Performance Analysis of Cell-Phone Worm Spreading in Cellular Networks through Opportunistic Communications

Yahui WU[1], Su DENG[1], Hongbin HUANG[1], Yiqi DENG[2]
[1]*Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha, 410073,China*
[2]*Department of Computer Science, University College London, London, WC1E 6BT,UK*
wuyahui@nudt.edu.cn

***Abstract*—Worms spreading directly between cell-phones over short-range radio (Bluetooth, WiFi, etc.) are increasing rapidly. Communication by these technologies is opportunistic and has very close relation with the social characteristics of the phone carriers. In this paper, we try to evaluate the impact of different characteristics on the spreading performance of worms. On the other hand, the behaviors of worms may have certain impact, too. For example, worms may make phones be completely dysfunctional and these phones can be seen as *killed*. We study the impact of the killing speed. Using the Markov model, we propose some theoretical models to evaluate the spreading performance in different cases. Simulation results show the accuracy of our models. Numerical results show that if users do not believe the data coming from others easily, the worms may bring less damage. Surprisingly, if the users are more willing to install the anti-virus software, the worms may bring bigger damage when the software becomes to be outdated with high probability. Though the worms can bring big damage on the network temporarily by killing phones rapidly, numerical results show that this behavior may decrease the total damage in the long time. Therefore, killing nodes more rapidly may be not optimal for worms.**

***Index Terms*—opportunistic communications, delay tolerant network, 3G networks, Markov process, cell-phone worms.**

I. INTRODUCTION

Smart phones are becoming more popularity at present, such as iPhone, Blackberry, and Android devices. Besides traditional cellular networks, smart phones can also communicate with each other in the peer-to-peer way through short-range radios. As described in the work [1], the operating systems in these smart phones are open-API and this openness would allow richer applications to run over these phones. On the other hand, this openness also brings a new chance for hackers to write malicious software which can control or damage the phones. Therefore, smart phones become a burgeoning target for malicious activates. Many newly reported families of worms, including Commwarrior, Cabir and Lasco can spread through the peer-to-peer way, and these new worms can easily persist in the network and remain undetected because of the decentralized infection and the dynamic topology, so they can bring much bigger loss [2]. To defense these worms efficiently, we should first understand the spreading performance of them in different cases, and this is the object of this paper.

Communications in the peer-to-peer way use the short-range radio. Because the transmission range is limited and we can not control the mobility of the phone carriers, the topology of the network formed by the short-range radio may be changing all the time. The end to end path between two users may not exist when they want to communicate with each other, so this policy can only provide intermittent and opportunistic network connectivity to users. Therefore, this communication form can be called opportunistic communication and these users with smart phones form a Delay Tolerant Network (DTN) [3]. DTN is a very hot topic recently and it is proposed to support many emerging networking applications, where end-to-end connectivity can not be assumed, examples include deep-space exploration [4], communication in urban areas [5], vehicular network [6], etc. Any two nodes can communicate with each other only when they come into the transmission range of each other in DTN. In order to overcome the network partitions, nodes of DTN communicate through a "store-carry-forward" mode [7-8]. Due to the node mobility, different links come up and down. If the sequence of connectivity graphs over a time interval is overlapped, then an end-to-end path might exist, so the message should be forwarded over the existing link, stored and carried at the next hop until the next link comes up, and so on and so forth.

Two users can communicate when they come into the transmission range of each other and if only one of them was infected by worms, the other one may be infected in this contact. However, many worms can infect phones only when the users download the files which contain malicious codes. Therefore, if users do not believe the data coming from others (because of the selfish nature) and do not download the data, these users can not be infected [1]. To defense these worms, users may install the anti-virus software, but some users may be not willing to install the software because they must pay some cost for the software [9]. In addition, worms may evolve and this will make the software be outdated. That is, the *immunize* nodes will become *susceptible* again. In this paper, we evaluate the impact of above behaviors on the worms spreading process by the Markov model and give the theoretical model.

The worms may make the phones be completely dysfunctional sometimes and these phones can be seen as *killed* [10]. Though the worms may bring bigger damage by killing nodes, this may also lose the chance of infecting more nodes.

