Bluetooth Worm Propagation: Mobility Pattern Matters!

Guanhua Yan[†] Leticia Cuellar[†] Stephan Eidenbenz[†]
Hector D. Flores[‡] Nicolas Hengartner[†] Vincent Vu[§]

[†] Discrete Simulation Sciences (CCS-5)[*]
Los Alamos National Laboratory
[‡] Department of Statistics
Rice University
[§] Department of Statistics
University of California, Berkeley

ABSTRACT

The alarm that worms start to spread on increasingly popular mobile devices calls for an in-depth investigation of their propagation dynamics. In this paper, we study how mobility patterns affect Bluetooth worm spreading speeds. We find that the impact of mobility patterns is substantial over a large set of of changing Bluetooth and worm parameters. For instance, a mobility model under which devices move among a fixed set of activity locations can result in worm propagation speeds four times faster than a classical mobility model such as the random walk model. Our investigation reveals that the key factors affecting Bluetooth worm propagation speeds include spatial distributions of nodes, link duration distributions, degrees to which devices are mixed together, and even the burstiness of successive links.

1. INTRODUCTION

While people have been annoyed by the malware prevalent in the Internet for more than one decade, they may soon have to wrestle with this monster on another field: mobile devices such as cell phones and PDAs. Recent occurrences of proof-of-concept mobile worms such as Cabir [9] and CommWarrior [13] can easily trigger our imagination: what if malicious malware steals private information stored on our cell phones? In light of the notorious history of Internet worms, security concerns over mobile malware can hardly be exaggerated.

Common to most existing mobile malware is that they all leverage Bluetooth capabilities to propagate themselves. Bluetooth is a short-range radio technology that is aimed at connecting different wireless devices operating at low power consumption and at low cost. It has a wide range of applications, such as wireless headsets, dial-up networking, and peer-to-peer file sharing. The market for Bluetooth devices has been growing tremendously in recent years: world-wide, 272 million such devices have been shipped in 2005, twice as many as in 2004 [16]. The popularity of Bluetooth devices increases our concern over wide-spread malware propagation on mobile devices.

Compared to the substantial efforts on investigating the characteristics of Internet malware and defense strategies against them, research on the malware propagation on Bluetooth devices is still in its infancy. The key difference is that the Bluetooth worm uses a proximity-based infection mechanism, which is to say, a Bluetooth-enabled device controlled by the worm can only infect neighbors within its radio range. Hence, the propagation means used by the Bluetooth worm differs from those of the Internet worms that scan the whole IP address space for susceptible victims. In addition, in contrast to Bluetooth worms which propagate on dynamic mobile networks, those topologically aware Internet worms (e.g., email worms) usually spread in relatively stable network topologies.

The distinctions between the spreading means of Bluetooth worms and existing Internet worms motivate us to study the characteristics of worm propagation on mobile Bluetooth devices. The aim of this paper is to understand the impact of device mobility patterns on Bluetooth worm propagation. Our finding is that device mobility patterns impose prominent impact on the Bluetooth worm propagation speed: a mobility model under which devices move among a fixed set of activity locations can result in worm propagation speeds four times faster than a classical mobility model such as the random walk model. The effects of mobility models remain dominant even if we vary Bluetooth worm parameters. Our investigation reveals that the key factors affecting Bluetooth worm propagation speeds include spatial distributions of nodes, link duration distributions, degrees to which devices are mixed together, and even the burstiness of successive links.

In our previous work presented in [17], we have performed a preliminary investigation on the nature and characteristics of Bluetooth worms. This paper significantly departs from our previous one in two main ways. First, one of the main focuses in the previous paper is to study how device speed, device density, and fractions of insusceptible devices affect Bluetooth worm propagation, and throughout that paper, only a single mobility model is considered. This paper, instead, investigates how mobility patterns impact Bluetooth worm propagation and thus involves multiple mobility models. On the other hand, to explain why mobility pattern matters, this paper provides an in-depth statistical analysis of the mobility traces generated from different mobility models and also some mathematical analysis; these are missing from our previous work.

The paper is organized as follows. Section 2 provides background knowledge for this paper. In Section 3, we investigate

[*] Los Alamos National Laboratory Publication No. LA-UR-06-4672

ASIACCS'07, March 20-22, 2007, Singapore.
Copyright 2007 ACM 1-59593-574-6/07/0003 ...$5.00.

the impact of mobility models on Bluetooth worm propagation speeds and then provide explanations for it. In Section 4, we explore the impact of worm model parameters on worm propagation speeds under different mobility models. Section 5 introduces related work and Section 6 concludes this paper.

2. BACKGROUND

2.1 Bluetooth Worm Modeling

We model an infection cycle of a Bluetooth worm as the flow chart in Figure 1. When a Bluetooth worm is activated, it starts searching for Bluetooth-enabled devices in its vicinity. The worm broadcasts Bluetooth inquiry packets and waits for responses. Because of the uncertainty about how many responses will be received, the worm has a parameter N_{inq} for how many neighbors it expects to discover and another parameter T_{inq} for the maximum amount of time it wants to wait. If N_{inq} responses are received before T_{inq} time units elapse, the worm stops the inquiry phase on the arrival of the N_{inq}-th response; otherwise, regardless of the number of responses it receives, the worm terminates the inquiry phase immediately after T_{inq} time units elapse.

After the worm collects a list of Bluetooth-enabled devices in its radio range, it iterates through the list, attempting the following steps with each neighbor device: establish a connection to it (*Step 1*), probe infection possibility (*Step 2*), replicate the worm code onto the victim device (*Step 3*), and disconnect from it (*Step 4*). Owing to link instability in mobile networks, each of these steps may fail without notice from the other end. Hence, a timer is scheduled in each phase, thus allowing the worm to detect possible connection failures. The maximum amount of times the worm is willing to wait in Steps 1, 2, 3, and 4 are denoted by T_{conn}, T_{probe}, T_{rep}, and T_{disc} respectively.

In Step 1, establishing a connection to a nearby device involves the *paging* process in Bluetooth. We refer the reader to [7] for the details in this process. In Step 2, whether a device can be infected hinges on the vulnerability the worm exploits. For example, the Commwarrior worm probes each victim device for the availability of the 'Obex Push' service; on a positive reply, the worm replicates itself onto the victim. In our model, we simplify this process by distinguishing 3 types of replies from a probed device: A REJECTED reply indicates that the probed device is *insusceptible*, an UNINFECTED reply indicates that the probed device is *susceptible and uninfected*, and an INFECTED reply indicates that the probed device is *susceptible but infected*. The last type of replies may not reflect the behavior of some Bluetooth worms, but this is possible once the worm controls a victim device. In Step 3, the time needed to replicate the worm code onto the victim depends on both the Bluetooth packet type and the size of the worm code. In our model, we use Bluetooth DH5 packet type for transmission because it, for the worm's good, provides the maximum data throughput. The size of the worm code varies from worm to worm. For example, the Cabir.H worm consists of about 7,000 bytes, but the Commwarrior.a!sys worm has 30,582 bytes. In our model, we use S_{worm} to denote the size of the worm code.

Once all the devices on the neighbor list have been iterated, the worm remains inactive for T_{idle} time units. After this amount of time elapses, the worm enters another infection cycle and the process repeats.

Based on our preliminary study in [17], we configured the model parameters as shown in Figure 1. We implemented the worm model as described in the ns-2 network simulator [1] extended with the UCBT Bluetooth simulation module [2]. The UCBT module has a full implementation of the Bluetooth protocol stack, but it does not provide a detailed radio propagation model. Hence, we have implemented the radio propagation model used in [11][20] to calculate signal attenuation. In this model, the path loss, which refers to signal attenuation on the propagation path, is defined as follows:

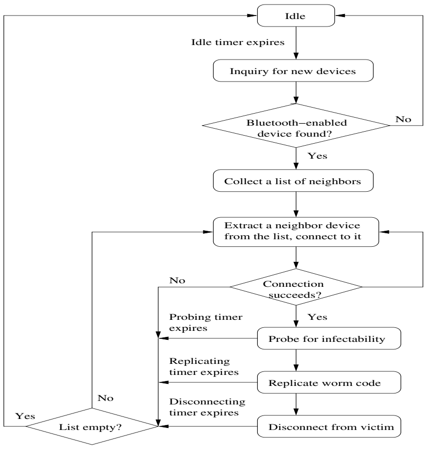


Figure 1: Infection cycle of a Bluetooth worm

Notation	Meaning	Value	
T_{inq}	inquiry timeout	10.24	(sec)
N_{inq}	#inquiry responses	3	
T_{conn}	conn. establishing timeout	5.12	(sec)
T_{probe}	probing timeout	0.1	(sec)
T_{rep}	replication timeout	10	(sec)
T_{disc}	disconnection timeout	0.1	(sec)
T_{idle}	inactive duration	20	(sec)
S_{worm}	worm size	20,000	(bytes)

Table 1: Notations and values in the baseline worm model

$$P_{loss} = \begin{cases} 20 \times log_{10}(4\pi d/0.1224) & d \le 8m \\ 58.3 + 33 \times log_{10}(d/8) & d > 8m \end{cases} \quad (1)$$

This model assumes free space propagation for the first 8 meters (i.e., it assumes that there always exists an unobstructed line-of-sight path between the source and the destination); beyond this range, the signal attenuates at a rate that is a function of $d^{3.3}$, where d is the distance to the source.

