the number of neighbors and for each neighbor the probability of success β is the same. The parameter β is called the infection rate. Since the number of time steps needed to pass the information from one node to its neighbor is equal to the number of independent trials (with the probability β) needed for first occurrence of success, it is described by the geometric distribution and therefore the mean propagation time per edge is $\mu = 1/\beta$ and the variance is $\sigma^2 = (1-\beta)/\beta^2$. It follows that the propagation ratio $\lambda = \mu/\sigma$ for SI model is $\lambda = 1/\sqrt{1-\beta}$.

Efficiency measures.

Accuracy. The accuracy of a single realization is $a_i = 1/\left|V_{top}\right|$ if $s^* \in V_{top}$ or $a_i = 0$ otherwise, where s^* is the true source and V_{top} is a group of nodes with the highest score (top scorers). The total accuracy a is an average of many realizations a_i, therefore $a \in [0,1]$. This measure takes into account the fact that there might be more than one node with the highest score (ties are possible).

Rank. The rank is the position of the true source on the node list sorted in descending order by the score. In other words this measure shows how many nodes, according to an algorithm, is a better candidate for a source than the true source. If the real source has exactly the same score as some other node (or nodes), the true source is always below that node (these nodes) on the score list sorted in descending order. The rank takes into account the fact that an algorithm which is very poor in pointing out the source exactly (low accuracy) can be very good at pointing out a small group of nodes among which is the source.

Distance error. The distance error is the number of hops (edges) between the true source and a node designated as the source by an algorithm. If $|V_{top}| > 1$, which means that an algorithm found more than one candidate for the source, the distance error is computed as a mean shortest path distance between the real source and the top scorers.

Baseline method.

The baseline method serves as the benchmark for accuracy and distance error tests. It assumes that the real source is the first observer reporting the spread. The baseline method works in no time and its accuracy is expected to be equal to the density of observers; this follows from the fact that if the true source is among the observers, it has to be the observer with the smallest arrival time. One can expect a quite low value of the mean distance error in this case, because the baseline method never makes big mistakes in terms of distance from the true source. Apart the poor accuracy, the baseline method does not assign the scores to the nodes which means that it cannot be used to find the rank of the real source.

Gnutella peer-to-peer network.

We used the data from SNAP Datasets[38–40]. This dataset consists of a snapshot of the Gnutella peer-to-peer file sharing network from 8 August 2002. Nodes represent hosts in the Gnutella network topology and edges represent connections which were established on 8 August 2002. The data has been anonymized by the researchers from Stanford University before it was made available. The graph contains $N_{tot} = 6301$ nodes and $E_{tot} = 20777$ edges, but we use the largest connected component which consists of $N = 6299$ nodes and $E = 20776$ edges ($\langle k \rangle = 6.6$). The diameter of the network is 9, the average path length is 3.7 and the average clustering coefficient is 0.0109.

Testbed.

The time tests were performed in Java 7 using AMD FX-8350 4 GHz processor. We used jblas v.1.2.4[42] as a fast linear algebra library for Java.

References

1. Barabási, A.-L. *Linked: How Everything Is Connected to Everything Else and What It Means for Business, Science, and Everyday Life* (Plume, 2003).
2. Newman, M. E. J. The structure and function of complex networks. *SIAM Review* **45**, 167–256, https://doi.org/10.1137/S003614450342480 (2003).
3. Helbing, D. & Balietti, S. From social data mining to forecasting socio-economic crises. *The European Physical Journal Special Topics* **195**, 3, https://doi.org/10.1140/epjst/e2011-01401-8 (2011).

