EasyChair Preprint

№ 3282

Real-Time Parameter Estimation for Modelling Malware Propagation on Business and Social Networks Within a Corporate Environment

Stephanie Kiss, Xiao-Si Wang and Jessica Welding

July 9, 2020

Real-time parameter estimation for modelling malware propagation on business and social networks within a corporate environment

Stephanie Kiss[1], Xiao-Si Wang[1] and Jessica Welding[1,2]

[1]Applied Research, British Telecommunications plc
[2]Lancaster University, UK
{steph.kiss, selina.wang}@bt.com
jesswelding@hotmail.com

Abstract. Tackling malware that spreads through business and social networks is a big cyber security challenge for large organisations and enterprises. To address this problem, we propose a new real-time parameter estimation method for forecasting Trojan malware propagation in such an environment. We set up a novel framework to estimate the per-interaction transmission rate p and verify the results of the estimation through a combination of real and simulated data sets. We discuss the benefits of integrating interactions into malware propagation models and study the accuracy and performance of our estimator for the parameter p. We examine how this method enables us to incorporate early detection data into real-time forecasts and how we are thus able to model newly detected malware.

Keywords: malware propagation model · forecasting · real-time · zero-day attack · parameter estimation · compartmental model · agent-based model · Trojan malware · networks · spreading agent · stochastic modelling · simulations.

1 Introduction and Motivation

With the debilitating effects some malware have had in recent years on corporations and public bodies [1], it's imperative to prepare ahead and have foresight when fighting against malicious software, i.e. malware. It has been theorised that malware propagation modelling can be used to anticipate the damage malicious software can cause [2]. The theory developed has drawn inspiration from human epidemiology, with the first application of a compartmental model developed for computer malware considered by Kephart and White [3].

A Trojan or Trojan horse is a prevalent type of malware that uses social engineering to trick users into executing malicious code, for example via clicks on links sent through email or instant messages; in other words, the between user interactions become the main vehicle of infection transmission. Utilising the business and social network of each victim user and the user-to-user interactions, different types of Trojan malware or Trojan components of compound

malware can spread into the larger business and social network communities within an entire organisation. Therefore to model Trojan propagation within such an environment, it is important to consider the interactions through email or other types of business or social communications. Hence this work focuses on the spread of Trojan malware with an infection vector of phishing emails and other malicious digital communications via corporate social networks.

A recent example of this type of malware is the Emotet banking Trojan that was first discovered in 2014. This virus has a very simple infection vector, it is initiated via a spearphising email posing as high-urgency communication usually with a subject line like "Your payment details" or posing as a well-known delivery service. The virus is executed using clicks on malicious links in the email or downloaded attachments such as PDFs or Word Documents. Once established, it propagates using spreading modules, which harvest Outlook contacts and gather sensitive information such as passwords [4]. The compromised accounts are then used to send out more phishing emails using the user's contacts lists. For these types of malware, it is therefore imperative to take into account the user's email contacts list and the volume of interactions since this is the main vehicle of transmission.

These types of malware have very serious consequences for businesses around the world. The Emotet Trojan is widely known as a banking Trojan inflicting disruption on financial institutions by stealing highly sensitive information, processing fraudulent transactions and creating disruption to a company's regular operations, however it is not limited to only banks and has targeted governments and other businesses [4]. However, Emotet is not the alone, some other common banking Trojans are Zeus and Dridex, the latter is becoming more prevalent and the number of detected cases has been steadily growing [5]. The damages of the malware can be costly and could take long time to resolve since they can lay undetected for long periods of time. Therefore, it's important for companies to be able to not only detect but once detected, quickly start to resolve these problems and deploy responses to save on lost time and resources. This is why it is important for large organisations particularly large enterprises to be able to model the spread of malware and create end-to-end malware modelling and response tools. Small and Medium sized Enterprises (SME) because of their sizes, are generally less likely to have the capacity and resources to model malware propagation, so the study presented in this paper is less relevant to the SME environment.