Since co-channel interference can significantly impact the performance of Bluetooth networks [11][15], we take it into consideration in our study. Co-channel interference occurs when devices on different channels use the same frequency to transmit packets at the same time. The Bluetooth specification requires that in order for the receiver to successfully decode a signal, the carrier to interference ratio (C/I) must be at least 11 dB. In our study, we assume that a lower C/I ratio directly leads to packet loss.

2.2 Mobility Models

A plethora of models have been proposed that attempt to characterize the mobility of real-world entities. We refer the reader to [8][19] for comprehensive surveys. In the following sections, we give a brief introduction to those that will be used in our experiments. These four mobility models have been widely used in mobile network simulation. Realistic mobility patterns are so complicated that they can hardly be characterized in a single mobility model. We choose these models because they are sufficient to reveal some important factors affecting Bluetooth worm propagation, as will be illustrated in later sections. All these models assume that entities move within a square, the side length of which is denoted by W. For simplicity, some notations are used redundantly for different models.

Random waypoint model. This model has been widely used in the ad-hoc networking community, largely because of its simplicity. In this model, an entity randomly chooses a destination in the rectangle. It then moves straightly from its current position towards the new destination at a speed uniformly drawn between v_{min} and v_{max}. Once it arrives at the destination, it sleeps for a period that is uniformly distributed between τ_{min} and τ_{max}; after it wakes up, it chooses another destination and the process repeats.

Random walk model. The random walk model is often used to capture mobility of entities moving in extremely unpredictable manners. In this model, an entity periodically chooses a direction uniformly distributed between $(0, 2\pi)$ and a speed uniformly distributed between (v_{min}, v_{max}); the duration of each period, denoted as w (in seconds), is constant among all entities. It is possible that in a movement period an entity bumps into the boundary of the area considered. Under such circumstance, the entity is reflected from the boundary at the incoming speed.

Random direction model. In this model, when an entity reaches the boundary, it pauses for a period uniformly distributed between τ_{min} and τ_{max}, then chooses a speed uniformly distributed within $[v_{min}, v_{max}]$ and a direction uniformly drawn between 0 and π (relative to the boundary), and move straightly in that way until it reaches the boundary again. The above process repeats throughout the simulation.

Random landmark model. This model is similar to the random waypoint model except that destinations are randomly picked from a predefined set of locations instead of from the whole area. Once a node arrives at the destination, it pauses there for a fixed period of time τ_{wait}; once it wakes up, it chooses another destination from the set of locations and moves straightly towards it at a speed uniformly distributed between v_{min} and v_{max}.

3. IMPACT OF MOBILITY MODELS

In our experiments, we simulate Bluetooth worm propagation in a network with 200 devices. The movements of the devices are restricted in an area with side length 75 or 150 meters. For the random landmark mobility model, we choose 68 locations, whose coordinates are derived from the Rice University campus map after proper downscaling. For the random walk model, devices update their directions and speeds every 5 seconds (i.e., $w = 5$).

As described in Section 2.2, different mobility models have different control parameters. For instance, the random waypoint model provides the flexibility to change the pause time of a node after a trip. The random walk model, however, does not have such control. To make a fair comparison between the worm propagation speeds under different mobility models, we design our experiments in such a way that no device pauses. Therefore, both τ_{min} and τ_{max} under the random waypoint model and the random direction model are 0, and τ_{wait} is also 0 under the random landmark model. On the other hand, it is observed in [18] that under the random waypoint mobility model the average node speed decays towards v_{min} as simulation progresses. Actually we observed the same phenomenon from the random direction model and the random landmark model but *not* from the random walk model. To achieve a fair comparison among the four mobility models, we let the devices move at a constant speed, which is varied between 1 m/s and 3 m/s in our experiments.

In our study, we assume that all devices are vulnerable to the Bluetooth worm attack. We simulate each scenario for 3600 seconds. The first worm is randomly chosen from the 200 devices and it starts infection at simulation time 600 second. The first 600 seconds is used to stabilize mobility patterns (based on the suggestion given in [8]). For each scenario, we perform 20 sample simulation runs with different random number generation seeds.

Figure 2 depicts the average cumulative number of infected devices as a function of simulation time under the four mobility models. We can draw the following conclusions from it:

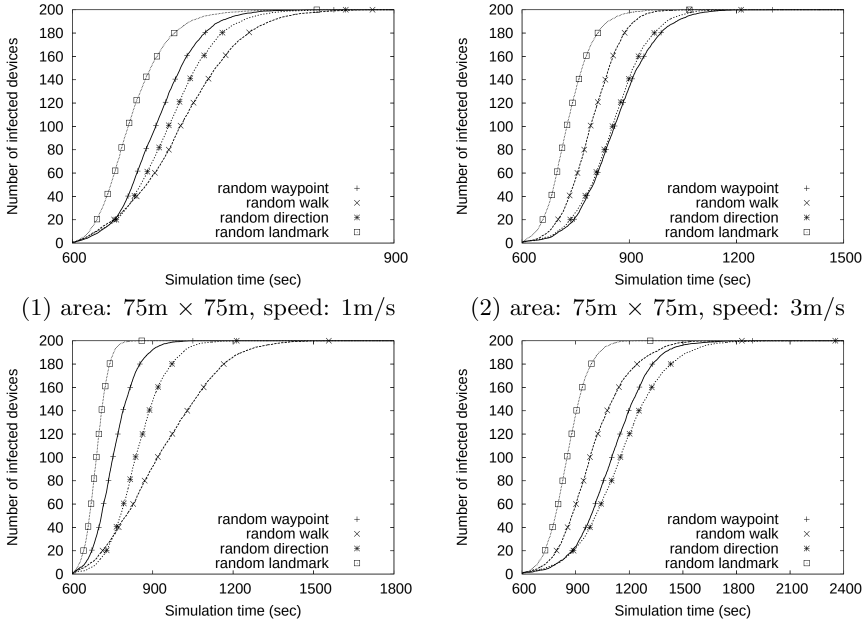
- Mobility models have noticeable impact on the propagation speed of the Bluetooth worm. For instance, if the area is 150m × 150m and the nodal speed is 1m/s, to achieve 50% (80%) infection coverage, the random landmark model requires only 90 (122) seconds, but the random walk model needs 318 (490) seconds, which is about 3.53 (4.02) times the time used under the random landmark model.

- The relative order in the worm propagation speed among the four mobility models is hardly consistent across all the four scenarios. Consider the 80% infection coverage horizon. Only the rank of the random landmark model is preserved irrespective of the nodal speed and the area size: it always needs the least amount of time to infect 80% of the population. The random walk model requires the most amount of time when the speed is 1 m/s but needs the second least amount of time when the speed is 3 m/s, regardless of the area size. The random waypoint model requires less time than the random direction model if the speed is 1 m/s, but the propagation times needed under both models are close to each other if the speed is 3 m/s.

These above observations motivate us to understand the underlying reasons behind them. In the following sections, we are interested in unraveling what statistical properties derived from these mobility models drive the noticeable differences in the Bluetooth worm propagation speeds. Since the orders in the worm propagation speeds among the four mobility models are similar between the two different area sizes, we focus on the 150m × 150m area in the following discussion.

3.1 Spatial Distribution

A Bluetooth worm can only spread itself from an infected device onto those that are within its radio range. Such a proximity-based infection mechanism suggests that the worm propagation speed can be affected by how devices are spatially distributed in the area. To illustrate the spatial distributions under different mobility models, we divide the 150m×150m area into 100 15m×15m grids. We define the *traverse density* of a grid to be the average number of devices that visit that grid per second throughout the simulation (the first 600 seconds is not considered). We present the traverse density distribution under each mobility model in Figure 3. Under the random waypoint model, devices tend to concentrate at the center of the area. This phenomenon has also been observed in many other places [5][14]. Under the random walk model, devices are comparatively evenly distributed in the whole area.