Digital Object Identifier 10.4316/AECE.2012.02001

At present, a number of works have demonstrated the severe threat of malware spreading through opportunistic communications. Su *et al*. proposed a preliminary investigation of worm infectious in a Bluetooth environment and demonstrated that an effective way for malware to propagate is via Bluetooth by some simulations based on the Bluetooth scanner traces [11]. Bose *et al*. showed that a worm that uses both SMS/MMS and Bluetooth can propagate faster than that by messaging alone [12]. However, these papers did not consider the impact of the users' social characteristics. Cheng *et al*. proposed a malware propagation model in generalized social networks, and they propose a novel differential equation-based model to analyze the mixed behaviors of delocalized infection and ripple-based propagation for the hybrid malware [13]. This paper also failed to consider the social characteristics of users and did not consider the impact of the behaviors of worms. The spreading speed of worms was studied using traditional epidemiological modeling tools and high-fidelity realistic human mobility data [14]. Specially, this paper mainly takes into account the effects of exposure times, wireless propagation radii, and limited population susceptibility. Obviously, it is different from our works. Tang *et al.* explored the malware spreading problem in dynamic and temporal graphs [15]. Some works considered the worm detection methods. Yan *et al*. [16] proposed Blue-Watchdog which can detect the Bluetooth worm propagation in public areas and Cheng *et al*. presented SmartSiren, a collaborative virus detection and alert system for smart phones [17]. To defense the worm, Zhu *et al*. [1] proposed a social network based patching scheme in cellular networks. Li *et al*. [2] proposed CPMC which is an efficient proximity malware coping scheme in smart phone-based mobile networks. Khouzani *et al*. [18] proposed to quarantine the malware infection by regulating the communication range of the nodes, and then they proposed an epidemic model to represent the propagation of malware in a battery-constraint mobile wireless network in which the worm can dynamically control the rate at which it kills the infected node and also the transmission range and/or the media scanning rate [19]. Because the distribution of patches consumes bandwidth which is scarce in wireless networks, a trade-off method between security risks and resource consumption is proposed [20], and then Khouzani *et al*. proposed a dynamic game method to defense malware attack which is spreading in the peer-to-peer way [21]. An optimal distributed malware defense system for mobile networks with heterogeneous devices is proposed in paper [22]. A survey of mobile malware in the wild can be found in [23].

To our best knowledge, none of the papers explored the impact of different behaviors of both users and worms on the worm spreading performance. The main contributions of this paper can be summarized as follows:

- Using the continuous time Markov model, we give the accurate theoretic model of malware spreading through opportunistic communication in different cases, such as the selfish nature, killing probability, etc.;
- We check the accuracy of our model through simulation. Numerical results show that the behaviors of both users and worms can have big impact on the spreading performance.

The rest of the paper is organized as follows: in next section, we briefly describe the network model, and in section III we describe the malware spreading model by the Markov model in different cases. Then we give the simulation and numerical results. At last, we summarize the work of this paper.

II. NETWORK MODEL

Suppose the network has a set of users and every one of them has a smart phone. By abuse of language, the symbol users, phones and nodes denote the same thing in the next sections of this paper. The set of all the users is denoted by **V**, and we have $|\mathbf{V}|=N$. That is, there are totally N nodes. The opportunistic link exists between two users only when they come into the transmission range of each other, which means a communication contact, so the mobility rule of the users is critical. In this paper, we assume that the occurrence of contacts between two nodes follows a Poisson distribution, which is found in many well-known mobility models, such as random waypoint and random direction [24]. This assumption is validated by works on studying the mobility behaviors of both human and vehicles [25]. So the exponential inter-contact time holds for mobility behaviors of both the human and vehicles, and we can assume that the inter-contact time between two users follows an exponential distribution with the parameter denoted by λ. As shown in many papers [20-21], nodes in the network can be divided into four classes. Nodes which are not contaminated by the worm, but are prone to infection can be called *susceptible* nodes. A node is *infective* if it is contaminated by the worm. The *infective* nodes can be killed, i.e., render it completely dysfunctional and these nodes can be seen as *killed* (also seen as *dead*). On the other hand, users may install the anti-virus software to defense the worm attack and these nodes are referred to as *recovered*. A *susceptible* node may be infected when it meets the *infective* nodes. However, because the selfish nature of users, they may not believe the data coming from these *infective* users. Recently, many papers explored the selfish nature, and many of them denote the selfish level between two nodes by a probability value [1-2]. In this paper, we also use the probability policy and one node downloads the data coming from other users with probability p. Though anti-virus software can make nodes be immune, but because it is not free, users may not install the software. In this paper, we assume that *susceptible* nodes install the software with a probability which follows the exponential distribution and the parameter is denoted as υ. In addition, the *infective* users also install the anti-virus software according to the exponential distribution and the parameter is denoted as ε.