As mentioned in the following Section of Prior Art, the modelling of malware propagation has been theoretical for a long time and different parameter combinations have been used to simulate hypothetical "What-if" scenarios. Real-time model models using real world data to estimate the parameters was unseen. In real-time modelling, using real-world data poses new challenges to the modelling problem. To use real-world data, we have to first consider what type of data feeds are suitable for the problem and then find methods to use such data to estimate the parameters. Our problem as defined earlier was to model Trojan propagation with a corporate environment in real time and Trojan largely relies on people's

social and business network to propagate and between user interaction frequency is one of the real-world data which are collectable and therefore we will use this data set as part of the parameter estimation. Our work then introduces a new estimation method to estimate the infection transmission rate per interaction p which can then be used in forecasting systems and simulations. This early-life real-time data would have to contain real-time interaction frequency which was introduced earlier, network structure, and infection incidence information which are routinely collected. Incorporating our estimator into an end-to-end malware modelling framework can enable cyber security analysts within a company to respond in real-time and anticipate how a malware could spread in a network. This can lead to shorter disruptions and lessening the costs of an attack.

To our knowledge, this is the first study to estimate p, the transmission rate per interaction for Trojan propagation real-time forecasting. In Section 3 we present the modelling framework and the subsequent estimator; in Section 4 we describe the results and conclude this paper in Section 5 with suggestions for future work.

2 Prior Art

Malware propagation models have been studied in the literature for some time [2, 6, 7]. Liu et al. [2] implemented an S-I-R compartmental model with the aim of studying the theoretical dynamics of online malware spread and to theorise on the best response approaches. Newman et al. [6] analysed different email networks and the effects of various response strategies on the studied graphs. Komninos et al. [7] developed a worm propagation model that models the spread of malware through people's contact lists. In this work, they created acquaintance graphs by generating edges between nodes in a network, however, they did not take into account the effect of weighted graphs.

Weighted networks were first introduced by Deijfen [8] to study human epidemics on graph-based networks where the transmission does not take place with the same probability between individuals and analysed the effect this had on vaccination and epidemic thresholds. In their following work with Britton [9], they showed a relationship between the volume of connections and the propagation of an epidemic by incorporating the degree distribution of the network graph and estimating the basic reproduction number (R_0), which is one of the key characteristics of an infection. They also suggested that R_0 can be over- or underestimated by overlooking this relationship. Their suggestions highlight the need to utilise non-homogeneous transmission rates and to account for the interplay between interactions and infection spread in malware propagation models.

Further work has been done by Faghani [10] on modelling the propagation of Trojan malware on online social networks where they also validated their results through experiments. This work makes no attempt to estimate the actual propagation characteristics and only states a non-exhaustive list of pre-defined parameters which they then use to calibrate their models. To understand the evolving malware better, we need timely and dynamic parameters which are not

reliant on overly restrictive assumptions. Therefore, we focus on estimating p, the transmission probability per time unit per interaction. This parameter does not require us to assume the states of each individual. Nevertheless, this importance has been noted in the human and animal infectious disease literature and has been studied for some time [11, 12].

3 Methods

3.1 Model assumptions

Nodes in a network A node in our network is defined as an end-point device that a member of a business or social network uses to communicate with other members of the network. This communication can be instant messages, emails, voice calls, etc. We use a node to denote that there is a relationship between a device and human and we are not looking at autonomous devices. An infection in a node occurs because an end-user has executed malware therefore we do not differentiate between nodes and end-users. Because our malware propagation models are temporal models, the scenario where a user owns two or more systems and execute malware on these systems at the same time is rare and is ignored in the study.

Network structure The network structure of the business and social network in a corporate in this study is represented by a graph. The nodes are connected by edges in the graph, where each edge denotes that a direct interaction has taken place between any two nodes at some point in the past. The weight of these edges represents the number of interactions that have occurred between two connected nodes in a given time period, this can be communication via email, instant messages, etc. An edge does not necessarily mean that an interaction has taken place during the time period which we are modelling but it represents a possible channel of interaction. Neighbours are the set of nodes a given node has interacted with directly at some point in the past.

Interactions between nodes in the network An interaction between any two nodes in a network is bi-directional, which means any two nodes can communicate with each other as long as they are connected by an edge. An edge forms once the first interaction has taken place, this is logged by systems like contact lists, email, and instant messages history. In our estimation method, we focus only on incoming interactions from infected nodes to their susceptible neighbours. We look at incoming interactions because we assume that only through new communication can nodes be infected and that their outgoing interactions cannot infect themselves if they are susceptible. We assume that infected individuals cannot be re-infected therefore interactions between any two infected neighbours are not incorporated into the model. We assume removed nodes in the network cannot re-infect the susceptible nodes because when computing systems are cleaned up or taken out of the network it is rare or not possible to infect other

systems again. As we are estimating an average infection transmission rate, we consider each type of interaction to have equal probability of infecting, in which case we only care about the sum of all interactions from infected neighbouring nodes. A simple example is given in Figure 1.