(1) area: 75m × 75m, speed: 1m/s

(2) area: 75m × 75m, speed: 3m/s

(3) area: 150m × 150m, speed: 1m/s

(4) area: 150m × 150m, speed: 3m/s

Figure 2: Infection curves of the baseline worm under different mobility models

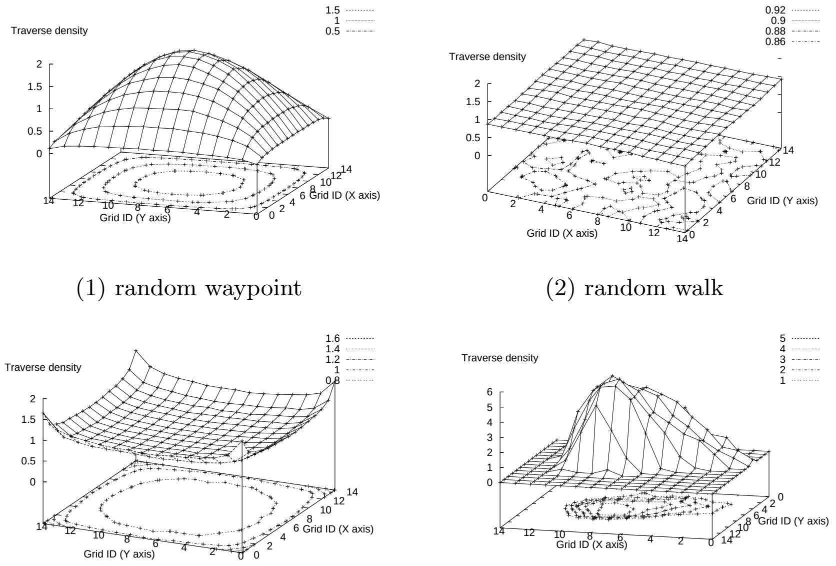
For the random direction model, the distribution of the devices in the area is also very even, except that they appear in the bordering area slightly more often than the other area. Under the random landmark model, the spatial distribution of devices depends on how activity locations are placed in the area; devices only appear at these predefined activity locations and on the paths between them.

The spatial distribution of devices is directly related to the average number of neighbors a device has at any given time. If devices are more closely clustered in the same region, it is more likely that a device can have many neighbors in its radio range. In Figure 4, we depict the node degree histogram under each mobility model. Each curve is derived in the following way. In each sample run, we create a snapshot of the network every 100 seconds after the initial 600 seconds. In a snapshot, if two devices are within each other's radio range, we add an edge between them; hence, each snapshot corresponds to a graph, which is not necessarily connected. For each mobility model, we average the frequencies of node degrees in the networks derived from all the snapshots of the 20 sample runs. Figure 4 shows that there is little difference between the degree histograms under two node speeds. When the node speed is 1 m/s (3 m/s), the average node degrees are 3.80 (3.87), 2.64 (2.68), 2.57 (2.62), and 11.41 (11.66) under the random waypoint model, the random walk model, the random direction model, and the random landmark model respectively. It is pronounced that regardless of the moving speed, devices under the random landmark model tend to have much more neighbors than those under the random waypoint model, which further tend to have more neighbors than those under both the random walk model and the random direction model, but the differences in node degree distribution between the random walk model and the random direction model are only marginal.

To examine how the spatial distribution of the devices affects the behavior of the Bluetooth worm, we present in Figure 5 the average number of responses collected from each inquiry as a function of the number of infected devices in the network. The general trend of each curve is that as the network is populated with more infected devices, fewer devices are discovered in each infection cycle. This is because when an infectious device is performing active scanning, it may not respond to inquiries issued by its neighbors. Here, we are more interested in the relative orders of the curves corresponding to the four different mobility models. We notice that an infectious device under the random landmark model can almost always discover three neighbors in a single infection cycle, irrespective of the nodal speed. Recall that the maximum number of neighbors that can be found in an inquiry attempt (i.e., N_{inq}) is set to be three in our simulation. Furthermore, among the four models, the random waypoint model results that on average an infectious device discovers the second most neighbors in an infection cycle given the same number of devices already infected in the network. The above observations can easily be explained by the degree histograms illustrated in Figure 4: under the random landmark model, a device under the random landmark model tends to have much more neighbors than the other three models, and next to the random landmark model, the random waypoint model leads to the second largest average node degree among the four models.

Interestingly, we find that the device degree distribution can not always explain the difference in the number of neighbors discovered in a single infection cycle. This is manifest when we compare the random walk model and the random direction model. In Figure 5, it is shown that when the node speed is 1 m/s, an infectious device under the random direction model tends to find more neighbors in an infection cycle than under the random walk model, if the worm propagation progress is the same. By contrast, such a difference does not exist when the nodal speed is as high as 3 m/s. Figure 4, however, tells us that the degree distributions are almost the same under these two models, regardless of the nodal speed. We call this observation *Dilemma 1*, which will be explained in Section 3.3.

It is worth mentioning that the spatial distribution of the devices can also affect the duration of the inquiry phase in an infection cycle. If devices are more closely clustered together, an infectious device can easily collect N_{inq} inquiry responses



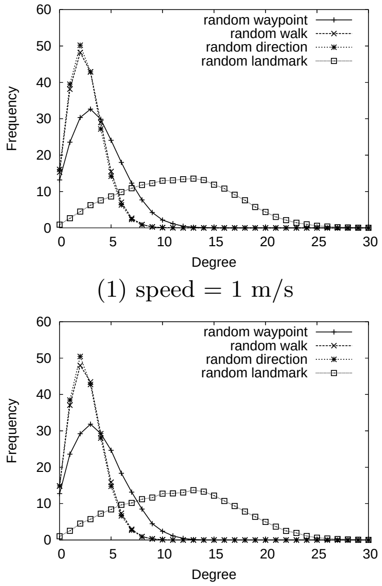
(1) random waypoint

(2) random walk

(3) random direction

(4) random landmark

Figure 3: Traverse densities under the four mobility models



(1) speed = 1 m/s

(2) speed = 3 m/s

Figure 4: Degree histogram

from its neighbors and it thus does not need to wait until the inquiry timer expires. On the other hand, if devices are more sparsely distributed in the area, an infectious device may not be able to find N_{inq} neighbors and thus has to wait for T_{inq} time units before stepping into the next phase. The impact of the mobility models on the durations of inquiry phases is illustrated in Figure 6. It is clear that as the network is populated with more infected devices, the average duration of the inquiry phases increases. This is because an infected device may not receive responses from its neighbors if they are also actively scanning for victims and it thus has to wait until its inquiry timer expires or some of its neighbors finish scanning and respond to its inquiry. Among the four mobility models, the random landmark model leads to the shortest inquiry phase, and next to it is the random waypoint model. If the node speed is 1 m/s, infectious devices under the random walk model tend to collect fewer inquiry responses than those under the random direction model, as shown in Figure 5(1), and in result, the average durations of the inquiry phases under this model are larger. If the node speed is 3 m/s, the differences in the average durations of the inquiry phases under these two models are not pronounced.

3.2 Link-Oriented Metrics

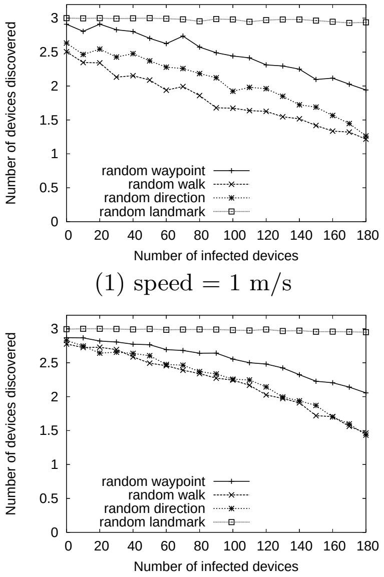
Node degree distribution can not solely explain the difference in worm propagation speeds under the four mobility models; otherwise, we should observe the orders in the worm propagation speeds be consistent between the scenarios with different node speeds. Previous work has already defined several metrics such as link change rate, link duration, and path availability to capture the characteristics of mobility models [4]. Provided that the Bluetooth worm spreads itself onto the devices in its vicinity in a single-hop fashion, we focus on the link-oriented metrics rather than the path-oriented metrics derived from the mobility models. It is, however, worth mentioning that links under discussion here actually are *not* Bluetooth links generated between neighboring devices. Instead, if two devices are physically in each other's radio range, irrespective of whether a Bluetooth connection is established between them, we say that there exists a link between them.