III. SPRADING PROCESS

A. Basic Spreading Process

In this section, we assume that worms never kill nodes. Therefore, no *dead* nodes exist in the network. Let $S(t)$ denote the number of *susceptible* nodes at time t, and $I(t)$ denote the number of *infective* users. Obviously, the number of *recovered* users $R(t)$ at time t equals to $N- S(t)- I(t)$. Therefore, the state of the network can be denoted by ($S(t)$, $I(t)$). From state ($S(t)$, $I(t)$), there are three events may make the state of the network change. The first one is that a *susceptible* node encounters with an *infective* user and the

former believes the latter one. The second event is that a *susceptible* user installs the anti-virus software, and the third one means that an *infective* user installs the anti-virus software. Above three events will make the network changes into one of the following three states through one-step transition, respectively, that is: (*S(t)*-1, *I(t)*+1), (*S(t)*-1, *I(t)*), (*S(t)*, *I(t)*-1). The transition process of the network's state through one-step transition is shown in figure 1.

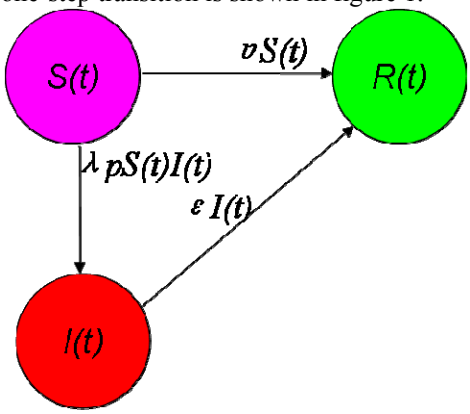

Figure 1. One-step transition graph with basic model

Because the inter-contact time between two users follows an exponential distribution with parameter λ, there are totally *S(t)I(t)* independent and identically distributed (I.I.D) Possion process may make one *susceptible* node become *infective* at time *t*. Therefore, as shown in figure 1, we can easily get the following formula,

$$(S(t),I(t)) \to (S(t)-1,I(t)+1), with\ rate \lambda S(t)I(t)p \quad (1)$$

On the other hand, because the *susceptible* nodes install the software follows an exponential distribution with parameter υ, we can get that,

$$(S(t),I(t)) \to (S(t)-1,I(t)), with\ rate \upsilon S(t) \quad (2)$$

Similarly, we can get another expression,

$$(S(t),I(t)) \to (S(t)-1,I(t)), with\ rate\ \varepsilon I(t) \quad (3)$$

Let **Q** denote the generate matrix and it represents the transition rate from one state to another. According to above three expressions, we can obtain every element in **Q** which is shown as follows,

$$\begin{cases} Q(S(t)-1,I(t)+1 \mid S(t),I(t)) = \lambda S(t)I(t)p \\ Q(S(t)-1,I(t) \mid S(t),I(t)) = \upsilon S(t) \\ Q(S(t),I(t)-1 \mid S(t),I(t)) = \varepsilon I(t) \\ Q(others \mid S(t),I(t)) = 0 \end{cases} \quad (4)$$

Symbol *others* denotes any state other than (*S(t)*-1, *I(t)*+1), (*S(t)*-1, *I(t)*) and (*S(t)*, *I(t)*-1).

B. Extend Model with Dead nodes

In this section, we will explore the spreading performance when the worms kill *infective* nodes according to an exponential distribution with parameter μ. In this case, nodes will be divided into four classes, so only two elements cannot describe the state of the network any more. For this reason, we use (*S(t)*, *I(t)*, *R(t)*) to denote the state of the network at time *t*. Symbol *R(t)* denotes the number of *recovered* nodes at time *t*. Obviously, there are *D(t)=N-S(t)-I(t)-R(t)* nodes were killed by the worms. From state (*S(t)*, *I(t)*, *R(t)*), there are four events may make the state of the network change. The first three events are the same as those in above section, and the last one means that one *infective* node was killed by the worms and this will make the network transmit into state (*S(t)*, *I(t)*-1, *R(t)*). We can get the one-step transition process easily which is shown in figure 2.

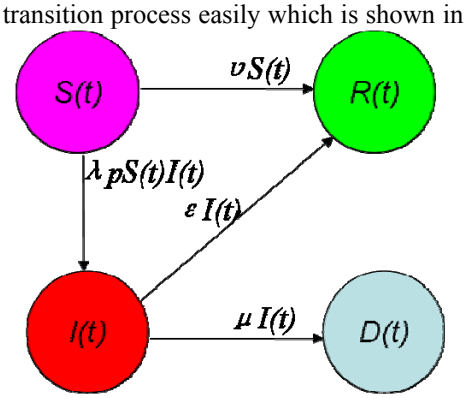

Figure 2. One-step transition graph with *dead* nodes

So we can get the generate matrix **Q** as follows,

$$\begin{cases} Q(S(t)-1,I(t)+1,R(t) \mid S(t),I(t),R(t)) = \lambda S(t)I(t)p \\ Q(S(t)-1,I(t),R(t)+1 \mid S(t),I(t),R(t)) = \upsilon S(t) \\ Q(S(t),I(t)-1,R(t)+1 \mid S(t),I(t),R(t)) = \varepsilon I(t) \\ Q(S(t),I(t)-1,R(t) \mid S(t),I(t),R(t)) = \mu I(t) \\ Q(others \mid S(t),I(t),R(t)) = 0 \end{cases} \quad (5)$$