Communication methods In this model we used a simplified approach where we weighted each type of inter-personal digital communication that could occur within a business setting equally. In this paper, we do not analyse the impact of different types of communication methods on the spread of malware, however we note the importance in the case of Trojan malware.

Time period The modelling framework allows for user-specified time periods, we set a daily granularity which we will refer to as our time unit for the rest of this paper. The interaction data and infection incidence data are broken down into daily time periods as well.

Input data The input data to this model comprises multiple sources. The network structure, set of nodes and their neighbours and their interactions are all data sets that can be obtained from corporate network and system logs.
The estimator has been calibrated on simulated interaction data sets which are simulated according to assumptions made on the network and the data. We set up an experiment where we used different distributions to generate the interactions between neighbours for each time unit.
We also use an open-source data set that represents a Facebook social network to substitute for a real-life corporate interaction network [13]. This network contains 4069 nodes representing members of a social network and their corresponding edges representing Facebook friendship. We sample from the interactions for each existing edge for each day in an infectious period which we set to 30 days in this experiment.

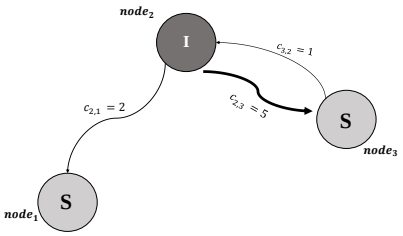


Fig. 1: An illustration of an example set of nodes, where $node_2$ is infected and $node_1$ and $node_3$ are susceptible. Their interactions are represented by $c_{i,j}$ where i, j represent the starting and end node respectively. The direction of the arrow indicates the direction of interaction.

3.2 Malware propagation model

The model for malware dynamics is an S-I-R compartmental model that describes the transitions of individuals in a given corporate social network between these three states:

Susceptible (S) healthy but can be infected
Infected (I) contracted the malware and can spread it
Removed (R) no-longer infected, cannot infect others nor can it be reinfected

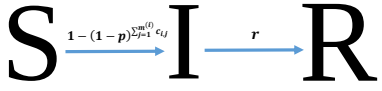


Fig. 2: An illustration of the malware propagation model between each state.

We use an S-I-R model as a simplistic transmission model because it encapsulates the key stages of infection transmission and it is sufficient for demonstrating the estimation method of our key parameter p, the average rate of transmission for a single interaction for every node in our network.The malware spreading is based on infection probabilities and therefore historical data on who has been infected and their interactions can be used as the main variables to calculate the infection probabilities. Too many characteristics on the users are out of our scope and may be used in further studies. Nodes move through the states in the S-I-R sequence and we use p to model the average transmission rate per interaction and use r as the average removal rate. The transition between states is modelled using the relationship in Figure 2. Our removed state denotes the set of nodes that have either been infected then removed from the network or have been patched and cannot be infected again. We only focus on the S to I transitions as we seek to find p given real-time data. We incorporate interactions into this model to parallel a weighted network where transmission rates may not be the same for all nodes. The likelihood of transmission is higher for nodes that have more interactions with infected nodes. We derive the transmission rate for any susceptible node using this interaction-based model.

At time unit t, we take any $node_i$ where $i \in \{1, ..., n_t\}$ and n_t = total number of infected and susceptible nodes in the network at time unit t.
We also denote any infected neighbour of $node_i$ as $node_j$, where $j \in \{1, ..., {m_t}^{(i)}\}$ and ${m_t}^{(i)}$ = total number of infected neighbours of $node_i$ at time unit t.
The corresponding set of neighbourhood interactions are $Inter_{i,j} = \{c_{i,1}, ..., c_{i,{m_t}^{(i)}}\}$ where $c_{i,j}$ represents the total sum of interactions between nodes i and j at time unit t.