Figure 7 depicts the normalized frequency histogram of link durations from simulation time 600 second to 3600 second. Note that the graph uses logarithmic scale on the vertical axis. The link duration distribution under the random direction model has a heavier tail than those under the other three models, regardless of the node speed. The link duration distribution curves of the random waypoint model and the random landmark model almost overlap with each other and both of them have a heavier tail than the random walk model. The normalized frequency curve of the random walk model is close to a straight line, suggesting that the empirical distribution of the link duration under this model should decrease exponentially.

We further investigate how normalized link duration distribution affects the dynamics of the Bluetooth worm. For ease of explanation, we introduce a few notations here. Let random variable L_{link} be the duration of a link and L_{inq} be the duration of the inquiry phase. We also use D_k, for $1 \leq k \leq N_{inq}$ to denote the time that the active infected device spends on processing the k-th neighbor. If the number of neighbors collected by the worm is smaller than k, we simply let D_k be 0. We further assume that the time needed to successfully establish a connection to a neighbor is τ_{conn}. Furthermore, the inquiry phase and the link may not start at exactly the same time. We thus let δ be the difference between the starting times of the inquiry phase and the link; if it is negative, it means that the link appears after the inquiry phase starts. Then the probability that the infected device can establish a connection to the k-th neighbor, denoted by $P_{conn}(k)$, is

$$\begin{aligned} &P_{conn}(k) = \mathbb{P}(L_{link} \geq \Theta(k)), \\ &\quad \text{where } \Theta(k) = \delta + L_{inq} + \textstyle\sum_{i=1}^{k-1} D_i + \tau_{conn}. \end{aligned} \qquad (2)$$

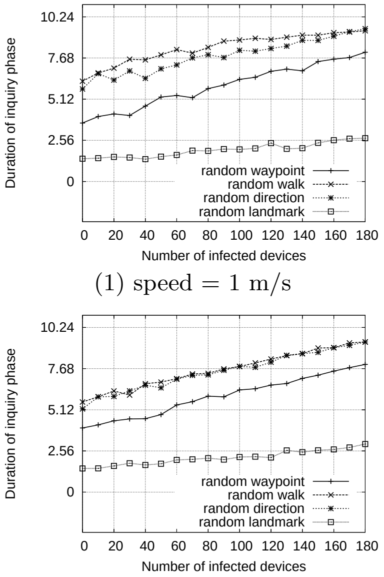
According to Equation (2), if k is 1, the cutoff $\Theta(1)$ is $\delta + L_{inq} + \tau_{conn}$. In order for the neighbor to be discovered, the inquiry phase and the link should overlap for at least τ_{inq}, where τ_{inq} is the time needed to discover a neighbor. Hence, δ is



(1) speed = 1 m/s

(2) speed = 3 m/s

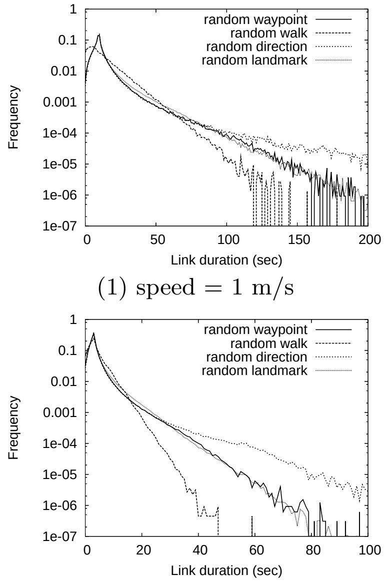
Figure 5: Number of neighbors discovered per infection cycle



(1) speed = 1 m/s

(2) speed = 3 m/s

Figure 6: The average durations of the inquiry phases



(1) speed = 1 m/s

(2) speed = 3 m/s

Figure 7: Normalized frequency histogram of link durations

upper-bounded by $L_{inq} - \tau_{inq}$. Since L_{inq} is at most 10.24 seconds (i.e., T_{inq}) and τ_{conn} is at most 5.12 seconds (i.e., T_{conn}), the cutoff $\Theta(1)$ is at most 25.6 seconds. Figure 7 shows that the durations of the majority of the links are also within this range, regardless of the mobility model and the node speed. Therefore, the difference in L_{inq} can have a significant impact on the connection probability when k is 1. As k increases in Equation (2), the cutoff threshold $\Theta(k)$ also increases, because of the addition of the term $\sum_{i=1}^{k-1} D_i$. Typically, D_i can be about 5 seconds. Hence, as we increase k in Equation (2), the tail fatness of the link duration distribution becomes more important in deciding $P_{conn}(k)$ and the difference in L_{inq} becomes less important.

Actually, these observations can be verified from the simulation results. Figure 8 presents $F_{conn}(k)$, which denotes the fraction of successful connection attempts for the k-th position on the neighbor list, under different combinations of k ($1 \leq k \leq 3$) and mobility models. From Figure 8, we can make the following observations:

- **Observation 1.** $F_{conn}(1)$ under the random landmark model is much higher than that under the random direction model, and $F_{conn}(1)$ under the random waypoint is slightly higher than that under the random direction model for most of the points.
- **Observation 2.** As k increases, the difference in $F_{conn}(k)$ between the random landmark model and the random direction model becomes smaller and the curve corresponding to the random waypoint model gradually moves towards that of the random direction model and then moves farther away underneath it.
- **Observation 3.** $F_{conn}(k)$ under the random walk model is almost always higher than that under the random direction model.

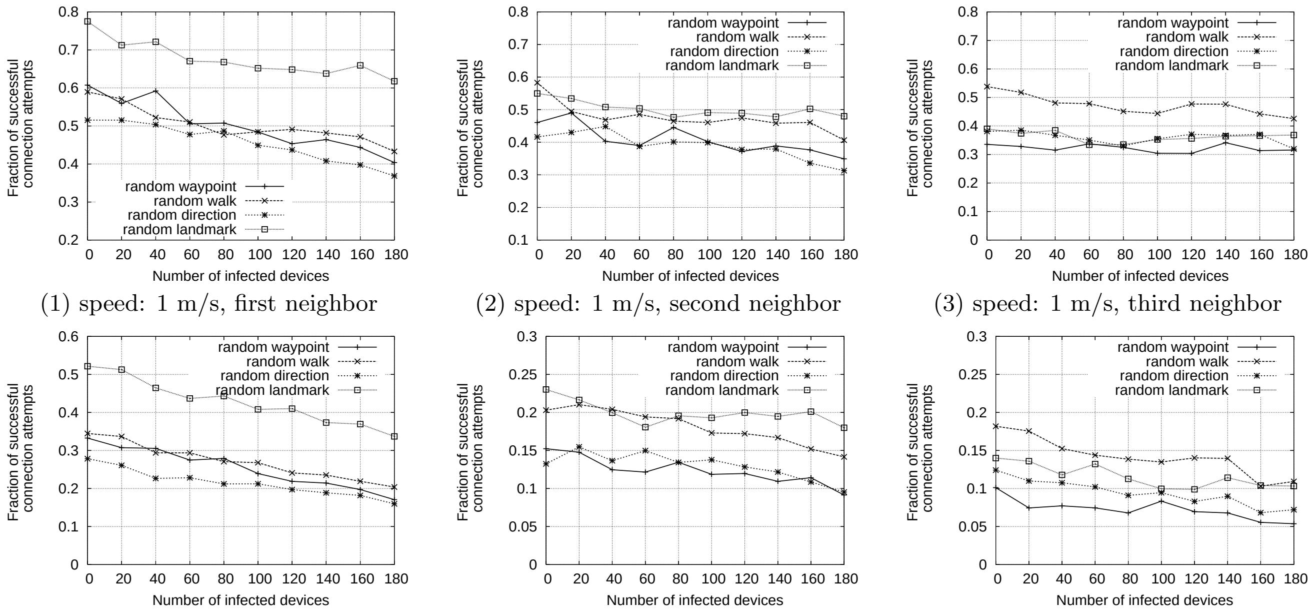
Observation 1 results from the relative order of L_{inq} under the random landmark model, the random waypoint model, and the random direction model, as illustrated in Figure 6. But as k increases, because the normalized link duration distribution curve under the random direction model has a heavier tail than those under both the random waypoint model and the random landmark model (as seen from Figure 7), we have Observation 2.

