nature human behaviour

ARTICLES

https://doi.org/10.1038/s41562-017-0290-3

Hiding individuals and communities in a social network

Marcin Waniek[1,2], Tomasz P. Michalak[1,3*], Michael J. Wooldridge[3] and Talal Rahwan[2*]

The Internet and social media have fuelled enormous interest in social network analysis. New tools continue to be developed and used to analyse our personal connections, with particular emphasis on detecting communities or identifying key individuals in a social network. This raises privacy concerns that are likely to exacerbate in the future. With this in mind, we ask the question 'Can individuals or groups actively manage their connections to evade social network analysis tools?' By addressing this question, the general public may better protect their privacy, oppressed activist groups may better conceal their existence and security agencies may better understand how terrorists escape detection. We first study how an individual can evade 'node centrality' analysis while minimizing the negative impact that this may have on his or her influence. We prove that an optimal solution to this problem is difficult to compute. Despite this hardness, we demonstrate how even a simple heuristic, whereby attention is restricted to the individual's immediate neighbourhood, can be surprisingly effective in practice; for example, it could easily disguise Mohamed Atta's leading position within the World Trade Center terrorist network. We also study how a community can increase the likelihood of being overlooked by community-detection algorithms. We propose a measure of concealment—expressing how well a community is hidden—and use it to demonstrate the effectiveness of a simple heuristic, whereby members of the community either 'unfriend' certain other members or 'befriend' some non-members in a coordinated effort to camouflage their community.

The ongoing process of datafication continues to turn many aspects of our lives into computerized data[1]. These data are being collected and analysed for various diverse applications by both public and private institutions. An important type of such data concerns our social connections[2–4]. To date, a wide variety of methods have been proposed in the literature for mining and analysing these data[5]. To this end, one of the most widely used methods involves analysing the topology of social networks using graph-theoretic tools, with particular emphasis on detecting communities or identifying key individuals within the network.

For all their benefits, social network analysis tools raise legitimate privacy concerns[6–9]. For instance, it has been demonstrated[10] how, by analysing the topology of Facebook's social network, as well as the attributes of some Facebook users, it is possible to infer otherwise-private information about other Facebook users. To tackle such privacy concerns, various countermeasures have been suggested, ranging from strict legal controls, through a variety of network modification techniques and algorithmic solutions[9,11,12], to market-like mechanisms that allow participants to monetize their personal information[13]. However, to date, only a few such countermeasures have been implemented on a large scale, leaving the privacy issue largely unresolved, as is evident, for example, from Facebook's Global Government Requests Report, which revealed a global increase in government requests to secretly access user data. Furthermore, it is unlikely that effective legal mechanisms will be introduced in countries with authoritarian regimes, where social networking sites and other internet content are policed and anti-governmental blogs and activities are censored[14,15].

Against this background, we ask the question 'Can individuals or communities proactively manage their social connections so that their privacy is less exposed to the workings of graph-theoretic network analysis tools?' To put it differently, 'Can individuals or groups disguise their standing in the network to escape detection?' This matters because, on the one hand, it assists the general public in protecting their privacy against intrusion from government and corporate interests, while on the other hand, it assists counter-terrorism units and law-enforcement agencies in understanding how criminals and terrorists could escape detection, especially given the increasing reliance of terrorists on social media survival strategies[16,17]. To date, however, this fundamental question has received little attention in the literature, as most research efforts have focused on developing ever-more sophisticated graph-theoretic network analysis tools, without considering how such tools can be evaded.

To address the above question from an individual's perspective, we focus on three fundamental measures of node centrality—namely, degree, closeness and betweenness. We study how an individual can avoid being highlighted by these measures while minimizing the negative impact that this may have on his or her influence within the network. Since, from a graph-theoretic perspective, this is fundamentally an optimization problem, we analyse its computational complexity to illuminate the theoretical limits of such capability as disguising oneself. We prove that an optimal solution is indeed difficult to compute. Despite this hardness, we demonstrate that even a simple heuristic can be surprisingly effective (albeit not optimal) in practice. Our heuristic involves rewiring some of the social connections within the individual's immediate network neighbourhood. Importantly, this requires two types of action that are already available on popular social media platforms: (1) 'unfriending' a certain friend; and (2) introducing two friends to each other. As such, our heuristic can readily be applied by members of the general public.

From a group's viewpoint, we study how a community can conceal itself to increase the likelihood of being overlooked by community-detection algorithms. To this end, we propose a measure

[1]Institute of Informatics, University of Warsaw, Warsaw, Poland. [2]Department of Computer Science, Khalifa University of Science and Technology, Abu Dhabi, UAE. [3]Department of Computer Science, University of Oxford, Oxford, UK. *e-mail: tpm@mimuw.edu.pl; trahwan@masdar.ac.ae

of concealment, designed to quantify the degree to which a group of individuals is hidden. Using this measure, we demonstrate the effectiveness of a simple heuristic, whereby members of the community either 'unfriend' certain other members or 'befriend' some non-members in order to blend in with the surrounding web of social connections.

Our findings suggest that counter-terrorism units may benefit from developing tools that identify not only the individuals and groups whose ranking (according to any measure of choice) is high, but also those whose ranking increases suspiciously and unexpectedly after making just a few modifications to the network. Such tools would contribute towards addressing the need to develop specialized software packages that account for more than just a snapshot of the network topology[18–20].

Our study also contributes to the literature on privacy preservation in online social networks[21,22]. This literature is primarily concerned with identifying privacy threats and possible countermeasures, both in generic social media (for example, Facebook[6]) and domain-specific social media (for example, health-related sites[23,24]). Perhaps the most widely studied problem in this literature is that of anonymization[9,11,25], where the network under consideration is anonymized and the goal is to prevent an adversary from exposing the true identities of the network users. In our case, the goal is to hide not the name but rather the role of a user, be it leadership or membership to a certain community.

Another line of research to which our study contributes is that on the secrecy–efficiency trade-off[26,27], where secrecy in our case is measured in terms of centrality ranking (the lower the better) and efficiency is measured in terms of influence (the greater the better). Viewed from a different perspective, our work can be seen as an extension of the sensitivity analyses of centrality measures[28] and community-detection algorithms[29]; while such analyses typically consider random network alterations, we focus on the effects of strategic changes.

Results

We consider a general model, defined by the following tuple: $(G, \mathcal{T}, V^{\dagger}, \mathcal{A})$. Here, $G=(V,E)$ denotes a network (be it directed or undirected) of which the set of nodes is V and the set of links is E. Moreover, $V^{\dagger} \subseteq V$ denotes a set of 'evaders' equipped with a set of graph-modifying actions, $\mathcal{A}$. Finally, $\mathcal{T}$ denotes a set of graph-theoretic social network analysis tools available to the 'seeker'. The goal of the evader(s) is to use the actions in $\mathcal{A}$ in order to become less exposed by $\mathcal{T}$. The choice of actions to take may be subject to certain constraints; for example, the evader(s) may want to avoid being disconnected entirely from the network. We assume that the seeker's set of tools $\mathcal{T}$ is known to the evader(s). We also assume that the seeker analyses only the topology of the network. Finally, we assume that the seeker is unaware of the evasion efforts made by the evader(s); that is, he or she analyses the network after it has been modified by the evader(s).

We study two instances of the aforementioned model. The first focuses on settings in which the following holds: (1) the set $V^{\dagger}$ contains exactly one node, called $v^{\dagger}$; (2) the actions in $\mathcal{A}$ are modifications of the network links, with each action being either an addition or a removal of a single link; and (3) the set $\mathcal{T}$ consists of three centrality measures: degree, closeness and betweenness[30]. The objective is then to conceal the importance of $v^{\dagger}$ by decreasing its centrality (according to the measures in $\mathcal{T}$) while minimizing the impact that this may have on the influence of $v^{\dagger}$ (according to two fundamental models of influence—namely, independent cascade[31] and linear threshold[32]). To this end, the evader must choose which actions to perform from $\mathcal{A}$ without exceeding a certain budget, which specifies the maximum number of links allowed to be modified. To simplify our analysis, we divide the process of disguising $v^{\dagger}$ into two consecutive phases. The first involves solving what we call the problem of disguising centrality, whereby some of the budget is spent on minimizing the centrality of $v^{\dagger}$ according to the measures in $\mathcal{T}$. Since this first phase is likely to decrease the influence of $v^{\dagger}$, the second phase involves spending the remaining budget to recover as much as possible of the influence of $v^{\dagger}$ while avoiding the addition of any links that were removed during the first phase. We consider two variants of this problem: (1) the 'individual influence recovery' problem, where the goal is to recover the original influence of $v^{\dagger}$ over every single node; and (2) the 'global influence recovery' problem, where the goal is to recover the sum of influences of $v^{\dagger}$ over all nodes.