P_i represents the infection transmission probability for $node_i$ at time unit t.

$$\begin{aligned} P_i &= P(\mathbf{node_i}\ is\ infected\ at\ time\ unit\ t) \\ &= P(\mathbf{node_i}\ is\ infected\ by\ \mathbf{Inter_{i,j}}\ \ interactions\ at\ time\ unit\ t) \\ &= 1 - P(\mathbf{node_i}\ is\ not\ infected\ by\ \mathbf{Inter_{i,j}} \\ &\quad interactions\ at\ time\ unit\ t) \\ &= 1 - \prod_{j=1}^{m_t{}^{(i)}} (1-p)^{c_{i,j}} \\ &= 1 - (1-p)^{\sum_{j=1}^{m_t{}^{(i)}} c_{i,j}} \end{aligned} \tag{1}$$

3.3 Maximum likelihood estimation

Maximum likelihood estimation is a well-understood statistical approach to parameter estimation and for large sample sizes can be used as an unbiased estimator of a distribution parameter. We obtain the likelihood function using P_i from equation (1) where c_{inf} is the set of interactions and x is the incidence vector.

$$L(c_{inf}; x \mid p) = \prod_{i}^{n_t} P_i{}^{x_i} \, (1-P_i)^{(1-x_i)} \tag{2}$$

where

$$x_i = \begin{cases} 1 & \text{if } node_i \text{ is infected} \\ 0 & \text{if } node_i \text{ is susceptible} \end{cases}$$

This likelihood relates to the outcome of n_t Bernoulli trials, and through this, we can find the the parameter p that maximises the likelihood given the observed data sets x and c_{inf}. We use a bounded scalar minimisation approach on the negative log-likelihood function to find the parameter p for each time unit. We use a scalar minimisation approach because we are dealing with a scalar function of one variable with bound between 0 and 1 since it is a probability. Additionally, this was a method which was fast to implement and is often used for minimising scalar functions [14].

4 Results

We set up the experiment with the assumptions and input data sets described above. We first simulate real interactions using *Poisson* distributions with varied parameter λ. All of these distributions mimic the non-uniform interaction patterns each node has in the network. Business and social interactions within

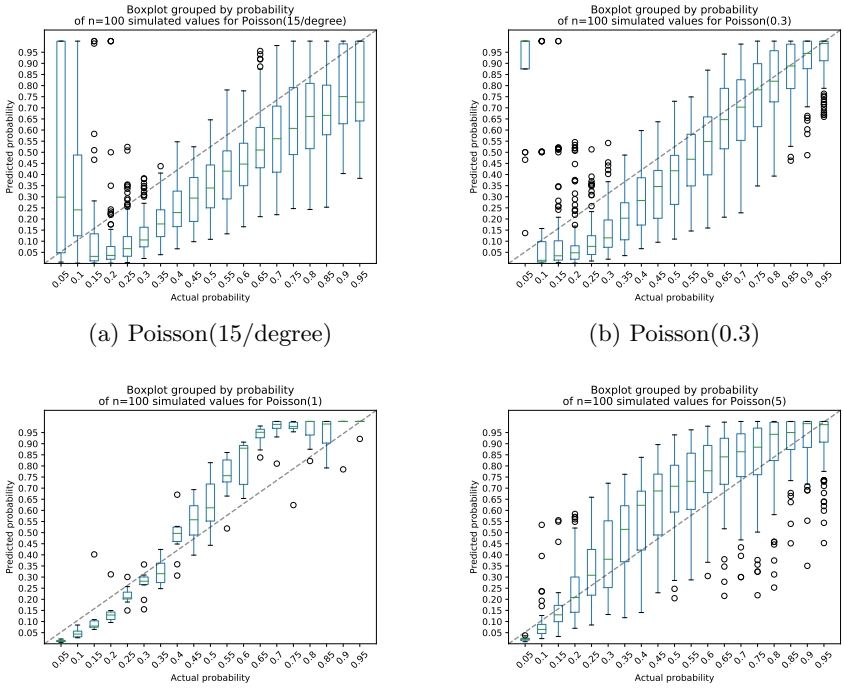
a corporate environment are likely to result in non-uniform distributions of interaction frequencies per edge, for example someone in a sales role may interact with more people than someone in a research role. We sampled from $Poisson(\lambda)$ distributions where $\lambda = 0.3$, 1, 5, $15/degree$.