C. Extend Model with Outdated Anti-virus Software

In many applications, worms may dynamically change their parameters in response to the dynamics of the network, in order to maximize their overall damage [19]. For this reason, the anti-virus software may be outdated, and the *recovered* nodes may become *susceptible* again. To our best knowledge, none of the works considered the impact of this phenomenon on the worm spreading performance theoretically before. For simplicity, we assumed that the anti-virus software in the *recovered* nodes becomes outdated according to the exponential distribution with parameter ρ.

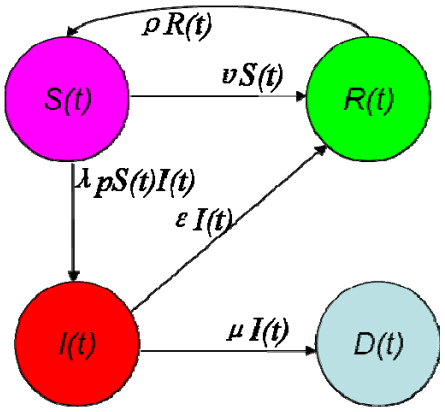

Figure 3. One-step transition graph with outdated anti-virus software

From state (*S(t)*, *I(t)*, *R(t)*), there are five events may make the state of the network change and the first four events are the same as those in above section. The last one denotes that one *recovered* node becomes *susceptible* again, and this event will make the network change into state (*S(t)*+1, *I(t)*, *R(t)*-1). The one-step transition model is shown in figure 3. Therefore, the generate matrix **Q** can be shown as follows,

$$\begin{cases} Q(S(t)-1, I(t)+1, R(t) \mid S(t), I(t), R(t)) = \lambda S(t) I(t) p \\ Q(S(t)-1, I(t), R(t)+1 \mid S(t), I(t), R(t)) = \upsilon S(t) \\ Q(S(t)+1, I(t), R(t)-1 \mid S(t), I(t), R(t)) = \rho R(t) \\ Q(S(t), I(t)-1, R(t)+1 \mid S(t), I(t), R(t)) = \varepsilon I(t) \\ Q(S(t), I(t)-1, R(t) \mid S(t), I(t), R(t)) = \mu I(t) \\ Q(others \mid S(t), I(t), R(t)) = 0 \end{cases} \quad (6)$$

D. Performance Analysis

First, we define the one-step transition probability matrix **M** according to the generator matrix **Q**. Matrix **M** denotes the probability of the transition from one state to another through one-step transition. From analysis above, we know that every row of **Q** represents the transition rate from one state to any other state. Therefore, the sum of all elements in one row denotes the rate of leaving the current state. For example, given state *SS*, the rate of leaving this state denoted by *speed(SS)* can be shown as follows,

$$speed(SS) = \sum_{i \in Sspace} Q(i \mid SS) \quad (7)$$

Symbol *Sspace* represents the set of all valid states. Now, we can get the probability of the transition from state *SS* to another state *i* through one-step transition, that is,

$$P(i \mid SS) = Q(i \mid SS) / speed(SS), i \in Sspace \quad (8)$$

So according to formula (8), we can get every element of matrix **M**.

Let symbol *Dst* denote the set of the absorption states. In the basic model (first model), worms cannot kill node and the *recovered* node cannot become *susceptible* again, so when every node become *recovered*, the network cannot change its state, so the absorption state sets is {(0, 0)}. When worms can kill nodes (second model), *Dst* contains two states: {(0, 0, *N*)}, {(0, 0, 0)}. When the *recovered* node can become *susceptible* again (third model), state {(0, 0, *N*)} is not the absorption state any more, so we have *Dst*={(0, 0, 0)}. Note that as shown in figure 3 of the third model, for state (*S*, *I*, *R*), if *S*+*R*=*N*, the network does not contain worms any more. However, nodes may still change their state between *recovered* and *susceptible*, and this is not necessary. Therefore, we can assume that if *S*+*R*=*N*, state (*S*, *I*, *R*) is also absorption state.