The second instance of our model that we study is the one where: (1) the set $V^{\dagger}$ consists of multiple nodes; (2) the actions in $\mathcal{A}$ are modifications of the network links; and (3) the set $\mathcal{T}$ consists of the following community-detection algorithms: Louvain[33], Eigenvector[34], Betweenness[35], Walktrap[36], Greedy[37], Infomap[38] and Spinglass[39]. Each such algorithm returns a community structure, CS, which is a partition of the set of nodes into disjoint and exhaustive subsets or 'communities'. As such, $V^{\dagger}$ is completely exposed as a group if $V^{\dagger} \in \mathrm{CS}$. The objective of $V^{\dagger}$ is to then avoid such exposure by rewiring the links of the network without exceeding a certain budget that specifies the maximum number of permitted modifications.

Disguising individuals. *Hardness results.* Our theoretical analysis shows that finding an optimal way to disguise one's importance in a social network is often extremely difficult from a computational point of view. As shown in Table 1, while the problem of minimizing degree centrality belongs to the complexity class known as P (that is, solvable in polynomial time), each of the remaining problems under consideration is NP–hard, that is, it is at least as hard as any NP (non-deterministic polynomial time) problem, which implies that no known algorithm can solve it in polynomial time. Despite these results, the situation may not be entirely hopeless, provided that the evader is content with a reasonable, albeit not optimal, solution.

A practical heuristic. Typically, one has very limited knowledge of the social ties beyond his or her friends, or maybe friends of friends. However, even if one was able to somehow acquire information about the entire network topology, our hardness results suggest that it is extremely unlikely for such an individual to have the necessary computational power to optimally disguise himself or herself. Against this background, we investigate the possibility of disguising one's centrality adequately (albeit not optimally) while restricting one's attention to only his or her immediate network neighbourhood, and without requiring massive computational power nor expertise in sophisticated optimization techniques. With this in mind, we propose a heuristic whose instructions are simple enough for an average user of social media to understand and use, regardless of their technical background. Given a budget b, our heuristic—'remove one, add many' (ROAM)—works via the following two steps: (1) remove the link between the evader, $v^{\dagger}$, and its neighbour

Table 1 | Summary of our computational-hardness results

Disguising centrality (degree)	P
Disguising centrality (closeness)	NP-complete
Disguising centrality (betweenness)	NP-complete
Individual influence recovery (linear threshold)	NP-hard
Individual influence recovery (independent cascade)	NP-hard
Global influence recovery (linear threshold)	NP-hard
Global influence recovery (independent cascade)	NP-hard

of choice, v_0; and (2) connect v_0 to $b-1$ nodes of choice, who are neighbours of $v^\dagger$ but not of v_0 (if there are fewer than $b-1$ such neighbours, connect v_0 to all of them).

Figure 1 shows how the algorithm can disguise the leading position of Mohamed Atta—one of the ringleaders of the 9/11 attack[40]—within the World Trade Center terrorist network, and that is by only rewiring a small number of connections.

Let us now comment on this heuristic, starting with step 1. As far as the centrality of $v^\dagger$ is concerned, this step can only be beneficial. More specifically, cutting off $v^\dagger$ from one of its neighbours is the only way to reduce the degree of $v^\dagger$. Likewise, step 1 can only decrease the closeness of $v^\dagger$ (this happens when all shortest paths between $v^\dagger$ and some other node run through the removed link) and can only decrease the betweenness of $v^\dagger$ (this happens when some of the shortest paths going through $v^\dagger$ contain the removed link). However, as far as the influence of $v^\dagger$ is concerned, step 1 may be detrimental, as it deprives $v^\dagger$ from its direct influence over v_0.

Moving on to step 2, this step is primarily designed to compensate for any influence that $v^\dagger$ may have lost during the previous step. Specifically, it creates new, indirect connections between $v^\dagger$ and v_0 to compensate for the direct one that was removed earlier. As far as the centrality of $v^\dagger$ is concerned, while step 2 does not affect the degree of $v^\dagger$, it increases the degrees of some of its neighbours, which in turn contributes towards concealing the relative importance of $v^\dagger$ within the network. Furthermore, the addition of a link, (v_0,v_i)—where v_i is some neighbour of $v^\dagger$—cannot increase the closeness centrality of $v^\dagger$ beyond its original state; that is, its state before running the ROAM heuristic altogether. This is because any path containing (v_0,v_i) and $(v_0,v^\dagger)$ is certainly longer than an original path in which (v_0,v_i) and $(v_0,v^\dagger)$ were replaced with $(v_0,v^\dagger)$. Likewise, the addition of this link cannot increase the betweenness centrality of $v^\dagger$ beyond its original state because replacing a direct connection between $v^\dagger$ and v_0 with an indirect one cannot increase the percentage of shortest paths going through $v^\dagger$.

In a directed network, step 1 removes the edge(s) between $v^\dagger$ and v_0, whereas step 2 adds bidirected edges between v_0 and each of the chosen $b-1$ neighbours of $v^\dagger$.

Finally, let us comment on how to choose v_0 and how to choose the $b-1$ neighbours of $v^\dagger$ to connect to v_0. Based on the simulation study reported in the Supplementary Materials, we choose v_0 to be the neighbour of $v^\dagger$ with the most connections and we connect v_0 to the $b-1$ neighbours of $v^\dagger$ with the least connections. With such choices, it is straightforward to execute the ROAM heuristic on some leading social media platforms. Facebook, for example, provides a list displaying one's friends, as well as the number of connections that each of those friends has, except for those who make this information private. Hence, it is straightforward to choose v_0 as the most connected friend in the list (out of all those whose number of connections is visible) and to choose the remaining $b-1$ friends as the least connected ones. After that, step 1 simply requires $v^\dagger$ to 'unfriend' v_0, whereas step 2 requires $v^\dagger$ to 'suggest' the friendship of v_0 to the other chosen nodes. Note that, on Facebook, $v^\dagger$ can only introduce two individuals to each other if they were both friends of $v^\dagger$. As such, step 1 must be executed after step 2; that is, $v^\dagger$ must terminate the friendship with v_0 after introducing v_0 to the other nodes.

Note that the heuristic requires the cooperation of v_0. However, if we relax this requirement, $v^\dagger$ will not be able to take any action other than 'unfriending' some of its neighbours, which greatly limits the possible strategies that $v^\dagger$ can use to hide itself. Also, note that the heuristic never takes an action that reverses any of its previous actions, as it only removes links that include $v^\dagger$ and adds links that do not. As such, the heuristic does not require the evader to keep track of past modifications to the network.

Figure 2 depicts the evader's ranking after each execution of ROAM (see Methods for experimental details). As can be seen, the heuristic is able to decrease the evader's ranking according to four centrality measures (degree, closeness, betweenness and eigenvector) with varying levels of success, depending on both the network at hand and the budget being spent on rewiring the network.

Next, we evaluate ROAM in terms of recovering (some of) the influence that the evader, $v^\dagger$, lost during the evasion process. More precisely, we evaluate the effectiveness of step 2, whose main purpose is to recover the influence that $v^\dagger$ lost during step 1 of the heuristic. To this end, it suffices to calculate the influence of $v^\dagger$ given different budgets. To see why this is the case, recall that the budget, b, is spent as follows: one modification is spent in step 1, while the remaining $b-1$ modifications are spent in step 2. This basically means that, by setting $b=1$, we effectively disable step 2. Conversely, by increasing b, we increase the impact of step 2, and thus we expect the evader's influence to increase accordingly. To verify this, we plot the evader's relative influence value (compared with his or her original influence value before executing ROAM). This value was measured according to two alternative models of influence—namely, independent cascade and linear threshold (see Fig. 3). As expected, when $b=1$, the evader's influence decreases steadily, since step 2 is disabled. Conversely, when $b>1$, step 2 is activated and some of the evader's lost influence is recovered as a result (see how the recovery improves with the budget). Better still, in some of our experiments, when $b>3$, the influence of $v^\dagger$ actually exceeds its original value (that is, its value before executing ROAM altogether). This means that ROAM is not only able to hide the evader, but may even boost the evader's influence, depending on the network and the budget at hand.