Observation 3, however, seems to contradict with the explanation we gave. First, L_{inq} under the random walk model is not smaller than that under the random direction model. Actually, as illustrated in Figure 6, when the node speed is 1 m/s, the random walk model leads to a longer inquiry phase. Second, Figure 7 shows that the normalized link duration distribution curves under the random walk model have a thinner tail than that under the random direction model. The above explanation should lead to the opposite of Observation 3. We call such a contradiction *Dilemma 2*, which will be explained in the next section.

From Figure 8, we also notice that regardless of the position on the neighbor list, as more devices are infected in the network, $F_{conn}(k)$ decreases. The reasons can be twofold. First, as the network is populated with more devices, channel congestion leads to higher packet loss rates. Second, an attempt at establishing a connection to an infected device that is not in paging scanning state because of its own scanning activity can not succeed.

3.3 Explaining Dilemmas of The Random Walk Model

It is noticed that both dilemmas introduced in Sections 3.1 and 3.2 involve the random walk model. Recall that in our simulation study a device under the random walk model updates its direction every 5 seconds. Further examination reveals that this model updates the direction of a device much more frequently than the other three. Now we examine how much impact such a mobility pattern has on the degree to which devices in the network are mixed with each other. For each mobility trace, we take a snapshot every Δ seconds after simulation time 600 second. Since each simulation trace lasts 3600 seconds, we have $\lfloor 3000/\Delta \rfloor + 1$ snap shots per trace, where $\lfloor x \rfloor$ denotes the



(1) speed: 1 m/s, first neighbor

(2) speed: 1 m/s, second neighbor

(3) speed: 1 m/s, third neighbor

(4) speed: 3 m/s, first neighbor

(5) speed: 3 m/s, second neighbor

(6) speed: 3 m/s, third neighbor

Figure 8: Fraction of successful connection attempts under different mobility models

largest integer that is not greater than x.

In Figure 9, for each mobility model, we depict the average number of neighbors that a device sees in the current snapshot but not in the previous one, as a function of time step size Δ. It is not surprising to see that for each curve, as we increase the time step size Δ, there are more different neighbors seen by a device between two consecutive snapshots. On the other hand, increasing the device speed brings more dynamics to the device neighborhood relationship, irrespective of the model governing device mobilities. Furthermore, we observe that under the random walk model, the device neighborhood relationship is less dynamic than that under the random direction model, although Figure 4 shows that they have similar node degree distributions, .

The dynamics of the device neighborhood relationship affects the Bluetooth worm propagation process. If the device neighborhood relationship is more static, it is more likely that an infected device observes the same set of neighbors in consecutive infection cycles and therefore, after it infects some neighbors in an infection cycle, it is still surrounded by them in the next one. Hence, the dynamics of the device neighborhood relationship affects how quickly infected and uninfected devices can be mixed together. If the mixing rate is low, infected devices tend to cluster together and one being active may not be able to collect enough neighbors because some of its neighbors that have already been infected do not respond to its inquiry owing to their own scanning activities. This explains Dilemma 1 introduced in Section 3.1: under the random walk model, if the node speed is 1 m/s, an infected device collects fewer neighbors than that under the random direction model, although the node degree distributions produced from both models are similar. However, if we increase the node speed up to 3 m/s, the increased dynamics of the device neighborhood relationship under the random walk model results that infected and uninfected devices are well mixed, thus eliminating the constraint that exists at a lower node speed.

In Section 3.2, we have analyzed normalized link duration distributions. As Dilemma 2 reveals, they are not sufficient in explaining some aspects of the Bluetooth worm propagation. To unveil the cause of Dilemma 2, we examine the evolution of link durations between a pair of nodes for 10 hours under each mobility model. Figure 10 illustrates the status of the link between these two nodes as simulation time advances. For clarity of presentation, we only show part of the whole plot. Figure 10 shows that under the random walk model, links between the two devices tend to come in bursts, while under the other three models, links are more evenly distributed over the whole simulation period.

To further verify this observation, we compress successive links between which the down times are less than a given threshold Ω into a single one. We operate on the same mobility traces used to derive Figure 7 and obtain normalized frequency histograms of compressed link durations as Ω varies among 10, 20, and 30 seconds. The results are illustrated in Figure 11. We observe that as compression threshold Ω increases, the tail of the normalized link duration distribution curve under the random walk model moves farther above those under the random waypoint model and the random direction model, regardless of the node speed. This differs from what we have observed from Figure 7, where the normalized link duration distribution curve under the random walk model has the thinnest tail.

When the node speed is 1 m/s, the tail of the normalized link distribution curve under the random landmark model still overlaps with that under the waypoint model, regardless of Ω, which is similar to what we have observed in Figure 7; when the node speed is 3 m/s, however, as we increase Ω, the tail of the normalized link distribution curve under the random landmark model moves upwards more quickly than those under the random waypoint model and the random direction model. This is because under the random landmark model, there are much more links between a pair of devices than under the other models, as illustrated in Figure 10, and when the node speed is high, the down time becomes so short that a substantial number of successive links can be compressed together.

With the observations made from Figure 11, we now can explain Dilemma 2. Although the tail of the original normalized link duration curve under the random walk model is thinner than those under the other models, if we compress consecutive links that are close to each other into a single link, the tail of the new normalized link duration curve under the random walk model can be fatter than those under the others. There are two reasons that the link down period within a compressed link does not necessarily lead to a failed infection attempt. First, Bluetooth worms in design have timers to recover from lossy links. Recall that the connection timeout is 5.12 seconds. If the link down time is short enough so that the next link appears before the connection timer expires, it is still possible that the connection can be successfully established. Second, the link utilized to discover a neighbor can be different from the one that is used to infect it. For example, suppose that there exists two links between infected device A and uninfected device B. The down time between them is 15.24 seconds. We also assume that B is the third victim on A's neighbor list and the first link disappears 5 seconds earlier before A's inquiry phase finishes (so A's inquiry phase must last longer than 5 seconds). If A cannot connect to both neighbors before B on its list, then A starts to connect to B at exactly the same time when the second link between A and B appears, which makes it possible for A to establish a connection to B successfully.

3.4 Infection Cycle Analysis

In the previous sections, we have discussed how mobility patterns affect the behavior of Bluetooth worms such as the number of neighbors discovered per infection cycle and connection probabilities. With those results as the foundation, this section is aimed at explaining the relative orders of the infection speeds under the four mobility models. We abstract the Bluetooth worm propagation as a continuous process. We use $\beta(I)$ to denote the worm growth rate when there are I devices in the network. Suppose that all infected devices behave exactly the same but they can be at different stages in an infection cycle. When there are I infected devices in the network, let $C(I)$ and $M(I)$ denote the duration of an infection cycle and the number of victims that are infected in an infection cycle respectively. We further use $t(I)$ to denote the time when the I-th device is infected. Consider the $C(I)$ time units after time $t(I)$. During this period, $I \times M(I)$ new devices are infected. Hence, $\beta(I)$ can be approximated as follows:

$$\beta(I) \approx \frac{I \times M(I)}{C(I)} \qquad (3)$$

To rank the worm growth rates when there are I infected devices in the network under the four mobility models, we only need to compare $M(I)/C(I)$. Figure 12 depicts $C(I)$ collected from the simulation results. Let $L_{cycle}(I)$ denote the duration of an infection cycle provided that there are I infected devices. It can be established as follows

$$L_{cycle}(I) = L_{inq}(I) + \sum_{k=1}^{N(I)} D_k(I) + T_{idle}, \qquad (4)$$

where $L_{inq}(I)$ is the duration of the inquiry phase, $N(I)$ is the number of neighbors discovered, and $D_k(I)$ is the time spent on processing the k-th neighbor when there are I infected devices in the network. Recall that T_{idle} is the inactive duration of the worm. Although Figure 6 shows that $L_{inq}(I)$ keeps increasing with I, the number of neighbors an infected device needs to contact in an infection cycle, i.e., $N(I)$, decreases with I, as illustrated in Figure 5. Figure 12 reveals that as the network is increasingly populated with new infected devices, the duration of an infection cycle decreases slightly.