Now, we begin to explore the damage when the network goes into *Dst*. To denote the damage induced by the worms, we use some utility values. Once the *susceptible* node installs anti-virus software to defense the worms, the user must pay some cost denoted by *U1*, and the cost can be seen as the damage that the worms induced. On the other hand, if the user does not adopt some policy and the worms kill the *infective* node finally, the damage will be denoted by *U2*. In addition, the *infective* node may also install anti-virus software and the cost can be denoted by *U3* (including the cost for the software and the damage induced by worms, so may be bigger than *U1*). In this paper, we assume that $U2 \geq U3 \geq U1$. In addition, we also assume that $I(0) > 0$, that is, at the initial state *S0*, at least one node was *infective* node. Note that if other event also brings cost, we can tackle these cases easily as above three events.

Let *DU(k)* denote the expected total damage till the network first reaching to one state in *Dst*, starting from state *k*. Obviously, for any state *j* in *Dst*, we have *DU(j)*=0. Therefore, by conditioning on the one-step transition out of the current state, we have,

$$DU(k) = \sum_{j \in Sspace} P(j \mid k)(DU(j) + SU(k, j)) \quad (9)$$

Symbol *SU(k,j)* denotes the damage when the network changing from state *k* to *j*. According to above analysis, we can get the value of *SU(k,j)* easily. For example, in the extend model with outdated anti-virus software, from state *k*=(*Sk*, *Ik*, *Rk*), the network may go into any one of five states. If the network goes into state *j*=(*Sk*-1, *Ik*, *Rk*+1), one *susceptible* node becomes *recovered* and the cost is *U3*, so we have *SU(k,j)* =*U3*. Note that if *Ik*=0, worms cannot spread again, so there is no need to install the anti-virus software and the worms cannot bring damage further. Therefore, for state *k*=(*Sk*, *Ik*, *Rk*), if *Ik*=0, we have *DU(k)*=0, and for any other state *j*, we have *SU(k,j)*=0.

Define **DU** as a column vector of the expected damage starting from any valid state, and **SU** a matrix which denotes the damage by one transition from one state to another. We can obtain the following formula easily,

$$\begin{aligned} &\mathbf{DU} = \mathbf{P} * \mathbf{DU} + (\mathbf{P} * \mathbf{SU}')^{Vector} \\ &\Rightarrow \mathbf{DU} = (\mathbf{I} - \mathbf{P})^{-1} (\mathbf{P} * \mathbf{SU}')^{Vector} \end{aligned} \quad (10)$$

Symbol $\mathbf{SU}'$ denotes the transposed matrix of **SU** and $(\mathbf{P}*\mathbf{SU}')^{Vector}$ denotes a column vector composed by the diagonal elements in matrix $\mathbf{P}*\mathbf{SU}'$. For example, $(\mathbf{P}*\mathbf{SU}')^{Vector}(i) = (\mathbf{P}*\mathbf{SU}')(i,i)$. Symbol **I** denotes the identity matrix. According to formula 10, we can get the total damage *DU(S0)* from the initial state *S0* easily.

IV. SIMULATION AND NUMERICAL RESULTS

A. Simulation Result

In this section, we will evaluate the accuracy of our continuous Markov model, and we run several simulations using the Opportunistic Network Environment (ONE) simulator [26]. Our simulation is based on the Random Waypoint (RWP) mobility model, which is commonly used in many mobile wireless networks. Related to the simulation settings, there are totally $|V|$=20 mobile nodes moving according to a speed chosen from a uniform distribution from 4m/s to 10m/s within a 100m×100m terrain. The transmission range of the nodes is set to 2m. One node believe the data coming from others with probability p=0.5. Other settings are: ε=0.002, μ=0.001, υ=0.001, ρ=0.001. The utility value of *U1*, *U2* and *U3* may be different in different applications, and the value of them may be varying. However, for simplicity, we assume that their value is fixed, and we set *U1*=2, *U2*=20 and *U3*=10. More simulations with other datasets and settings will be our future work.

Let the number of *infective* nodes *I(0)* at the initial state increase from 1 to 10 (*S(0)*=*N*- *I(0)*). The simulations run 20 times, and we can get the simulation result as shown in figure 4. Comparing the simulation and theoretical results, we can see that the total damage of the theoretical model is very close to that obtained by simulation. For example, figure 4 shows that the average deviation between the simulation and theoretical result is about 2.9%. This demonstrates the accuracy of our continuous time Markov model. For this reason, in the next subsection, we will use the theoretical results obtained by our model in the

performance analysis in different cases.