Similar trends were observed when testing ROAM given other directed and undirected networks and other centrality measures taken from UCINET[41] (see Supplementary Materials), all of which

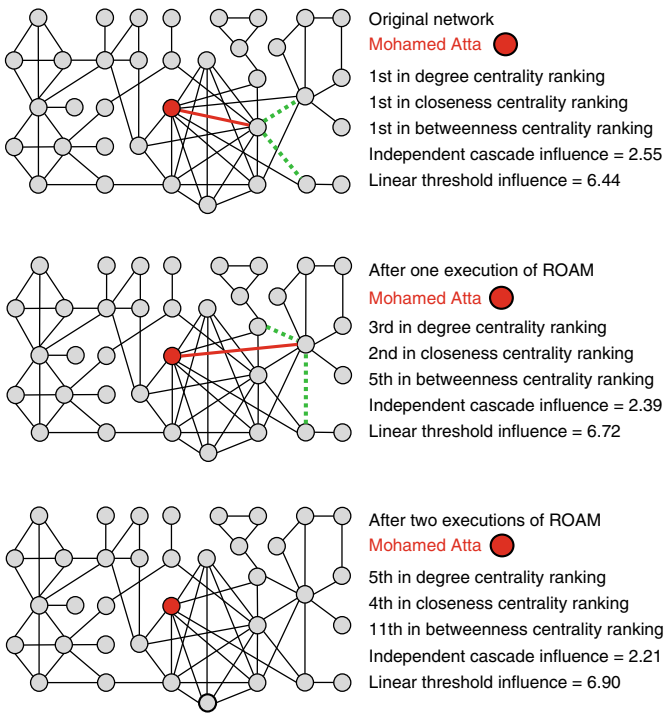

Fig. 1 | Executing the ROAM heuristic twice on the World Trade Center 9/11 terrorist network. The goal is to hide Mohamed Atta—one of the ringleaders of the attack. The red link is the one to be removed by the heuristic and the dashed links are the ones to be added.

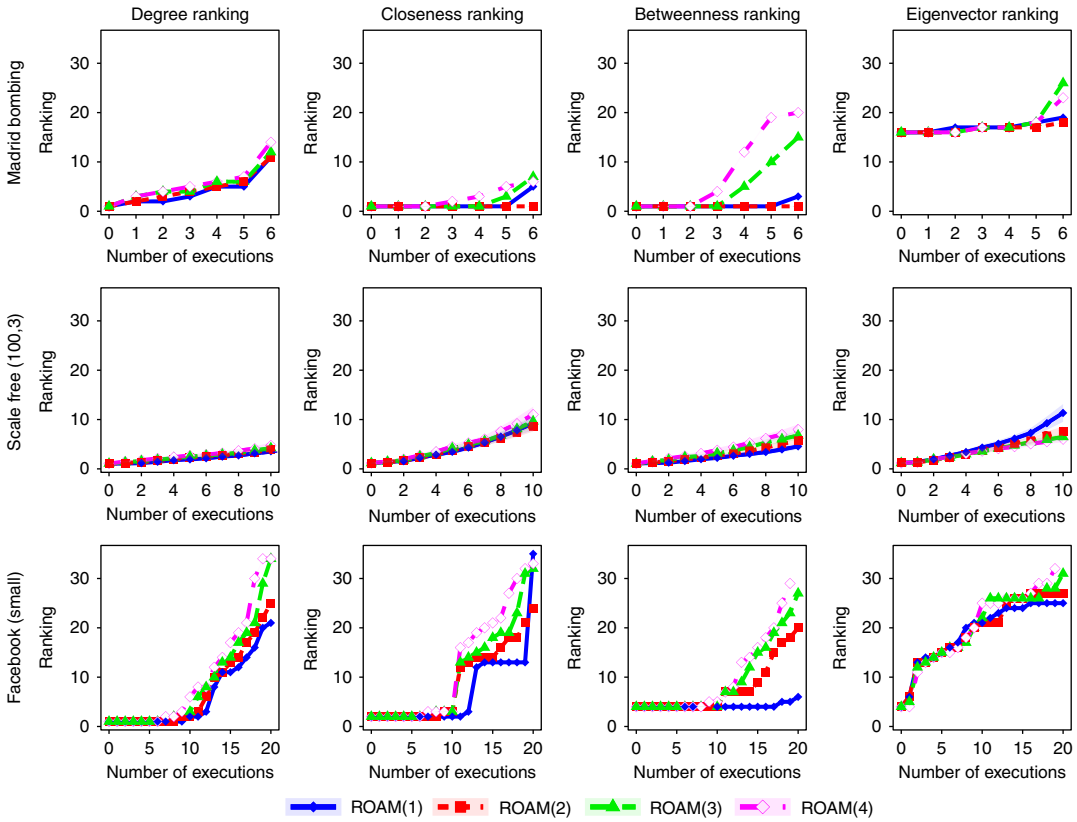

Fig. 2 | Executing ROAM multiple, consecutive times. The panels show how the evader's ranking is affected by the repeated execution of ROAM(b), where $b = 1, 2, 3$ or 4 is the budget in each execution. The results are shown for different ranking methods (based on degree, closeness, betweenness or eigenvector centrality) and for different types of networks; that is, the Madrid-attack network; 50 scale-free networks (each having 100 nodes and 3 links added with each node); or a small-sized fragment of Facebook's network (61 nodes and 272 links). The x axis represents the number of executions and the y axis represents the evader's ranking. Shaded areas for the scale-free networks represent 95% confidence intervals.

further demonstrate the universality of our findings across different networks and centrality measures.

Disguising communities. *A measure of concealment.* We propose a measure of how well a group of evaders, $V^\dagger$, is hidden in a community structure, CS. To this end, we start by proposing two measures, denoted by μ' and μ'', which capture different aspects of concealment. In particular, μ' is defined for every group of evaders, $V^\dagger \subseteq V$, and every community structure, CS, as follows:

$$\mu'(V^\dagger, \mathrm{CS}) = \frac{|\{C_i \in \mathrm{CS} : C_i \cap V^\dagger \neq \emptyset\}| - 1}{\max(|\mathrm{CS}| - 1, 1) \max\limits_{C_i \in \mathrm{CS}} (|C_i \cap V^\dagger|)}$$

Basically, this measure focuses on how well the members of $V^\dagger$ are spread out across the communities in CS. In more detail, we have $\mu'(V^\dagger, \mathrm{CS}) \in [0, 1]$ and the greater $\mu'(V^\dagger, \mathrm{CS})$, the greater the concealment of $V^\dagger$ in CS. Note that the numerator grows linearly with the number of communities that $V^\dagger$ is distributed over. Subtracting 1 from both the numerator and the $|\mathrm{CS}|$ term of the denominator is meant to handle the worst case, where all members of $V^\dagger$ appear in a single (possibly larger) community in CS; in this case, we have: $\mu'(V^\dagger, \mathrm{CS}) = 0$. In contrast, the term $\max_{C \in \mathrm{CS}}(|C \cap V^\dagger|)$ increases the concealment measure for such community structures in which the members of $V^\dagger$ are more evenly distributed across different communities. As such, the maximum concealment is achieved when the members of $V^\dagger$ are uniformly distributed, with each member appearing in a separate community; in this case: $\mu'(V^\dagger, \mathrm{CS}) = 1$.

Moving on to the second measure, μ'' is defined as:

$$\mu''(V^\dagger, \mathrm{CS}) = \sum_{C_i \in \mathrm{CS}} \frac{|C_i \setminus V^\dagger|}{\max(n - |V^\dagger|, 1)}$$

where n is the number of nodes in the network. Intuitively, μ'' focuses on how well $V^\dagger$ is 'hidden in the crowd'. It grows linearly with the number of non-members of $V^\dagger$ that appear with members of $V^\dagger$ in the same community in CS. Note that $\mu''(V^\dagger, \mathrm{CS}) \in [0, 1]$ and the greater the value, the greater the concealment of $V^\dagger$ in CS.

Having defined both μ' and μ'', we now use the two as building blocks to construct a single measure whereby the trade-off between μ' and μ'' is controlled by a parameter, $\alpha \in [0, 1]$. More formally, our proposed measure of concealment of a group of evaders, $V^\dagger$, in a community structure, CS, is:

$$\mu(V^\dagger, \mathrm{CS}) = \alpha\mu'(V^\dagger, \mathrm{CS}) + (1 - \alpha)\mu''(V^\dagger, \mathrm{CS})$$

Figure 4 presents a sample network with three different community structures, and illustrates how the concealment of $V^\dagger$ differs from one community structure to another according to the measures μ', μ'' and μ.