As to the number of neighbors that can be infected in an infection cycle, it is actually decided by two factors. First, it is upper-bounded by the total number of neighbors that are discovered in that infection cycle. Second, it depends on the probability that the infected device can connect to a neighbor and then infect it. If the node speed is lower, links between devices tend to last for a longer time and thus the second factor is less an issue. Hence, how many neighbors can be infected in an infection cycle is, to a higher degree, determined by how many neighbors can be discovered in an infection cycle. On the other hand, if the node speed is higher, although an infected device can meet more neighbors because of higher mixing ratio and thus discovers more neighbors, the links between it and its neighbors last for a shorter time. Under such circumstance, the number of neighbors it can infect in an infection cycle relies more on the second factor.

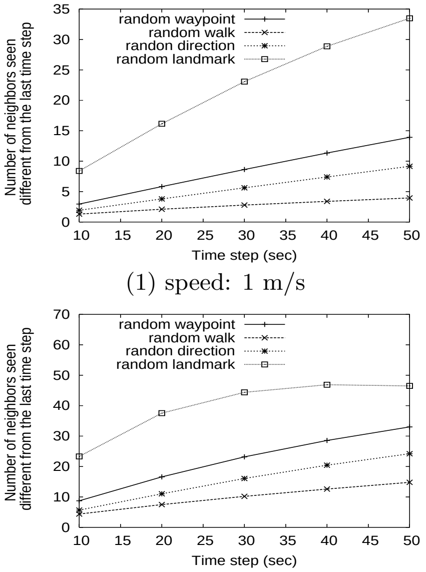
These can be confirmed by our experiments. Figure 13 depicts $M(I)$ collected from the simulation results. Apparently, when the node speed is 1 m/s, the relative orders in $M(I)$ under the four mobility models are exactly the same as their orders in the number of neighbors discovered per infection cycle, which are illustrated in Figure 5. When the node speed is 3 m/s, their relative orders in $M(I)$ can be implied from Figure 8. Although for the third position on the neighbor list, the connection probabilities under the random walk model are higher than those under the random landmark model, an infected device under the random walk model can hardly collect three neighbors (as shown in Figure 5), and therefore, their relative orders in connection probabilities at the first position decide that more neighbors can be infected in an infection cycle under the random landmark model than under the random walk model. Since for all three positions on the neighbor list, the connection probabilities under the random walk model are higher than those under the random waypoint model and the random direction model, an infected device under the random walk model can infect the second largest number of neighbors in an infection cycle. Finally, connection probabilities under the random waypoint model are slightly higher than those under the random direction model for most of the points at the first position, but they are lower than the latter for the other two positions. This leads to close curves between these two models in Figure 13(2).

We then derive $M(I)/C(I)$ from Figures 12 and 13 and depict the results in Figure 14. We find that the shape of each curve is actually dictated by that of $M(I)$. This results from the small differences in $C(I)$ among the four mobility models. If we compare Figure 14 and Figure 2, we can conclude that the relative orders in $M(I)/C(I)$ among the four mobility models reflect their orders in the worm propagation speeds.

3.5 Mathematical Analysis

In the previous sections we introduced two very important aspects for comparing and examining the differences between mobility models as they pertain to worm propagation. Conceptually, the first deals with the effect that the mobility model has on the number of neighbors that an infectious device discovers during an infection cycle. The second pertains to the effect of the mobility on the length of contact between an infectious device and its neighbors.

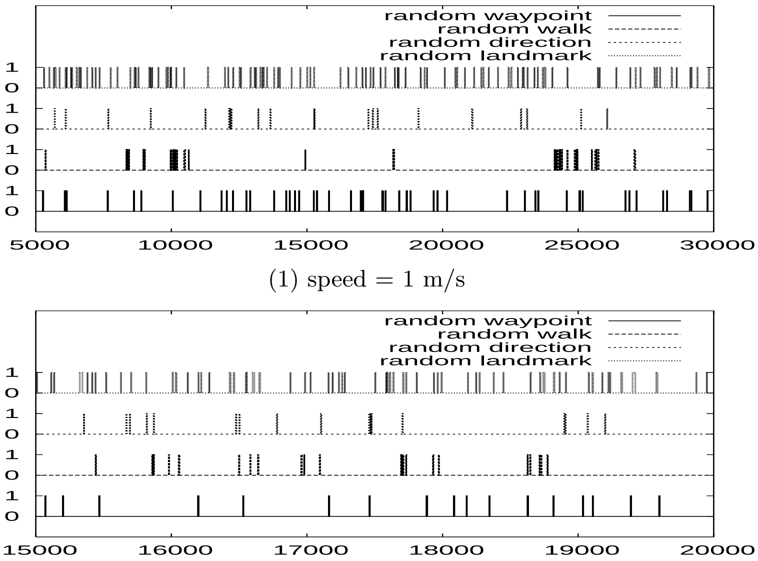
Here we take a high-level view of the dynamics of a single infection cycle, from the perspective of a single infected device. We begin by proposing a simple probabilistic model for the number of new infections that a single infectious device creates following an infection cycle. When an infectious device begins



(1) speed: 1 m/s

(2) speed: 3 m/s

Figure 9: Number of neighbors seen different from the last time step



(1) speed = 1 m/s

(2) speed = 3 m/s

Figure 10: Illustration of the status of the link between two devices (0 indicates that the link does not exists and 1 indicates that the link exists)

its infection cycle, it discovers a random number N of neighbors. Note that the worm parameter N_{inq} can be absorbed into N by replacing N with $\min(N, N_{inq})$. Each discovered neighbor i, where $1 \leq i \leq N$, is within range for time T_i. Some of these neighbors may already be infected. Let X_i be the indicator of whether or not neighbor i is already infected (i.e., $X_i = 1\{i \text{ is already infected}\}$). We assume that the devices are indistinguishable, and in particular that the link duration, infection status pairs $(T_1, X_1), \ldots, (T_N, X_N)$, given N are exchangeable[1]. However, we make no assumptions about the independence of N, and the link durations T_i.

Here, for ease of analysis, we assume that the infectious device requires exactly c time units to transmit its payload to a single neighbor. This time is spent whether or not the neighbor drops its connection or is already infected. Both of these are unknown to the infectious device. The assumption differs from that in the previous simulation experiments but we believe it is not crucial to the conclusions made from the mathematical analysis. The infectious device attempts to infect each neighbor sequentially in a random order. A neighbor will be counted among the new infections iff the payload is delivered while the neighbor is within range and the neighbor is not already infected. Thus, the number of new infections is

$$M = \sum_{k=1}^{N} 1\{T_{\pi(k)} \geq ck\}(1 - X_{\pi(k)}) \tag{5}$$

where π is a uniformly distributed permutation of the first N integers, representing the order in which the infection attempts are made.

Intuitively, the number of new infections should decrease with both the typical number of discovered neighbors and their link durations. Moreover, the number of new infections resulting from a single infection cycle should decrease as the proportion of infected devices increases. The following theorem makes this precise. Its proof is given in the Appendix A.

Theorem 1. *Let $c > 0$ be the time required for an infectious device to transmit its payload. Suppose that the link duration, infection status pairs $(X_1, T_1), \ldots, (X_N, T_N)$ are exchangeable given N. Moreover, assume that the link duration T_i and N are independent of the infection status X_i. Then the number of new infections M satisfies*

$$\mathbb{E}\, M = (1 - p)\, \mathbb{E} \min(N, \lfloor T/c \rfloor), \tag{6}$$

where T is any one of the T_i, and $p = \mathbb{E}\, X_i$ is the chance that a discovered neighbor is infected.

We also have the following crude upper-bound with the proof given in Appendix B.

Corollary 1.

$$\mathbb{E}\, M \leq (1 - p) \min(\mathbb{E}\, N, \mathbb{E}\, T/c). \tag{7}$$

When $p = 0$, the theorem makes a statement about the average number of secondary infections. This number is known in the epidemics literature as the reproductive ratio R_0.