In the first three formulations of λ, we increase the values of λ gradually to investigate how sensitive our estimation method is to the simulated data with varying underlying values of λ. These three values are chosen to represent value ranges of below 1, 1, more than 1. The $Poisson(1)$ distribution models the scenario when we have an average frequency of interactions as one but a small set of nodes with much higher frequency of interactions. The $Poisson(5)$ distribution parallels an interaction network where the average frequency is higher than in the $Poisson(1)$ case and there is also a much larger variance in the frequency of interactions.

In particular, the $Poisson(0.3)$ distribution is chosen to make comparisons with the fourth choice in which λ equals $15/degree$. When λ is a fixed value, the average frequency of interactions per unit time does not account for the scenario where the frequency of interactions is associated with the number of edges connected to each node. In the fourth and final formulation, λ is inversely proportional to the degree of each node and varies for each node. We use this distribution to model a setting where nodes with large neighbourhoods will have low volume interactions on each edge, e.g. workers who have many individual co-workers in their direct business and social networks are receiving fewer emails per co-worker compared to workers who have fewer co-workers in the direct network but are in frequent contact with them. Since the average degree in our network is approximately 44, $\lambda = 15/44 \approx 0.3$ which is in comparison to the average frequency outlined in the $Poisson(0.3)$ distribution.

We define our time unit as one day. The infection transmission rate per interaction per day for each simulation is set as $p \in \{0.05, 0.02, ..., 0.95\}$, a set of values which we iterate through. Each simulation produces a set of infected and susceptible nodes for each day. Together with the simulated interaction data set and the network structure data set, we are then able to test the efficacy of the estimator. We take the estimated p for each day of the infectious period and average them to produce the average transmission rates. To compare efficacy over different simulation runs of the estimator, we also set up $r = 100$ runs with each distribution described. Each of these runs represents a 30-day infectious period for each p we set. We represent the spread of the estimates using box plots and show the perfect estimation with the diagonal dotted line.

The estimation has varying accuracy for the simulated values of p and also for the different interaction frequency distributions we sample the contacts from as shown in Figure 3. We observe that for the relatively low frequency of interactions, such as the case of $Poisson(0.3)$, the method tends to underestimate the per-contact transmission rates, particularly for lower per-contact transmission rates. In the case of $Poisson(1)$, the estimator seems to produce results with the least variation but overestimates for larger transmission rates. When interaction frequency follows $Poisson(5)$, the estimator seems to estimate well for very



(a) Poisson(15/degree)

(b) Poisson(0.3)

(c) Poisson(1)

(d) Poisson(5)

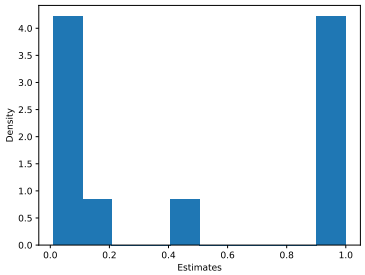
Fig. 3: Box plot of r=100 simulations using (a) Poisson(15/degree); (b) Poisson(0.3); (c) Poisson(1); (d) Poisson(5)

small transmission rates but overestimates for the rest of the values. Finally for interaction pattern modelled by $Poisson(15/degree)$, the variation in the estimation results is much bigger than that for $Poisson(0.3)$ for very small values. Apart from the very small values, the estimator for $Poisson(15/degree)$ tends to underestimate which is a similar behaviour to the estimator for $Poisson(0.3)$.

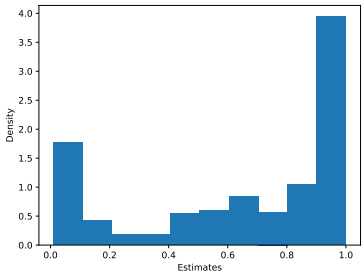
5 Discussion and Future Work

The explanation for the general variation that we see in all of the results can be partially attributed to the structure of the underlying interaction network. Since we have certain isolated nodes, an infection may or may not take off. Therefore, the number of cases may be very small and however many interactions occur, the cluster may not end up interacting with other clusters and the spread of the infection is halted. This can contribute to the results that we see for the $Poisson(0.3)$ case where p is small. In Figure 4c, we can see that most estimated values of p are 1, this is a direct result of a constant likelihood function. This may be due to the fact that some nodes have no interactions or that some nodes that may act as a bridge between communities are removed early on.