A practical heuristic. We set out to develop a simple heuristic that can be applied by any group of people regardless of their technical background or their knowledge of the network topology. After all, it would be of limited use to have an exact algorithm that can only be understood or applied by optimization experts armed

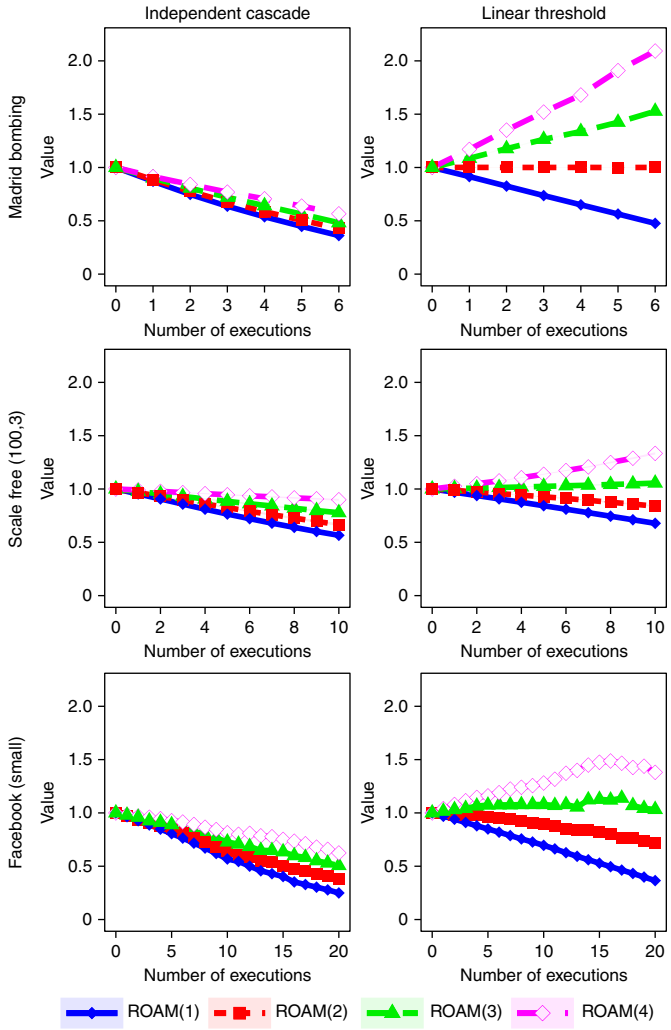

Fig. 3 | Relative change in the influence of the evader when executing ROAM multiple, consecutive times. Columns indicate the influence model and rows indicate the network(s); that is, the Madrid-attack network; 50 scale-free networks (consisting of 100 nodes each, constructed by adding 3 links with every node); or a small-sized fragment of Facebook's network (61 nodes and 272 links). Results are depicted for $ROAM(b) : b \in \{1, 2, 3, 4\}$, where b is the budget in each execution of ROAM. The x axis represents the number of executions. Shaded areas for the scale-free networks (almost invisible) represent the 95% confidence intervals.

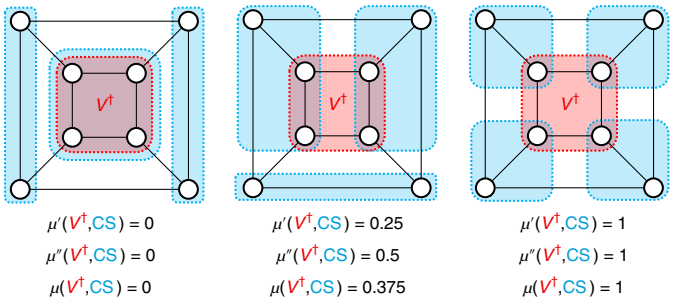

Fig. 4 | How the concealment of $V^{\dagger}$ differs from one community structure to another. Values are shown according to the measures μ', μ'' and μ with $\alpha = 0.5$.

with enormous processing power. Likewise, exact algorithms that require knowledge of the entire network topology may prove to be impractical, since such knowledge is rarely available. Our heuristic—'disconnect internally, connect externally' (DICE)—works via the following steps given a budget b: (1) disconnect $d \leq b$ links from within the community $V^{\dagger}$; and (2) connect $b - d$ nodes from within $V^{\dagger}$ to $b - d$ nodes from outside of $V^{\dagger}$.

This heuristic is inspired by modularity[35]—a widely used index for measuring the quality of any given community structure. More specifically, modularity promotes structures that have dense connections within communities and sparse connections between them. Importantly, community-detection algorithms are typically designed to search for a structure that maximizes modularity. Based on this, step 1 of our heuristic decreases the density of the connections within $V^{\dagger}$, whereas step 2 increases the connections between $V^{\dagger}$ and other communities. In so doing, a community-detection algorithm is more likely to overlook $V^{\dagger}$; that is, it would fail to recognize $V^{\dagger}$ as a community and would instead assign its members to multiple communities. The parameter d allows the group of evaders to control the trade-off between disguise and connectedness; increasing d sacrifices the group's connectivity in return for a better camouflage.

Let us comment on how DICE can be applied without the need for any tool support. On Facebook, for example, step 1 requires some members to 'unfriend' other members, which is rather straightforward. As for step 2, members must send a friendship request to non-members; these could be classmates, coworkers, neighbours living next door or even complete strangers (in fact, it is estimated that about 55% of people accept friendship requests from complete strangers[42]). Note that DICE can be executed without keeping track of any past modifications to the network, as it never takes actions that reverse previous actions.

We tested DICE given different networks and different community-detection algorithms (see Methods for experimental details). As shown in Fig. 5, DICE is able to hide the group of evaders, $V^{\dagger}$, with varying levels of success, depending on the evaders' budget and the seeker's community-detection algorithm. Surprisingly, the parameter d did not seem to have a significant impact on performance. The same observation was made when running DICE on a wider variety of networks and when testing DICE against a commercial tool for social network analysis—namely, UCINET[41] (see Supplementary Materials). This observation implies that the members of $V^{\dagger}$ can choose at their discretion how much of the budget is spent on removing internal links and how much is spent on adding external links without worrying about how this may affect their disguise. For example, the members of $V^{\dagger}$ might be interested in hiding as much as possible, while removing as few internal links as possible (after all, the added external links are fake, serving no purpose other than disguising the group of evaders, whereas the removed internal links are real; they existed within the group for a reason). However, since the addition of an external link is not entirely under the control of $V^{\dagger}$ (as it requires the consent of a non-member), the number of newly added external links may be insufficient for providing a satisfactory level of concealment, in which case the members can compensate for this by sacrificing more internal links; that is, by increasing the parameter d.

Figure 6 illustrates the average value of the concealment measure μ with $\alpha = 0.5$, taken over all of our experiments of DICE where $b = 4$ and $d = 2$. In particular, each row represents a community-detection algorithm, each column represents a network and the intensity of the colour in each cell represents the average value of μ, taken over 50 simulations, either by generating a new random network in each simulation or by re-running the simulation over and over on the same real-life network (note that our implementation of DICE is non-deterministic and may yield different results on the same network). The missing cells correspond to the cases