Corollary 2. *The average number of secondary infections*

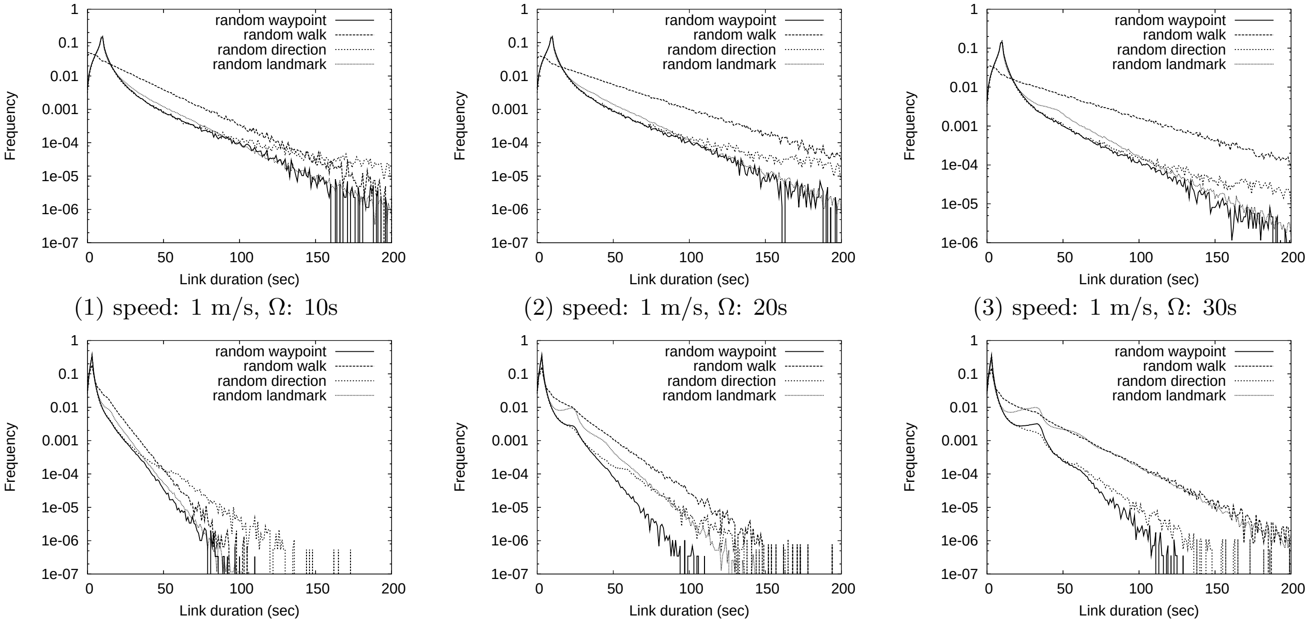
$$R_0 = \mathbb{E} \min(N, \lfloor T/c \rfloor). \tag{8}$$

This result is surprising because it says that the reproductive ratio is exactly equal to the average of the minimum of the number of discovered neighbors N and the scaled, integer link duration $\lfloor T/c \rfloor$. The offense implication for a malicious worm is that increasing the average number of neighbors discovered N alone (e.g., by increasing N_{inq} in a dense area) will not necessarily increase the number of new infections in an infection cycle. Corollary 2 also applies to the reproductive ratio and gives us

Corollary 3. *The reproductive ratio R_0 is bounded above by*

$$R_0 \leq \min(\mathbb{E}\, N, \mathbb{E}\, T/c). \tag{9}$$

[1] The joint probability of the pairs (T_i, X_i) is invariant under permutations of the indices i.



(1) speed: 1 m/s, Ω: 10s (2) speed: 1 m/s, Ω: 20s (3) speed: 1 m/s, Ω: 30s

(4) speed: 3 m/s, Ω: 10s (5) speed: 3 m/s, Ω: 20s (6) speed: 3 m/s, Ω: 30s

Figure 11: Normalized frequency histograms of link durations under different compression thresholds

4. IMPACT OF WORM PARAMETERS

In the previous section, we have observed that mobility models impose significant impact on the propagation speed of the Bluetooth worm under the parameter setting in Table 1. The question we address in this section is: if the Bluetooth worm configures its parameters differently, will the worm propagation speed still be sensitive to the mobility pattern? We pursue the answers from both experimental and analytical perspectives.

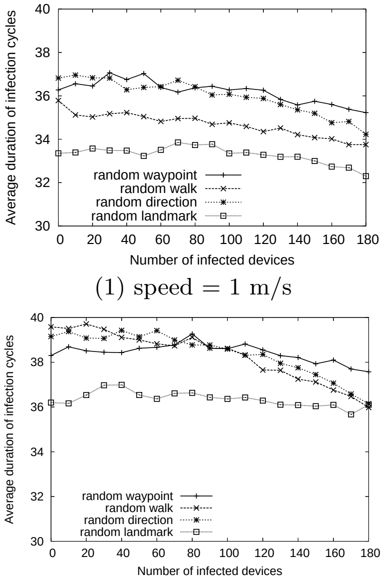
4.1 Experimental Analysis

We perform a new set of experiments in which we vary T_{inq} among 2.56, 5.12, 7.68 and 10.24 seconds, N_{inq} among 1, 3 and 5, and T_{idle} among 1, 10 and 20 seconds. We also vary the node speed between 1m/s and 3m/s. We study how these parameter settings affect the average time needed to infect 75% of the population under different mobility models. Figure 15 presents the results. Note that there are two horizontal axes: the top one shows the inquiry timeout values (i.e., T_{inq}) and the bottom one gives the expected number of inquiry responses (i.e., N_{inq}). Clearly, different mobility models lead to different times needed for 75% infection coverage, suggesting that the impact of device mobility patterns on the propagation of the Bluetooth worm does not disappear even when the worm varies its parameters.

On the other hand, no matter which mobility model is considered, the worm propagation speed is also affected by how the worm sets its parameters. From Figure 15, we observe that in most of the cases, using a larger N_{inq} leads to slower worm propagation. This is because device mobility makes it difficult to infect multiple devices in the same infection cycle. But if devices are clustered in a small region and thus many links between devices can last for a long time, it is possible that choosing a larger N_{inq} improves the propagation speed. For instance, under the random landmark mobility model, if the T_{idle} is 20 seconds, setting N_{inq} to be 3 leads to slightly faster worm propagation than when setting it to be 1, regardless of the node speed.

If we vary the parameter T_{inq}, the degree to which the worm propagation speed is affected depends on the mobility model. T_{inq} is effective only when the infectious device can not discover N_{inq} neighbors in its vicinity. Hence, if in a network devices have plenty of neighbors within their radio ranges, the inquiry timeout has little effect because the inquiry device can easily find N_{inq} victims in a short time. This is exactly what we observe for the random landmark model in Figure 15: if we fix the other parameters, varying T_{inq} leads to little variation on the propagation time for 75% infection coverage. By contrast, under the random walk model, the impact of T_{inq} is prominent. The general trend is that larger T_{inq} leads to slower worm propagation. Although a worm using a larger T_{inq} can discover more devices in a single infection cycle, it has such pitfalls: after the worm infects a victim, the next one on its neighbor list may have already moved out of its radio range, which the worm detects by relying on connection establishing timeout (i.e., T_{conn} in Table 1); furthermore, if a device is located in a sparsely populated region, the worm has to wait for a long time to realize this because of the large T_{inq}. Conversely, if a worm sets T_{inq} too small, it may not receive any response if there exists few neighbors in its vicinity, thus wasting infection cycles and slowing down its spreading. We observe that this occurs when the node speed is 1 m/s and T_{inq} is 2.56 seconds.

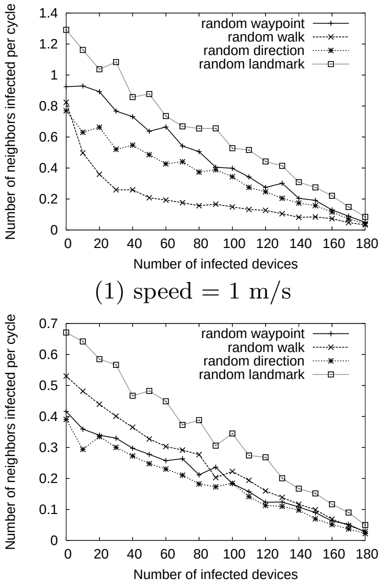
Parameter T_{idle} controls the aggressiveness of the Bluetooth worm: a more aggressive worm keeps idle for a shorter time in an infection cycle. From Figure 15, we observe that generally speaking, reducing the inactive duration of the worm helps shorten the worm propagation time. However, we also notice that a worm aggressively scanning for new neighbors can have negative effects on its propagation speed. First, if devices are sparsely populated in an area, the infection rate of a worm is limited by the number of neighbors it has and therefore, being more aggressive may not lead to its faster propagation. Second, in an area with dense Bluetooth-enabled devices, an aggressive worm can lead to severe channel congestion, which can slow down its propagation. Finally, an aggressive worm can drain the power of infected devices rapidly, which may limit their capability of infecting new victims in their vicinity.