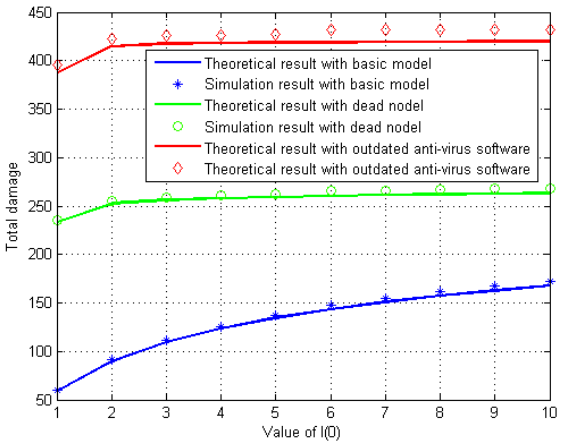

Figure 4. Simulation result with RWP mobility model

B. Performance Analysis with Numerical Results

In fact, figure 4 also shows that if the worms kill nodes with some probability, they may bring bigger damage. However, if the worms kill nodes with higher speed, whether can they bring bigger damage? As described above, the worms kill nodes according to the exponential distribution with parameter μ. If the value of μ is bigger, the worms can kill *infective* nodes with bigger probability, and we can say that the worms kill nodes more rapidly (and μ can be called killing speed). To explore above problem, we will study the numerical result with different value of μ. First, we set the number of *infective* nodes *I(0)* at the initial equals to 5, and let μ increase from 0.002 to 0.02. We give the numerical result when *U2*=20 and 10, respectively. Other settings are the same as that in above section.

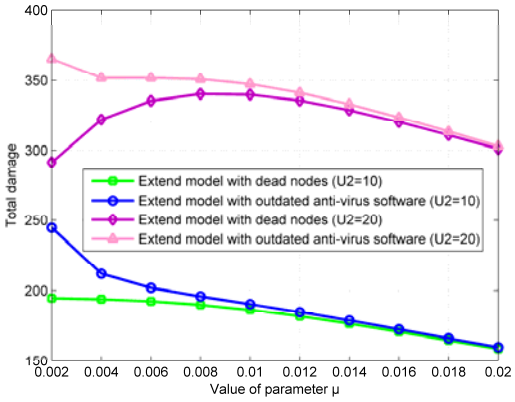

Figure 5. Total damage when worms kill nodes with different speed

From the result shown in figure 5, we can see that in the extend model with dead nodes (*U2*=20), the total damage is increasing with the increasing of μ when μ is smaller than 0.01, but it is decreasing with the increasing of μ when μ is bigger than 0.01. However, if the anti-virus software may be outdated, the total damage is decreasing with the increasing of μ all the time in above settings. Therefore, the optimal killing speed for worms may be different in different applications, and sometimes it is not rational for them to kill the *infective* nodes rapidly. This result shows that it is an interesting work to explore the optimal behaviors of worms.

The *infective* nodes may also become *recovered* instead of only *dead*, and the transition is according to an exponential distribution with parameter ε. Therefore, we need to explore the impact of this parameter. It is easy to see that if ρ=0, the anti-virus software never becomes outdated, and this is corresponding to the extend model with dead nodes. Therefore, the second model is a special case of the third model and we only give the numerical result for the third model. We let μ=0.001, *U2*=20, and ε increase from 0.002 to 0.02. Other settings are the same as that in figure 5. We give the numerical result in figure 6 when ρ=0, 0.001 and 0.01, respectively. From the result, we can see that the damage will decrease if the *infective* nodes are more willing to install anti-virus software when ρ=0 and 0.001. However, when ρ=0.01 and ε<0.006, the total damage increases with the increasing of ε, this is because that the *recovered* nodes may become *infective* again with bigger probability.

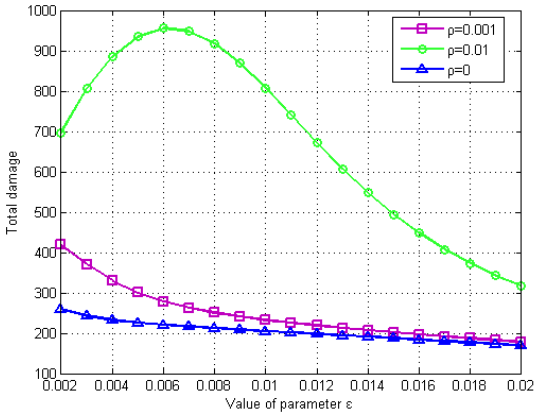

Figure 6. Total damage with different value of ε

In the next subsection, we will explore the total damage when the anti-virus software becomes outdated with different speed (different value of ρ). We set *I(0)*=5, *U2*=20, μ=0.001, and let the value of ρ increase from 0 to 0.01. Other settings are the same as that in above section. In this subsection, we also only give the result for the third Markov model. Figure 7 shows the results when *U3*=5 and 10, respectively. So if the anti-virus software becomes outdated with higher speed, the worms will bring more damage.