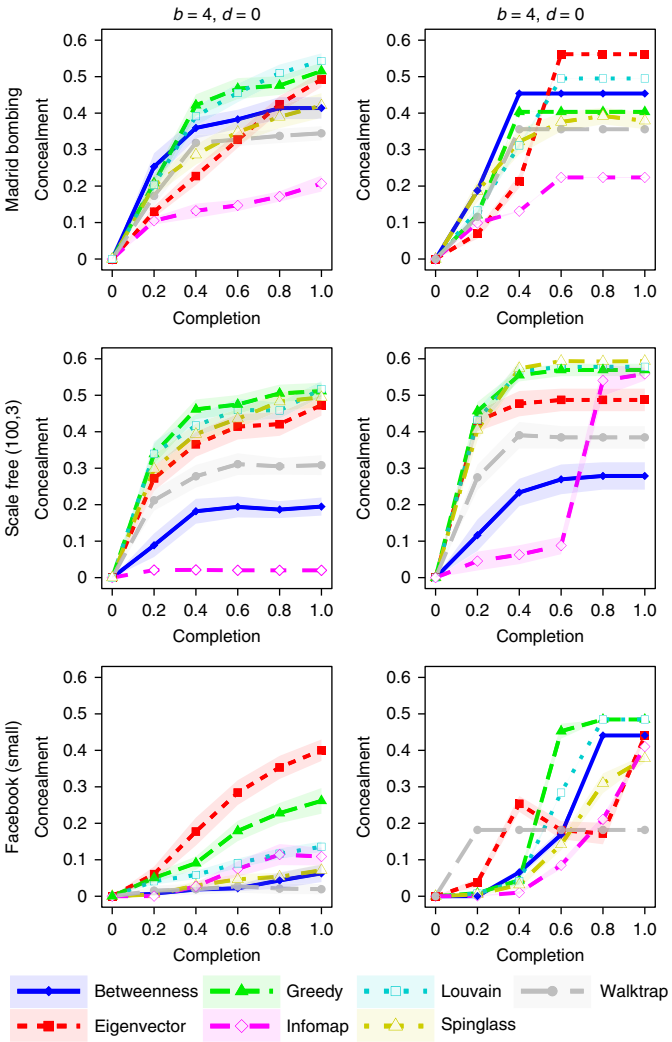

Fig. 5 | Average concealment level of the group of evaders, $V^{\dagger}$, during the execution of DICE. Results are from 50 experiments, each involving the execution of DICE multiple, consecutive times (the *x* axis represents the percentage of completed rounds and the *y* axis represents the concealment level according to the measure μ with $\alpha = 0.5$). Columns indicate parameter values for DICE and rows indicate the network(s); that is, the Madrid-attack network; 50 scale-free networks (consisting of 100 nodes each, constructed by adding 3 links with every node); or a small-sized fragment of Facebook's network (61 nodes and 272 links). Each panel depicts the concealment level of $V^{\dagger}$ in each of the community structures that is identified by the following community-detection algorithms: Eigenvector[34], Betweenness[35], Walktrap[36], Louvain[33], Greedy[37], Infomap[38] and Spinglass[39]. The shaded areas represent 95% confidence intervals.

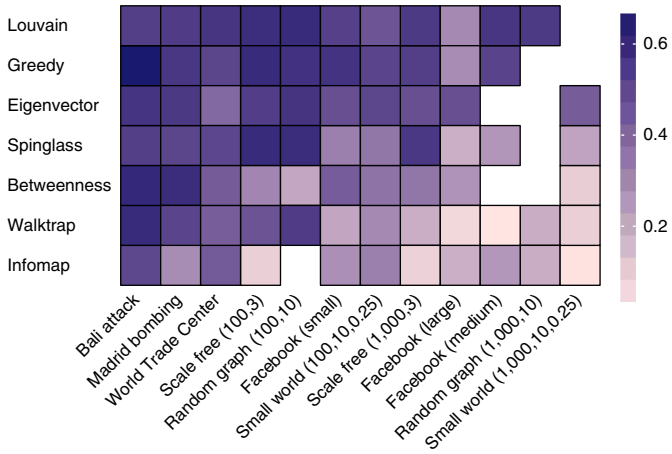

Fig. 6 | Average concealment level of the group of evaders, $V^{\dagger}$, after the execution of DICE. Results are from 50 experiments with DICE where $b = 4$ and $d = 2$. Columns correspond to the network and rows correspond to the community-detection algorithm; that is, Eigenvector[34], Betweenness[35], Walktrap[36], Louvain[33], Greedy[37], Infomap[38] or Spinglass[39]. For every network and community-detection algorithm, DICE was executed $|V^{\dagger}|$ times, after which the concealment of $V^{\dagger}$ was measured in the community structure that was identified by the algorithm. The cell colour represents the concealment level according to the measure μ with $\alpha = 0.5$.

where $V^{\dagger}$ happened to be either extremely small or extremely large (see Methods). Our results show that the Infomap algorithm[38] is, on average, the most difficult to fool given undirected networks and the easiest to fool given directed networks. The same observation was made when testing DICE on other directed and undirected networks (see Supplementary Materials).

Discussion

Our goal was to understand the practical limits of disguising individuals and communities by increasing the likelihood of them being overlooked by graph-theoretic network analysis tools. Our main result is that, despite the difficulty of finding an optimal solution, disguise can be surprisingly easy in practice, using simple heuristics that can be applied even by lay people. As our experiments have demonstrated, by strategically rewiring a relatively small number of links, it is possible for individuals or groups to significantly alter their standing within a social network. In contrast, our results highlight the fragility of basic graph-theoretic tools against strategic evaders. We also demonstrate the tremendous power of dummy links, which can easily be created by criminals and terrorists. This highlights the potential risk of accepting a friendship request from a complete stranger, which is especially alarming since an estimated 55% of people accept friendship requests from strangers[42] and some of those requests may have malicious intent.

In our study, we focused on basic models in which the goals are to evade some of the most widely used centrality measures and community-detection algorithms. Nevertheless, our approach has a number of limitations, which will be discussed next.

First, while our assumption of a non-strategic seeker may seem rather strong, we believe it corresponds well to the current state of the art in the literature and industrial practices, since most graph-theoretic tools implicitly assume that individuals or groups in a network do not act strategically to evade those tools on purpose. Our work can be considered as a preliminary step towards relaxing this assumption. More specifically, by showing that individuals can easily disguise their centrality, we highlight the need to look beyond just the centrality-based ranking of individuals. For instance, the seeker may benefit from taking snapshots of the network and analysing those snapshots in order to identify any individual whose centrality value has dropped abnormally. One way to develop such a tool would be to apply anomaly-detection algorithms[43,44] to a dataset in which every data point corresponds to a node in a multidimensional space and every dimension corresponds to the temporal change in ranking according to a particular centrality measure. Such a measure could be any of the standard ones or it could be a measure designed specifically for dynamic networks (see, for example, refs [45,46]). As far as groups are concerned, we have shown that they can easily evade community-detection algorithms by disconnecting some intra-group links and creating some dummy inter-group links. The fact that such dummy links can easily be created highlights the need to extend the existing community-detection algorithms so as

to take into consideration the possibility that the perceived network is different from the actual one. For instance, one way to develop such an extension would be to create tools that identify not only the links that are missing from the network (as is typically the case with link-prediction algorithms[47]), but also the links that might have been added by a strategic evader.

Second, our model assumes that the seeker's knowledge is restricted to the topology of the network. The motivation behind this assumption is twofold: (1) many social network analysis tools—including the ones studied here—are all based solely on the topology of the network; (2) the exclusion of domain knowledge makes the model more general as it can be applied to any network. Nevertheless, there are cases where the seeker has additional, domain-dependent information that might be used in conjunction with graph-theoretic tools; for example, as in the case of covert networks[48]. In such cases, further investigation is needed to understand the extent to which the evaders can protect themselves against the seeker.

Third, while our heuristic algorithms—namely ROAM and DICE—seem to be effective in practice, they do not provide any worst-case guarantees on solution quality. This is because they were primarily designed to be scalable and applicable even by lay individuals who typically do not know the topology of the entire network nor have the ability to rewire links between two complete strangers. Undoubtedly however, scalability and applicability are achieved at the expense of solution quality. As such, there is room to develop advanced algorithms that require more operations, but provide higher-quality solutions. For instance, one can develop an advanced version of ROAM that optimizes the choice of v_0 as well as the choice of the $b-1$ neighbours of $v^\dagger$ to connect to v_0. This can be done in polynomial time since there are at most $\deg(v^\dagger)\binom{\deg(v^\dagger)-1}{b-1}$ such choices, where $\deg(v^\dagger)$ denotes the degree of $v^\dagger$. Likewise, one can develop an advanced version of DICE that optimizes the choices of the links to be removed from within $V^\dagger$—there are at most $\binom{|V^\dagger|(|V^\dagger|-1)/2}{d}$ such choices—as well as the choices of the links to be added between members and non-members of $V^\dagger$—there are at most $\binom{|V^\dagger||V|/2}{b-d}$ such choices. However, lay individuals who wish to apply such advanced algorithms would probably require tool support.

Fourth, there are many challenges that are likely to face any group of evaders wishing to apply our heuristics. Suppose that this group is discriminated against and its members wish to conceal their membership in order to avoid prejudice. Our heuristic would then require some members to end their relationship with some other members. While ending a relationship can be achieved with the click of a button in the virtual world, the situation is not that simple outside the realm of social media; if two members of a community must end their relationship, this would require them to never meet again, nor make any contact, be it through phone calls, email exchanges or otherwise. Our heuristic also requires some members to create new relationships with non-members. Again, while the creation of such relationships is relatively easy in the virtual world, this can be very demanding outside the realm of social media. For instance, the group member(s) tasked with creating new relationships may have to invest significant time and effort creating believable and seemingly genuine relationships with non-members of the group. Befriending a non-member must be handled with care or else it may backfire, as the relationship may entail bringing the non-member closer to the evader and may accidentally result in leaking certain confidential information, thereby compromising the evader's disguise and revealing his or her membership. To circumvent this issue in social media, whenever such a relationship is created, the evader must restrict the privileges of the non-member to ensure that he or she can only access publicly available information about the evader. Unfortunately, outside the realm of social media, the potential risk from befriending a non-member cannot be circumvented that easily. As such, caution must be exercised when creating such a relationship; for example, by keeping the non-member close, but not too close.