4. Giannotti, F. *et al.* A planetary nervous system for social mining and collective awareness. *The European Physical Journal Special Topics* **214**, 49–75, https://doi.org/10.1140/epjst/e2012-01688-9 (2012).
5. Pastor-Satorras, R. & Vespignani, A. Epidemic spreading in scale-free networks. *Phys. Rev. Lett.* **86**, 3200–3203, https://doi.org/10.1103/PhysRevLett.86.3200 (2001).
6. Moya, I., Chica, M., Saez-Lozano, J. L. & Cordon, O. An agent-based model for understanding the influence of the 11-M terrorist attacks on the 2004 Spanish elections. *Knowledge-based Systems* **123**, 200–216, https://doi.org/10.1016/j.knosys.2017.02.015 (2017).
7. Sun, M., Zhang, H., Kang, H., Zhu, G. & Fu, X. Epidemic spreading on adaptively weighted scale-free networks. *Journal of Mathematical Biology* **74**, 1263–1298, https://doi.org/10.1007/s00285-016-1057-6 (2017).
8. Fu, F., Christakis, N. A. & Fowler, J. H. Dueling biological and social contagions. *Scientific Reports* **7**. https://doi.org/10.1038/srep43634 (2017).
9. Strauss, G., Shell, A., Yu, R. & Acohido, B. SEC, FBI probe fake tweet that rocked stocks. *USA Today* https://www.usatoday.com/story/news/nation/2013/04/23/hack-attack-on-associated-press-shows-vulnerable-media/2106985/ (2013).
10. Alcott, H. & Gentzkow, M. Social Media and Fake News in the 2016 Election. *Journal of Economic Perspectives* **31**, 211–236, https://web.stanford.edu/gentzkow/research/fakenews.pdf (2017).
11. Lind, P. G., da Silva, L. R., Andrade, J. S. & Herrmann, H. J. Spreading gossip in social networks. *Phys. Rev. E* **76**, 036117, https://doi.org/10.1103/PhysRevE.76.036117 (2007).
12. Stegehuis, C., van der Hofstad, R. & van Leeuwaarden, J. S. H. Epidemic spreading on complex networks with community structures. *Scientific Reports* **6**, 29748 https://www.nature.com/articles/srep29748 (2016).
13. Wang, J., Sun, E., Xu, B., Li, P. & Ni, C. Abnormal cascading failure spreading on complex networks. *Chaos, Solitons & Fractals* **91**, 695–701 http://www.sciencedirect.com/science/article/pii/S0960077916302442. https://doi.org/10.1016/j.chaos.2016.08.007 (2016).
14. Liu, Q.-H., Wang, W., Tang, M., Zhou, T. & Lai, Y.-C. Explosive spreading on complex networks: The role of synergy. *Phys. Rev. E* **95**, 042320, https://doi.org/10.1103/PhysRevE.95.042320 (2017).
15. Czaplicka, A., Hołyst, J. A. & Sloot, P. M. A. Stochastic resonance for information flows on hierarchical networks. *The European Physical Journal Special Topics* **222**, 1335–1345, https://doi.org/10.1140/epjst/e2013-01929-5 (2013).
16. Czaplicka, A., Holyst, J. A. & Sloot, P. M. A. Noise enhances information transfer in hierarchical networks. *Scientific reports* **3**, 1223 https://www.nature.com/articles/srep01223. https://doi.org/10.1038/srep01223 (2013).
17. Ash, C. Superspreaders are local and disproportionate. *Science* 355, 1036 LP–1036 http://science.sciencemag.org/content/355/6329/1036.1.abstract (2017).
18. Morone, F. & Makse, H. A. Influence maximization in complex networks through optimal percolation. *Nature* **524**, 65–68 http://www.nature.com/nature/journal/v524/n7563/abs/nature14604.html (2015).
19. Jankowski, J. *et al*. Balancing Speed and Coverage by Sequential Seeding in Complex Networks. *Scientific Reports* **7**, 891 http://www.nature.com/articles/s41598-017-00937-8., https://doi.org/10.1038/s41598-017-00937-8 (2017).
20. Singh, P., Sreenivasan, S., Szymanski, B. K. & Korniss, G. Threshold-limited spreading in social networks with multiple initiators. *Scientific reports* **3**, 2330 http://www.nature.com/srep/2013/130731/srep02330/full/srep02330.html. https://doi.org/10.1038/srep02330 (2013).
21. Shah, D. & Zaman, T. Rumors in a network: Who's the culprit? *IEEE Transactions on Information Theory* **57**, 5163–5181, https://doi.org/10.1109/TIT.2011.2158885 (2011).
22. Pinto, P. C., Thiran, P. & Vetterli, M. Locating the source of diffusion in large-scale networks. *Physical Review Letters* **109**, 1–5, https://doi.org/10.1103/PhysRevLett.109.068702 (2012).
23. Prakash, B. A., Vrekeen, J. & Faloutsos, C. Spotting culprits in epidemics: How many and which ones? *Proceedings - IEEE International Conference on Data Mining, ICDM* 11–20. https://doi.org/10.1109/ICDM.2012.136 (2012).
24. Lokhov, A. Y., Mézard, M., Ohta, H. & Zdeborová, L. Inferring the origin of an epidemic with a dynamic message-passing algorithm. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* **90**, 1–9, https://doi.org/10.1103/PhysRevE.90.012801 (2014).
25. Zhu, K. & Ying, L. Information Source Detection in the SIR Model: A Sample-Path-Based Approach. *IEEE/ACM Transactions on Networking* **24**, 408–421, https://doi.org/10.1109/TNET.2014.2364972 (2016).
26. Rumor source detection under probabilistic sampling. *IEEE International Symposium on Information Theory - Proceedings* 2184–2188. https://doi.org/10.1109/ISIT.2013.6620613 (2013).
27. Luo, W., Tay, W. P. & Leng, M. How to identify an infection source with limited observations. *IEEE Journal on Selected Topics in Signal Processing* **8**, 586–597, https://doi.org/10.1109/JSTSP.2014.2315533 (2014).
28. Brockmann, D. & Helbing, D. The Hidden Geometry of Complex, Network-Driven Contagion Phenomena. *Science* **342**, 1337–1342, https://doi.org/10.1126/science.1245200 (2013).
29. Antulov-Fantulin, N., Lančić, A., Šmuc, T., Štefančić, H. & Šikić, M. Identification of Patient Zero in Static and Temporal Networks: Robustness and Limitations. *Physical Review Letters* **114**, 1–5, https://doi.org/10.1103/PhysRevLett.114.248701 (2015).
30. Shen, Z., Cao, S., Wang, W. X., Di, Z. & Stanley, H. E. Locating the source of diffusion in complex networks by time-reversal backward spreading. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* **93**, 1–9, https://doi.org/10.1103/PhysRevE.93.032301 (2016).
31. Braunstein, A. & Ingrosso, A. Inference of causality in epidemics on temporal contact networks. *Scientific Reports* **6**, 27538 http://www.nature.com/articles/srep27538. https://doi.org/10.1038/srep27538 (2016).
32. Jiang, J., Wen, S., Yu, S., Xiang, Y. & Zhou, W. Rumor Source Identification in Social Networks with Time-varying Topology. *IEEE Transactions on Dependable and Secure Computing* **5971**, 1–1 http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7393814. https://doi.org/10.1109/TDSC.2016.2522436 (2016).
33. Fu, L., Shen, Z. S., Wang, W. X., Fan, Y. & Di, Z. R. Multi-source localization on complex networks with limited observers. *Epl* **113** DOI Artn 18006 10.1209/0295-5075/113/18006 (2016).
34. Fioriti, V., Chinnici, M. & Palomo, J. Predicting the sources of an outbreak with a spectral technique. *Applied Mathematical Sciences* **8**, 6775–6782 http://arxiv.org/abs/1211.2333. https://doi.org/10.12988/ams.2014.49693 (2014).
35. Jiang, J., Wen, S., Yu, S., Xiang, Y. & Zhou, W. Identifying Propagation Sources in Networks: State-of-the-Art and Comparative Studies. *IEEE Communications Surveys and Tutorials* **X**, 1–17, https://doi.org/10.1109/COMST.2016.2615098 (2014).
36. Spinelli, B., Celis, L. E. & Thiran, P. Observer Placement for Source Localization: The Effect of Budgets and Transmission Variance. 743–751 (54th Annual Allerton Conference on Communication, Control, and Computing (Allerton). https://doi.org/10.1109/ALLERTON.2016.7852307 (2016)
37. Albert, R. & Barabási, A. L. Statistical mechanics of complex networks. *Reviews of Modern Physics* **74**, 47–97, https://doi.org/10.1088/1478-3967/1/3/006 (2002).
38. Leskovec, J. & Krevl, A. Gnutella peer-to-peer network: snapshot from August 8, http://snap.stanford.edu/data/p2p-Gnutella08.html. Accessed: 2017-11-30 (2002).
39. Ripeanu, M., Iamnitchi, A. & Foster, I. Mapping the gnutella network. *IEEE Internet Computing* **6**, 50–57, https://doi.org/10.1109/4236.978369. (2002).
40. Leskovec, J., Kleinberg, J. & Faloutsos, C. Graph evolution: Densification and shrinking diameters. *ACM Trans. Knowl. Discov. Data* **1** https://doi.org/10.1145/1217299.1217301 (2007).
41. Bailey, N. T. J. *The Mathematical Theory of Infectious Diseases and its Applications*. (Hafner Press, New York, 1975).
42. Braun, N. L., Schaback, J. & Jugel, M. L. jblas - Linear Algebra for Java. http://jblas.org/.