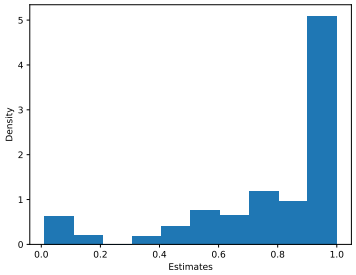
The likelihood function in this case is therefore a constant if c_i is equal to 0, then P_i equals to 0. We see the same pattern in the $Poisson(15/degree)$ case in Figure 4a, which also relates to the network structure for small values of p where if certain well-connected neighbourhoods are infected then we can estimate accurately but this is not the case for isolated clusters.



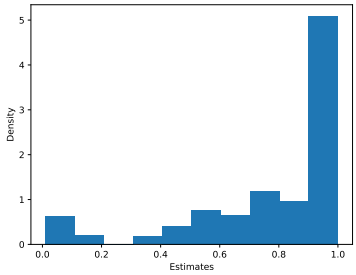
(a) p=0.05 and Poisson(15/degree)



(b) p=0.75 and Poisson(15/degree)



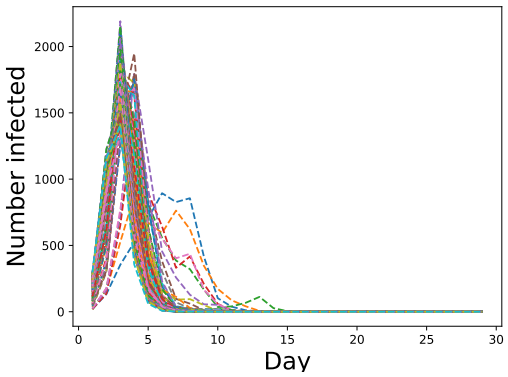
(c) p=0.05 and Poisson(0.3)



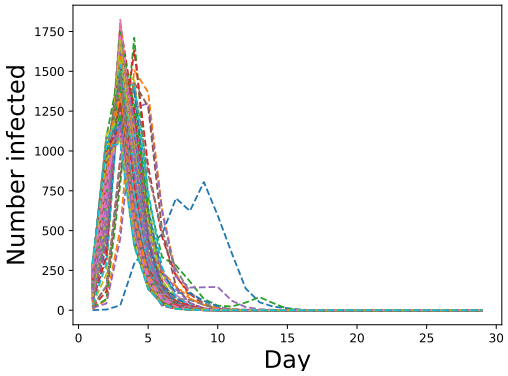
(d) p=0.75 and Poisson(0.3)

Fig. 4: Density histogram of r=100 simulations for (a) p=0.05 and Poisson(15/degree); (b) p=0.75 and Poisson(15/degree); (c) p=0.05 and Poisson(0.3); (d) p=0.75 and Poisson(0.3)

There is also a noticeable bias in the estimates for different distributions we sampled from. For the $Poisson(15/degree)$ and $Poisson(0.3)$ cases, we can see a trend of underestimating p. We see the opposite for the $Poisson(5)$ distribution. Where we underestimate values of p, we see that the density of values are generally around the correct p values however we also observed that a considerable amount of our estimations of p are significantly lower than expected. The resulting biases for the different interaction distributions are likely related and can be seen in Figure 4b and 4d. From the incidence curve in Figure 5a and 5b, we see no obvious difference between the different simulations and therefore attribute the bias to possibly very low volumes of interactions that occur which may then wrongly indicate low p values when minimising the likelihood function. This in



(a) p=0.75 and Poisson(15/degree)



(b) p=0.75 and Poisson(0.3)

Fig. 5: Number of infections per day for (a) $p = 0.75$ and Poisson(15/degree); (b) $p = 0.75$ and Poisson(0.3)

turn helps explain why the $Poisson(5)$ distribution overestimates values of p, since we are more likely to observe much higher levels of interactions which then wrongly overestimate p. Since we are aware of this bias, we may incorporate this into the forecasts when we observe similar interaction patterns. For the slight overestimation we observe for the $Poisson(1)$ case, the explanation is that when we are using such high values of transmission rate per interaction, we may be observing very similar spreading behaviour for a variety of p above that threshold. Therefore, it may be very difficult to distinguish between the scenarios where a node may have a single interaction for a malware with an extremely high transmission rate per interaction or one where many interactions have happened but the malware is not as infectious. In both cases the transmission probability may be very close to 1.