On a broader note, the questions that are addressed in this article can be generalized as follows: 'Given a seeker equipped with a set of graph-theoretic network analysis tools $\mathcal{T}$, and given some evader(s) equipped with a set of actions $\mathcal{A}$, which of those actions should the evader(s) choose in order to evade $\mathcal{T}$; how hard is it to make those choices; and how effective would they be against $\mathcal{T}$?'

In this article, the set of graph-theoretic network analysis tools, T, consisted of node centrality measures or community-detection algorithms. There are still many interesting instances of the above question that are yet to be studied. For instance, we still do not know how to handle settings in which communities may overlap[49], nor do we know how to hide a relationship from the eyes of link-prediction algorithms[47].

Methods

Centrality measures. A measure of centrality reflects the importance of any given node in any given network. Arguably, the standard centrality measures are: degree, closeness, betweenness and eigenvector[30]. Next, we briefly introduce each of these centralities.

The degree centrality[50] focuses on the number of neighbours that a node has (the more neighbours the better); it is formally defined for a node, v_i, in a network, G, as:

$$c_{\mathrm{degr}}(G, v_i) = \frac{|N_G(v_i)|}{n-1}$$

where $N_G(v_i)$ is the set of neighbours of node v_i, and n is the number of nodes in the network.

The closeness centrality[51] quantifies the importance of a node based on its average distance to other nodes (the closer the better); it is formally defined for a node, v_i, in a network, G, as:

$$c_{\mathrm{clos}}(G, v_i) = \frac{n-1}{\sum_{v_j \in V} d_G(v_i, v_j)}$$

where $d_G(v_i,v_j)$ is the length of the shortest path between the nodes v_i and v_j. The betweenness centrality[52,53] considers the number of shortest paths on which a node lies (the more paths the better); it is defined for a node, v_i, in a network, G, as:

$$c_{\mathrm{betw}}(G, v_i) = \gamma \sum_{v_j, v_k \in V \setminus \{v_i\}} \frac{|\{p \in sp_G(v_j, v_k) : v_i \in p\}|}{|sp_G(v_j, v_k)|}$$

where γ is equal to $2/((n-1)(n-2))$ and $sp_G(v_j, v_k)$ is the set of shortest paths between the nodes v_j and v_k.

The eigenvector centrality[54] quantifies a node's importance based on the importance of its neighbours. More formally, it is defined for a node, v_i, in a network, G, as:

$$c_{\mathrm{eig}}(G, v_i) = x_i$$

where x is the eigenvector corresponding to the largest eigenvalue of the adjacency matrix of the network.

Models of influence. Arguably, the best established mathematical models of influence are the independent cascade model[31] and the linear threshold model[32]. Both of these models start with some 'active' subset of nodes called the seed set. Then, as time passes, new nodes become activated due to the influence from other previously activated nodes. Assuming that time moves in discrete rounds, we denote by $I(t) \subseteq V$ the set of nodes that are active at round t, implying that $I(1)$ is the seed set. The way influence propagates from the seed set to the remaining nodes depends on the influence model under consideration.

In the independent cascade model, every pair of nodes is assigned an activation probability, $p : V \times V \to [0, 1]$. Then, in every round, $t > 1$, every node $v \in V$ that became active in round $t-1$ activates every inactive neighbour, $w \in N_G(v) \setminus I(t-1)$, with probability $p(v,w)$. The process ends when there are no new active nodes; that is, when $I(t) = I(t-1)$.

In the linear threshold model, every node $v \in V$ is assigned a threshold value, t_v, which is sampled (according to some probability distribution) from the following set: $\{0, \ldots, |N_G(v)|\}$. Then, in every round, $t > 1$, every inactive node, v, becomes active; that is, it becomes a member of $I(t)$, if the following holds: $|I(t-1) \cap N_G(v)| \geq t_v$. The process ends when $I(t) = I(t-1)$.

In either model, the influence of a node, v, on another, w, is denoted by $inf_G(v, w)$ and is defined as the probability that w gets activated given the seed set $\{v\}$ (we make the common assumption that $inf_G(v, v) = 0$ for every $v \in V$). The influence of v over the entire network G is then: $inf_G(v) = \sum_{w \in V} inf_G(v, w)$. Since it is intractable to compute the exact influence value according to the independent cascade model or the linear threshold model, we approximate this value using Monte Carlo sampling and stop the process when the improvement over the last 1,000 iterations is smaller than 0.00001.

Experimental design. *Datasets.* We experiment with two different types of real-life network—namely, covert organizations and social networks. For covert organizations, we consider three terrorist networks responsible for the World Trade Center 9/11 attack[40], the 2002 Bali attack[55] and the 2004 Madrid train bombing[55], respectively. For social networks, we study anonymized fragments of three social networks—namely, Facebook, Twitter and Google+. These fragments are taken from the Stanford Network Analysis Platform[56]. We also study the following randomly generated networks. (1) Scale-free networks, generated using the Barabasi–Albert model[57]. We denote any such network by $ScaleFree(x, y)$, where x is the number of nodes and y is the number of links added with each node. For directed networks, we write $dScaleFree(x, y)$ and set the added links to be directed from each new node to the existing ones. (2) Small-world networks, generated using the Watts–Strogatz model[58]. We denote any such network by $SmallWorld(x, y, z)$, where x is the number of nodes, y is the average degree and z is the rewiring probability. For directed networks, we write $dSmallWorld(x, y, z)$ and take y to be the average out-degree. (3) Random graphs generated using the Erdos–Renyi model[59]. We write $RandomGraph(x, y)$, with x being the number of nodes and y being the expected average degree. As for directed networks, we write $dRandomGraph(x, y)$ and take y to be the expected average out-degree.

For each type of randomly generated network, we report the average result taken over 50 such networks, with the shaded areas representing 95% confidence intervals. The Supplementary Materials contain our results on all these networks, as well as other types of network; for example, financial transactions, telecommunications and co-membership networks.

Experimenting with ROAM. Each of our experiments consists of a network, budget, evader and influence model. More specifically, we experiment with budgets of 1, 2, 3 and 4. The evader, $v^\dagger$, is chosen as the node with the lowest sum of centrality rankings (based on degree, closeness and betweenness), where ties are broken uniformly at random. Whenever the independent cascade model is used, an activation probability of 0.15 is assumed on each link. In contrast, whenever the linear threshold model is used, a uniform distribution of thresholds is assumed (see Supplementary Materials for more details). For both models, the influence values are approximated using the Monte Carlo method. In each of these experiments, the ROAM heuristic is executed multiple, consecutive times to see how this affects the centrality and influence of the evader.

As shown in the Supplementary Materials, we studied the case where some nodes from the evader's neighbourhood keep their connections private (that is, not visible to the evader). As expected, this lack of information affects the performance of ROAM, but the impact is often negligible. More specifically, in terms of influence recovery, the drop in performance seems negligible in about 70% of our experiments. Similarly, in terms of centrality minimization, the drop in performance appears to be negligible in about 80% our experiments. These results hold even when the majority of the evader's neighbours keep their connections private.

As is also shown in the Supplementary Materials, we studied the case in which multiple evaders exist (not just $v^\dagger$), each trying to decrease his or her centrality without any coordination with the remaining evaders. Overall, the closer the additional evaders are to $v^\dagger$, the greater their impact on the centrality and influence of $v^\dagger$. In many cases, when additional evaders are running ROAM, the process of lowering the centrality of $v^\dagger$ becomes less effective. Interestingly, in many of our experiments, the actions of the additional evaders increase, rather than decrease, the effectiveness of recovering the influence of $v^\dagger$.

Experimenting with DICE. For each network, we experiment with seven community-detection algorithms implemented in the *igraph* package of the *R* language (version 1.01)—namely, Eigenvector[34], Betweenness[35], Walktrap[36], Louvain[33], Greedy[37], Infomap[38] and Spinglass[39]. Every experiment consists of a network and a community-detection algorithm. The experiment starts by running the algorithm to obtain a community structure, CS. After that, the group of evaders; that is, $V^\dagger$, is chosen to be the element in CS whose size is the median of the sizes of all communities in CS (ties are broken uniformly at random). Although $V^\dagger$ does not necessarily have to be an element of CS, we choose it as such in order to study the worst-case scenario in which $V^\dagger$ is initially exposed completely by the algorithm. We proceed with the experiment only if $2 < |V^\dagger| < n-2$ to avoid extreme cases in which $V^\dagger$ happens to be either extremely small or extremely large (this explains the missing cells in Fig. 6).