Acknowledgements

The work was partially supported as RENOIR Project by the European Union Horizon 2020 research and innovation programme under the Marie Skodowska-Curie grant agreement No 691152, by Ministry of Science and Higher Education (Poland), grant Nos. 34/H2020/2016, 329025/PnH /2016, and by National Science Centre, Poland Grant No. 2015/19/B/ST6/02612. J.A.H. was partially supported by the Russian Scientific Foundation, Agreement #17-71-30029 with co-financing of Bank Saint Petersburg. X.L. and B.K.S. were partially supported by the Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053 (the ARL Network Science CTA) and by the Army Research Office grant W911NF-16-1-0524. B.K.S was partially supported by the National Science Centre, Poland, project no. 2016/21/B/ST6/01463. This research was also supported in part by PLGrid Infrastructure.

Author Contributions

R.P., K.S., B.K.S. and J.A.H. designed the research; R.P. implemented and performed numerical experiments and simulations; R.P., X.L., K.S., B.K.S. and J.A.H. analyzed data and discussed results; R.P., X.L., K.S., B.K.S. and J.A.H. wrote and reviewed the manuscript.

Additional Information

Supplementary information accompanies this paper at https://doi.org/10.1038/s41598-018-20546-3.

Competing Interests: The authors declare that they have no competing interests.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.