Overall, these results indicate that when we observe a Trojan malware, we are able to estimate its probability of infection per interaction on average fairly accurately. However, we see limitations in accuracy when we observe a malware that has characteristics that indicate it might be extremely infectious and when the interactions distributions have certain features. Although this is an undesirable effect, this result together with the intuitions of the transmission model shows that the way we model malware through interactions may need to be reconsidered. It is important that for forecasting the spread of infection, this inaccurate estimation is accounted for and some form of bias correction measures are applied. We also see that the variance within estimates for different runs of the same simulated data sets can potentially be problematic. This is something that needs to be considered when developing the real-time forecasting methods we have discussed. We have experimented with applying MCMC methods and will need to further explore this option which may be more flexible and provide more accurate results that incorporate this variance.

The method we propose enables us to estimate p and can provide more accurate and precise forecasts and simulations. The results of this investigation have shown that the properties of different malware lend to different estimation accuracy and this relationship has to be further researched and analysed. A natural progression of this work is to compare the spreading behaviour of highly infectious malware and assess how the bias in our estimates affects forecasts. We also aim to verify our results on real interaction and incidence data. In addition, we are working on models for non-homogeneous transmission rates and we are using the findings and methodologies of this work to develop real-time malware propagation forecasting models. We will thus be able to identify at-risk individuals and help cybersecurity analysts respond to threats in an informed and timely manner. This will aid the deployment of optimal malware control strategies by being more specific and detailed.

We also aim to analyse the effect of user behaviour on the spread of malware which could enhance this model and provide more details on effective and more targeted response strategies.

6 Declaration

The views expressed in this paper are solely those of the authors and do not necessarily represent the views of their employers.

References

1. Ghafur, S., Kristensen, S., Honeyford, K. et al. A retrospective impact analysis of the WannaCry cyberattack on the NHS. npj Digit. Med. 2, 98 (2019).
2. Liu, W., Zhong, S. (2017). Web malware spread modelling and optimal control strategies. Scientific reports, 7, 42308.
3. J. Kephart, S. White, Directed-graph epidemiological models of computer viruses, in: Proceedings of IEEE Computer Society Symposium on Research in Security and Privacy, 1991, pp. 343–359.

4. Cybersecurity and Infrastructure Security Agency (CISA), Department of Homeland Security. Emotet Malware. https://www.us-cert.gov/ncas/alerts/TA18-201A 2018
5. L. Remorin and M. Marcos, Online Banking Threats in 2015: The Curious Case of DRIDEX's Prevalence, https://blog.trendmicro.com/trendlabs-security-intelligence/curious-case-dridexs-prevalence/ 2016
6. M. E. J. Newman, Stephanie Forrest, and Justin Balthrop. Email networks and the spread of computer viruses. Physical Review E, 66(3), 2002.
7. Komninos T, Stamatiou YC, Vavitsas G. A worm propagation model based on people's email acquaintance profiles. International Workshop on Internet and Network Economics 2006 Dec 15 (pp. 343-352). Springer, Berlin, Heidelberg.
8. Deijfen, Maria. (2011). Epidemics and vaccination on weighted graphs. Mathematical bio-sciences. 232. 57-65.
9. Britton, Tom Deijfen, Maria Liljeros, Fredrik. (2011). A Weighted Configuration Model and Inhomogeneous Epidemics. Journal of Statistical Physics - J STATIST PHYS. 145.
10. Faghani MR, Nguyen UT. Modeling the propagation of Trojan malware in online social networks. arXiv preprint arXiv:1708.00969. 2017 Aug 3.
11. Jin F, Jansson J, Law M, et al. Per-contact probability of HIV transmission in homosexual men in Sydney in the era of HAART. Aids (London, England). 2010 Mar;24(6):907-913.
12. Ssematimba, A., Elbers, A. R., Hagenaars, T. J., de Jong, M. C. (2012). Estimating the per-contact probability of infection by highly pathogenic avian influenza (H7N7) virus during the 2003 epidemic in The Netherlands. PloS one, 7(7), e40929.
13. Leskovec J, Mcauley JJ. Learning to discover social circles in ego networks. In Advances in neural information processing systems 2012, pp. 539-547.
14. Brent, R. P.,Chapter 4: An Algorithm with Guaranteed Convergence for Finding a Zero of a Function, Algorithms for Minimization without Derivatives,1973, Englewood Cliffs, NJ: Prentice-Hall, ISBN 0-13-022335-2