The experiment proceeds in rounds, each involving the execution of DICE followed by the execution of the community-detection algorithm, to measure how well $V^\dagger$ is hidden in the new outcome of the algorithm; the measurement is taken using our measure of concealment, μ, with $\alpha = 0.5$. We set the number of rounds to be $|V^\dagger|$. In each round, we disconnect d links from within $V^\dagger$ (chosen uniformly at random) and then connect $b-d$ members of $V^\dagger$ to $b-d$ non-members of $V^\dagger$ (again chosen uniformly at random). Due to this randomness in our implementation, DICE may yield different results in different executions. Therefore, we repeat each experiment multiple times and report the 95% confidence intervals.

Finally, the Supplementary Materials include additional experiments in which DICE and ROAM are executed simultaneously, with the goal being to hide a community $V^\dagger$ (using DICE) and, at the same time, hide its leader $v^\dagger$ (using ROAM). In this case, the heuristics are still able to hide both the leader and the community with varying levels of success, but this comes at the expense of the evader's influence.

Life Sciences Reporting Summary. Further information on experimental design is available in the Life Sciences Reporting Summary.

Code availability. The code used to generate the results of this study is available from the corresponding authors upon request.

Data availability. The data that support the findings of this study are available from the corresponding authors upon request.

Received: 23 November 2016; Accepted: 21 December 2017;
Published online: 29 January 2018

References

1. Mayer-Schnberger, V. & Cukier, K. *Big Data: A Revolution That Will Transform How We Live, Work and Think* (John Murray Publishers, London, 2013).
2. Moreno, J. L. *Application of the Group Method to Classification* (National Committee on Prisons and Prison Labor, New York, NY, 1932).
3. Granovetter, M. S. The strength of weak ties. *Am. J. Sociol.* **78**, 1360–1380 (1973).
4. Freeman, L. C. *The Development of Social Network Analysis. A Study in the Sociology of Science* (Empirical Press, Vancouver, 2004).
5. Easley, D. & Kleinberg, J. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World* (Cambridge Univ. Press, Cambridge, 2010).
6. Gross, R. & Acquisti, A. Information revelation and privacy in online social networks. In *Proc. 2005 ACM Workshop on Privacy in the Electronic Society* 71–80 (ACM, 2005).
7. Zhang, C., Sun, J., Zhu, X. & Fang, Y. Privacy and security for online social networks: challenges and opportunities. *IEEE Netw.* **24**, 13–18 (2010).
8. Deliri, S. & Massimiliano, A. in *Data Management in Pervasive Systems* (eds Altshuler, Y., Elovici, Y., Cremers, A. B., Aharony, N. & Pentland, A.) 195–209 (Springer, New York, NY, 2013).
9. Wu, X., Ying, X., Liu, K. & Chen, L. in *Managing and Mining Graph Data* (eds Aggarwal, C. C. & Wang, H.) 421–453 (Springer, New York, NY, 2010).
10. Mislove, A., Viswanath, B., Gummadi, K. P. & Druschel, P. You are who you know: inferring user profiles in online social networks. In *Proc. 3rd ACM Int. Conf. on Web Search and Data Mining* 251–260 (ACM, New York, NY, 2010).
11. Zhou, B., Pei, J. & Luk, W. S. A brief survey on anonymization techniques for privacy preserving publishing of social network data. *SIGKDD Explor.* **10**, 12–22 (2008).
12. Kearns, M., Roth, A., Wu, Z. W. & Yaroslavtsev, G. Private algorithms for the protected in social network search. *Proc. Natl. Acad. Sci. USA* **113**, 913–918 (2016).
13. Lane, J. I., Stodden, V., Bender, S. & Nissenbaum, H. *Privacy, Big Data, and the Public Good: Frameworks for Engagement* (Cambridge Univ. Press, Cambridge, 2014).
14. King, G., Pan, J. & Roberts, M. E. How censorship in china allows government criticism but silences collective expression. *Am. Polit. Sci. Rev.* **107**, 326–343 (2013).
15. King, G.., Pan, J.. & Roberts, M. E.. Reverse-engineering censorship in china: randomized experimentation and participant observation. *Science* **345**, 1251722 (2014).
16. Nordrum, A. Pro-ISIS online groups use social media survival strategies to evade authorities. *IEEE Spectrum* https://spectrum.ieee.org/tech-talk/telecom/internet/proisis-online-groups-use-social-media-survival-strategies-to-evade-authorities (2016).
17. Johnson, N. F. et al. New online ecology of adversarial aggregates: ISIS and beyond. *Science* **352**, 1459–1463 (2016).
18. Carley, K. M., Lee, J.-S. & Krackhardt, D. Destabilizing networks. *Connections* **24**, 31–34 (2001).