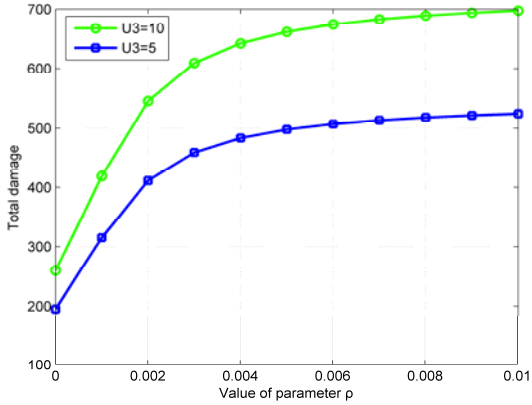

Figure 7. Total damage with different value of ρ

We know that one user may not believe the data coming from other users, and he downloads the data with probability *p*. Now, we begin to study its impact. For simplicity, we use the same settings as that in the simulation, but we fix *I(0)*=5 and let the value of *p* increase from 0 to 1.The result is shown in figure 8. The result shows that if nodes believe others more easily, the worms will bring bigger damage in the network. Therefore, though the selfish behavior may

bring adverse effect for some applications, it can prevent the worms spreading at some degree.

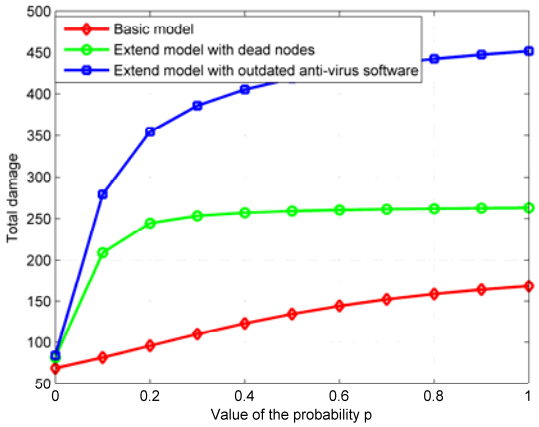

Figure 8. Total damage with different value of probability p

V. Conclusion

This paper explores the impact of different behaviors of users and worms on the damage bringing by the worms in cellular networks. Specially, for the selfish nature, users download the data coming from others with certain probability. In addition, users may be not willing to install the anti-virus software because they have to pay certain cost and the software may be outdated again. For worms, they may kill *infective* nodes with certain probability. We study the impact of all above behaviors on the worms spreading performance. The worms spread in a peer-to-peer way through the short-rang radio, and the communication is opportunistic. We describe the worms spreading process by the Markov model in different cases. Simulation result shows the accuracy of our model. Numerical results show that it is not rational for worms to kill nodes rapidly in some applications. If the *infective* nodes are more willing to install anti-virus software, the total damage may decrease. However, if the software becomes outdated rapidly, the total damage may increase. In addition, if users do not believe others, the worms may bring less damage.

In our work, we assume that the worms can transfer immediately every time the *infective* nodes forward data to the *susceptible* nodes. However, this is not true in some applications, the worms may be very big, and the contact duration may be very short. Therefore, the worms may be not transferred successfully in one contact. In the future, we will explore the impact of the worms' size. In addition, we want to extend our results and conclusions to more general case by using more general model and simulation settings.

References

[1] Z. Zhu, G. Cao, S. Zhu, S. Ranjan, A. Nucci, "A social network based patching scheme for worm containment in cellular networks," in *Proc. IEEE INFOCOM*, 2009. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.142.9664

[2] F. Li, Y. Yang, and J. Wu, "CPMC: an efficient proximity malware coping scheme in Smartphone-based mobile networks," in *Proc. IEEE INFOCOM*, 2010.

[3] K. Fall, "A delay-tolerant network architecture for challenged internets," in *Proc. ACM SIGCOMM*, 2003. [Online]. Available: http://dl.acm.org/citation.cfm?id=863960

[4] G. Papastergious, I. Psaras, and V. Tsaoussidis, "Deep-space transport protocol: a novel transport scheme for space DTNs," *Computer Communications*, vol.32, no. 16, 2009.

[5] J. Ott, E. Hyytiä, P. Lassila, T. Vaegs, and J. Kangasharju, "Floating content: information sharing in urban areas," in *Proc. IEEE Int. Conf. PerCom, Mar. 2011*. [Online]. Available: http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=5767578

[6] B. Oh, Y. Na, J. Yang, S. Park, J. Nang, and J. Kim, "Genetic algorithm-based dynamic vehicle route search using car-to-car communications," *Advances in Electrical and Computer Engineering*, vol. 10, no. 4, pp. 81-86, 2010. [Online]. Available: http://www.aece.ro/abstractplus.php?year=2010&number=4&article=13

[7] T. Spyropoulos, T. Turletti, and K. Obrazcka, "Routing in delay tolerant networks comprising heterogeneous populations of nodes," *IEEE Trans. Mobile Computing*, vol.8, no. 8, Aug. 2009.