19. Carley, K. M., Reminga, J. & Kamneva, N. Destabilizing terrorist networks. In *NAACSOS Conference Proceedings* (NAACSOS, Pittsburgh, PA, 2003).
20. McCulloh, I. & Carley, K. M. *Detecting Change in Longitudinal Social Networks* Technical Report (Defense Technical Information Center, 2011).
21. Cutillo, L. A., Molva, R. & Strufe, T. Safebook: a privacy-preserving online social network leveraging on real-life trust. *IEEE Commun. Mag.* **47**, 94–101 (2009).
22. Baden, R., Bender, A., Spring, N., Bhattacharjee, B. & Starin, D. Persona: an online social network with user-defined privacy.*Comput. Commun. Rev.* **39**, 135–146 (2009).
23. Li, J. Privacy policies for health social networking sites. *J. Am. Med. Inform. Assoc.* **20**, 704–707 (2013).
24. Li, J. A privacy preservation model for health-related social networking sites. *J. Med. Internet Res.* **17**, e168 (2015).
25. Narayanan, A. & Shmatikov, V. De-anonymizing social networks. In *Proc. 2009 30th IEEE Symposiumo n Security and Privacy* 173–187 (IEEE Computer Society, Washington DC, 2009).
26. Crossley, N., Edwards, G., Harries, E. & Stevenson, R. Covert social movement networks and the secrecy–efficiency trade off: the case of the UK suffragettes (1906–1914). *Social. Netw.* **34**, 634–644 (2012).
27. Stevenson, R. & Crossley, N. Change in covert social movement networks: the "inner circle" of the provisional Irish republican army. *Social. Mov. Stud.* **13**, 70–91 (2014).
28. Correa, C. D., Crnovrsanin, T. & Ma, K.-L. Visual reasoning about social networks using centrality sensitivity. *IEEE Trans. Vis. Comput. Graph.* **18**, 106–120 (2012).
29. Orman, G. K. & Labatut, V. in *Discovery Science* (eds Gama, J., Costa, V. S., Jorge, A. M. & Brazdil, P. B.) 242–256 (Springer, New York, NY, 2009).
30. Freeman, L. C. Centrality in social networks conceptual clarification. *Social. Netw.* **1**, 215–239 (1979).
31. Goldenberg, J., Libai, B. & Muller, E. Using complex systems analysis to advance marketing theory development: modeling heterogeneity effects on new product growth through stochastic cellular automata. *Acad. Mark. Sci. Rev.* **9**, 1–18 (2001).
32. Kempe, D., Kleinberg, J. & Tardos, É. Maximizing the spread of influence through a social network. In *Proc. 9th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining* 137–146 (ACM, New York, NY, 2003).
33. Blondel, V. D., Guillaume, J.-L., Lambiotte, R. & Lefebvre, E. Fast unfolding of communities in large networks. *J. Stat. Mech.* **2008**, P10008 (2008).
34. Newman, M. E. J. Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E* **74**, 036104 (2006).
35. Newman, M. E. J. & Girvan, M. Finding and evaluating community structure in networks. *Phys. Rev. E* **69**, 026113 (2004).
36. Pons, P. & Latapy, M. in *Computer and Information Sciences—ISCIS 2005* (eds Yolum, P., Güngör, T., Gürgen, F. & Özturan, C.) 284–293 (Springer, 2005).
37. Clauset, A., Newman, M. E. J. & Moore, C. Finding community structure in very large networks. *Phys. Rev. E* **70**, 066111 (2004).
38. Rosvall, M., Axelsson, D. & Bergstrom, C. T. The map equation. *Eur. Phys. J. Spec. Top.* **178**, 13–23 (2010).
39. Reichardt, J. & Bornholdt, S. Statistical mechanics of community detection. *Phys. Rev. E* **74**, 016110 (2006).
40. Krebs, V. E. Mapping networks of terrorist cells. *Connections* **24**, 43–52 (2002).
41. Borgatti, S. P., Everett, M. G. & Freeman, L. C. Ucinet for windows: software for social network analysis. *Connections* **15**, 12–15 (2002).
42. Nagle, F. & Singh, L. Can friends be trusted? Exploring privacy in online social networks. In *Proc. Int. Conf. on Advances in Social Network Analysis and Mining* 312–315 (IEEE, 2009).
43. Chandola, V., Banerjee, A. & Kumar, V. Anomaly detection: a survey. *ACM Comput. Surv.* **41**, 15 (2009).
44. Ahn, J., Plaisant, C. & Shneiderman, B. A task taxonomy for network evolution analysis. *IEEE Trans. Vis. Comput. Graph.* **20**, 365–376 (2014).
45. Lerman, K., Ghosh, R. & Kang, J. H. Centrality metric for dynamic networks. In *Proc. 8th Workshop on Mining and Learning with Graphs* 70–77 (ACM, 2010).
46. Federico, P., Pfeffer, J., Aigner, W., Miksch, S. & Zenk, L. Visual analysis of dynamic networks using change centrality. In *2012 IEEE/ACM Int. Conf. on Advances in Social Networks Analysis and Mining* 179–183 (IEEE, 2012).
47. Lü, L. & Zhou, T. Link prediction in complex networks: a survey.*Phys. A Stat. Mech. Appl.* **6**, 1150–1170 (2011).
48. Michalak, T., Rahwan, T., Skibski, O. & Wooldridge, M. Defeating terrorist networks with game theory. *IEEE Intell. Syst.* **30**, 53–61 2015).
49. Xie, J., Kelley, S. & Szymanski, B. K. Overlapping community detection in networks: the state-of-the-art and comparative study. *ACM Comput. Surv.* **45**, 43 (2013).
50. Shaw, M. E. Group structure and the behavior of individuals in small groups. *J. Psychol.* **38**, 139–149 (1954).
51. Murray, A. Beauchamp. An improved index of centrality. *Behav. Sci.* **10**, 161–163 (1965).
52. Anthonisse, J. M. *The Rush in a Directed Graph* (Univ. Amsterdam Mathematical Centre, Amsterdam, 1971).
53. Freeman, L. C. A set of measures of centrality based on betweenness. *Sociometry* **40**, 35–41 1977).
54. Bonacich, P. Power and centrality: a family of measures. *Am. J. Sociol.* **92**, 1170–1182 (1987).
55. Hayes, B. Connecting the dots: can the tools of graph theory and social-network studies unravel the next big plot? *Am. Sci.* **94**, 400–404 (2006).
56. Leskovec, J. & Mcauley, J. J. Learning to discover social circles in ego networks. In *Proc. 25th Int. Conf. on Neural Information Processing Systems* 539–547 (Curran Associates, 2012).
57. Barabási, A.-L. & Albert, R. Emergence of scaling in random networks. *Science* **286**, 509–512 (1999).
58. Watts, D. J. & Strogatz, S. H. Collective dynamics of small-world networks. *Nature* **393**, 440–442 (1998).
59. Erdös, P. & Rényi, A. On random graphs I. *Publ. Math. Debr.* **6**, 290–297 (1959).

Acknowledgements

M.W. was supported by the Polish National Science Centre (grant 2015/17/N/ST6/03686). M.J.W. was supported by the European Research Council under Advanced Grant 291528 ('RACE'). T.P.M. was supported by the Polish National Science Centre (grant 2014/13/B/ST6/01807) and, for the earlier versions of this article, also by the European Research Council under Advanced Grant 291528 ('RACE'). No funders had any role in study design, data collection and analysis, decision to publish or preparation of the manuscript.

Author contributions

T.P.M., M.J.W. and T.R. conceived the study and designed the experiments. M.W. and T.P.M. formalized the computational problems. M.W. and T.R. developed the heuristics. M.W. developed the proofs and performed the numerical simulations. All authors discussed the results and wrote the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information is available for this paper at https://doi.org/10.1038/s41562-017-0290-3.

Reprints and permissions information is available at www.nature.com/reprints.

Correspondence and requests for materials should be addressed to T.P.M. or T.R.

nature research

Corresponding author(s): Tomasz Michalak, Talal Rahwan

☐ Initial submission ☐ Revised version ☒ Final submission

Life Sciences Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form is intended for publication with all accepted life science papers and provides structure for consistency and transparency in reporting. Every life science submission will use this form; some list items might not apply to an individual manuscript, but all fields must be completed for clarity.

For further information on the points included in this form, see Reporting Life Sciences Research. For further information on Nature Research policies, including our data availability policy, see Authors & Referees and the Editorial Policy Checklist.

Experimental design

1. Sample size

Describe how sample size was determined.

Sampling was only done when testing our heuristics on random networks. In this case, we randomly generated multiple such networks, and reported the results along with the 95% confidence intervals, which indicated a trend clear enough for the purpose of our study.

2. Data exclusions

Describe any data exclusions.

When experimenting with the DICE heuristic, we excluded the cases where the community structures contained fewer than 3 or more than n-3 communities, to avoid extreme cases in which the hiding community happens to be either extremely small or extremely large. This is explicitly mentioned in the paper.

3. Replication

Describe whether the experimental findings were reliably reproduced.

All results are software-generated, and all details necessary to replicate the results have been clearly provided in the submission.

4. Randomization

Describe how samples/organisms/participants were allocated into experimental groups.

Not relevant, because our study does not involve experimental groups.

5. Blinding

Describe whether the investigators were blinded to group allocation during data collection and/or analysis.

Not relevant, because our study does not involve experimental groups.

Note: all studies involving animals and/or human research participants must disclose whether blinding and randomization were used.

6. Statistical parameters

For all figures and tables that use statistical methods, confirm that the following items are present in relevant figure legends (or in the Methods section if additional space is needed).

n/a	Confirmed
☐	☒ The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement (animals, litters, cultures, etc.)
☐	☒ A description of how samples were collected, noting whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☐	☒ A statement indicating how many times each experiment was replicated
☐	☒ The statistical test(s) used and whether they are one- or two-sided (note: only common tests should be described solely by name; more complex techniques should be described in the Methods section)
☐	☒ A description of any assumptions or corrections, such as an adjustment for multiple comparisons
☐	☒ The test results (e.g. *P* values) given as exact values whenever possible and with confidence intervals noted
☐	☒ A clear description of statistics including central tendency (e.g. median, mean) and variation (e.g. standard deviation, interquartile range)
☐	☒ Clearly defined error bars

See the web collection on statistics for biologists for further resources and guidance.

Software

Policy information about availability of computer code

7. Software

Describe the software used to analyze the data in this study.	To analyze the data we used publicly available UCINET software and igraph package in R programming language. As stated in the manuscript, our custom code is available upon request.

For manuscripts utilizing custom algorithms or software that are central to the paper but not yet described in the published literature, software must be made available to editors and reviewers upon request. We strongly encourage code deposition in a community repository (e.g. GitHub). *Nature Methods* guidance for providing algorithms and software for publication provides further information on this topic.

Materials and reagents

Policy information about availability of materials

8. Materials availability

Indicate whether there are restrictions on availability of unique materials or if these materials are only available for distribution by a for-profit company.	No unique materials were used.

9. Antibodies

Describe the antibodies used and how they were validated for use in the system under study (i.e. assay and species).	No antibodies were used.

10. Eukaryotic cell lines

a. State the source of each eukaryotic cell line used.	No eukaryotic cell lines were used.
b. Describe the method of cell line authentication used.	No eukaryotic cell lines were used.
c. Report whether the cell lines were tested for mycoplasma contamination.	No eukaryotic cell lines were used.
d. If any of the cell lines used are listed in the database of commonly misidentified cell lines maintained by ICLAC, provide a scientific rationale for their use.	No commonly misidentified cell lines were used.

Animals and human research participants

Policy information about studies involving animals; when reporting animal research, follow the ARRIVE guidelines

11. Description of research animals

Provide details on animals and/or animal-derived materials used in the study.	No animals were used.

Policy information about studies involving human research participants

12. Description of human research participants

Describe the covariate-relevant population characteristics of the human research participants.	The study did not involve human research participants.