[8] E. Bulut, Z. Wang, and B. Szymanski, "Cost effective multi-period spraying for routing in delay tolerant networks," *IEEE/ACM Trans. Networking*, Vol. 8, no. 5, Oct. 2010. [Online]. Available: http://dl.acm.org/citation.cfm?id=1959404

[9] J. Omic, A. Orda, and P. V. Mieghem, "Protecting against network infections: a game theoretic perspective," in *Proc. IEEE INFOCOM*, 2009. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5062065

[10] MHR. Khouzani and S. Sarkar, "Dynamic malware attack in energy-constrained mobile wireless networks." in *Proc. Fifth Symposium on Information Theory and Applications*, 2010. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5454093

[11] J. Su, K. Chan, A. Miklas, K. Po, A. Akhavan, S. Saroiu, E. de Lara, and A. Goel, "A preliminary investigation of worm infections in a Bluetooth environment," in *Proc. ACM WORM*, 2006. [Online]. Available: http://dl.acm.org/citation.cfm?id=1179545

[12] A. Bose and K. Shin, " On mobile viruses exploiting messaging and Bluetooth services," in *Proc. ICST Securecomm*, 2006. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp

[13] S. Cheng, W. C. Ao, P. Chen, and K. Chen, "On modeling malware propagation in generalized social networks," *IEEE Comm. Lett.*, vol. 15, no. 1, Jan. 2011.

[14] N. Husted and S. Myers, "Why mobile-to-mobile wireless malware won't cause a storm," in *Proc. USENIX Workshop on Large-scale Exploits and Emergent Threats,* 2011. [Online]. Available: http://dl.acm.org/citation.cfm?id=1972451

[15] J. Tang, C. Mascolo, M. Musolesi, and V. Latora, "Exploiting temporal complex network metrics in mobile malware containment," in *Proc. WOWMOM*, Jue. 2011. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5986463

[16] G. Yan, L. Cuellar, and S. Eidenbenz, "Blue-watchdog: detecting Bluetooth worm propagation in public areas," in *Proc. DSN*, 2009.

[17] J. Cheng, S. H. Y. Wong, H. Yang, and S. Lu, "SmartSiren: virus detection and alert for smart phones," in *Proc. Mobisys*, Jue. 2007. [Online].Available: http://dl.acm.org/citation.cfm?id=1247690

[18] MHR. Khouzani, E. Altman, and S. Sarkar, "Optimal quarantining of wireless malware through power control," in *Proc. Fourth Symposium on Information Theory and Applications*, University of California, San Diego, Feb. 2009. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5044961

[19] MHR. Khouzani, E. Altman, and S. Sarkar, "Maximum damage malware attack in mobile wireless networks," in *Proc. IEEE INFOCOM*, 2010.

[20] MHR. Khouzani, S. Sarkar, and E. Altman, "Dispatch then stop: optimal dissemination of security patches in mobile wireless networks," in *Proc. IEEE CDC*, 2010. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5717273

[21] MHR. Khouzani, S. Sarkar, and E. Altman, "A dynamic game solution to malware attack," in *Proc. IEEE INFOCOM*, Shanghai, China, Apr. 2011. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5935025

[22] Y. Li, P. Hui, D. Jin, L. Su, and L. Zeng, "An optimal distributed malware defense system for mobile networks with heterogamous devices," in *Proc. IEEE SECON*, Jun. 2011.

[23] A. P. Felt, M. Finifter, E. Chin, S. Hanna, and D. Wagner, " A survey of mobile malware in the wild," in *Proc. ACM Workshop on Security and Privacy in Mobile Devices (SPSM)*, 2011. [Online]. Available: http://dl.acm.org/citation.cfm?id=2046618

[24] R. Groenevelt, P. Nain, and G. Koole, "The message delay in mobile ad hoc networks," *Performance Evaluation*, 2005.

[25] T. Karagiannis, J. –Y. L. Boudec, and M. Zojnovic, "Power law and exponential decay of inter contact times between mobile devices," in *Proc. ACM MOBICOM*, 2007. [Online]. Available: http://dl.acm.org/citation.cfm?id=1287875

[26] A. Keranen, J. Ott, and T. Karkkainen, "The ONE simulator for DTN protocol evaluation," in *Proc. SIMUTOOLS*, 2009. [Online]. Available: http://dl.acm.org/citation.cfm?id=1537683