(1) speed = 1 m/s

(2) speed = 3 m/s

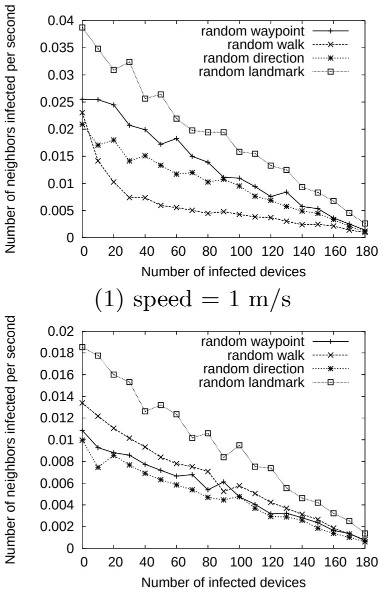
Figure 12: Average duration of infection cycles vs. number of infected devices



(1) speed = 1 m/s

(2) speed = 3 m/s

Figure 13: Number of neighbors infected per infection cycle vs. number of infected devices



(1) speed = 1 m/s

(2) speed = 3 m/s

Figure 14: Number of neighbors infected per second vs. number of infected devices

4.2 Mathematical Analysis

Mathematically analyzing the impact of Bluetooth worm parameter setting under a general mobility condition is difficult, if not impossible. Hence, we resort to studying some simple examples to gain some insights from this perspective.

Example 1. In the first example, we study a static network in which infectious device 0 has m susceptible ones within its radio range. We assume that $m > 1$. Let S_m denote the set of these m susceptible devices. We also assume that the time that the infectious device takes to receive the first response after an inquiry is broadcast is invariably τ_{rsp}, and the time interval between any two consecutive responses is also τ_{rsp}. We also let τ_{inf} be the time required for device 0 to infect any device in S_m. To simplify analysis, we assume that a device infected by device 0 will not attempt to infect another device in S_m before device 0 infects it. The worm has two schemes to set N_{inq}. In the first one, N_{inq} is set to be m. Since all devices are static, all the m devices can be infected by device 0 in a single infection cycle. We use T_1 to denote the duration of the single infection cycle, in which device 0 can infect all devices in S_m, and it is given by

$$T_1 = m(\tau_{rsp} + \tau_{inf}) + T_{idle}. \tag{10}$$

In the second scheme, N_{inq} is set to be 1. Let T_2 denote the total duration of all the infection cycles that are needed to infect the m susceptible devices. As device 0 discovers only one neighbor in its vicinity in each infection cycle, the first one that responds to its inquiry may have already been infected. Let *Assumption* A_1 be that it is equally likely that any device in S_m can become the first one responding to an inquiry request by device 0. The following theorem gives $E(T_2|A_1)$, the expectation on T_2 under Assumption A_1.

Theorem 2. *Under Assumption* A_1*, the expectation on* T_2 *is* $m(\tau_{rsp} + \tau_{inf} + T_{idle}) + \Delta$*, where* Δ *can be approximated by* $(\tau_{rsp} + T_{idle})(m \cdot ln(m) + \gamma m - 1)$*, in which* $\gamma \approx 0.577$ *is the Euler-Mascheroni constant.*

The proof is provided in Appendix C. In our earlier work [17], we noticed that Assumption A_1 overestimates the fraction of inquiry responses from those devices that have already been infected. This is because an infected neighbor, if it is also performing worm scanning activities, may not be discovered by device 0. Therefore, we have $\mathrm{E}(T_2) \leq \mathrm{E}(T_2|A_1)$. On the other hand, if we assume that in every infection cycle, device 0 can find a device in S_m that has not been infected yet, we can get the lower bound on $\mathrm{E}(T_2)$. Let A_2 denote this assumption. We thus have

$$\mathrm{E}(T_2|A_2) = m(\tau_{rsp} + \tau_{inf} + T_{idle}) \tag{11}$$

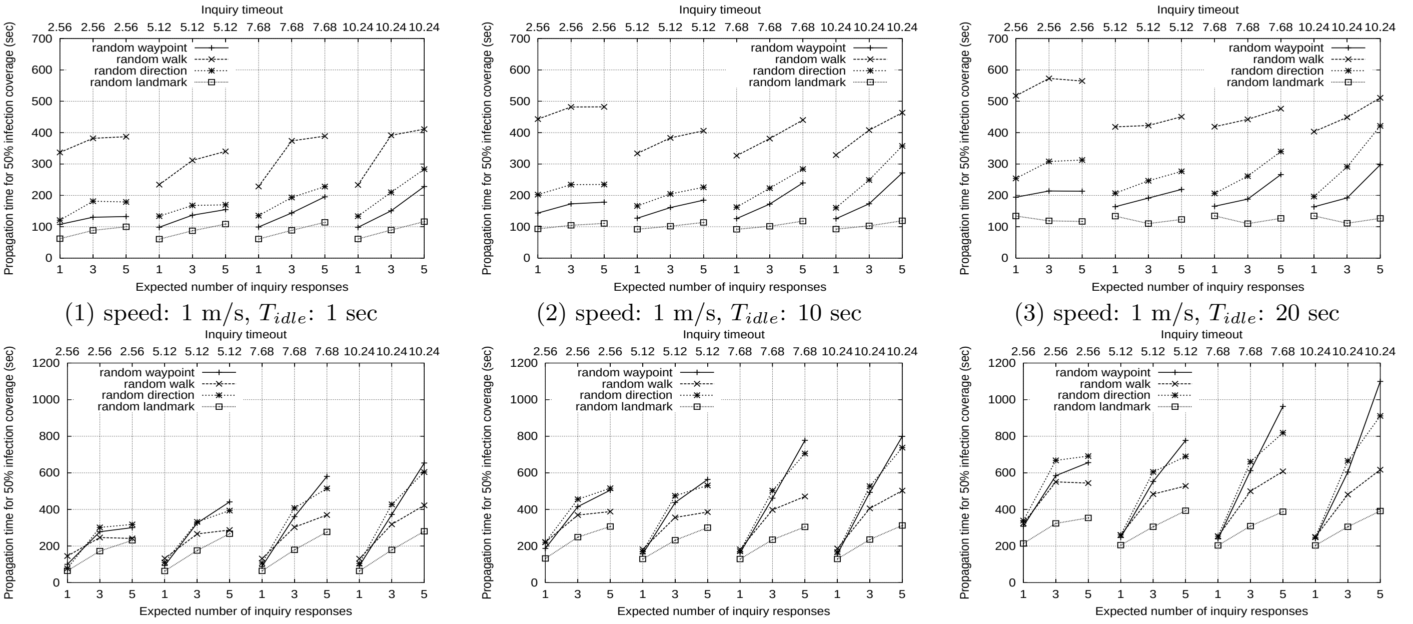
and

$$\mathrm{E}(T_2|A_2) \leq \mathrm{E}(T_2) \leq \mathrm{E}(T_2|A_1). \tag{12}$$

Comparing Equations (12) and (10), we can make two observations. First, using a larger N_{inq} leads to less time that the infectious device spends on the idle phases. T_1 has only one T_{idle} item in it but T_2 has at least m T_{idle} items. On the other hand, using a small N_{inq} increases the probability that a neighbor that has already been infected is discovered by the infectious device in an infection cycle; moreover, this probability grows with the number of infected devices around the infectious device.

Example 2. In the second example, we assume that devices in the network are mobile and there are m susceptible devices located in the radio range of an infectious device. We consider the scenario in which only the first device can be infected in an infection cycle because of mobility; in other words, once the infectious device infects the first victim, no other neighbors that it has discovered still remain within its radio range.

In this example, we use the same definitions on τ_{inq} and τ_{inf} as in the previous one. Similarly, we compare the two cases in which N_{inq} are m and 1 respectively. In both cases, the infectious device can infect only one device in a single infection cycle. Therefore, in the first case, there are $m - 1$ failed connection establishing attempts and the duration of an infection



(1) speed: 1 m/s, T_{idle}: 1 sec

(2) speed: 1 m/s, T_{idle}: 10 sec

(3) speed: 1 m/s, T_{idle}: 20 sec

(4) speed: 3 m/s, T_{idle}: 1 sec

(5) speed: 3 m/s, T_{idle}: 10 sec

(6) speed: 3 m/s, T_{idle}: 20 sec

Figure 15: Average propagation time for 75% infection coverage.

cycle, denoted by T'_1, is thus

$$T'_1 = m \cdot \tau_{rsp} + \tau_{inf} + (m-1) \cdot T_{conn} + T_{idle}. \qquad (13)$$

Recall that T_{conn} is the expiration timeout for establishing a connection to a victim device. The duration of an infection cycle in the second case, denoted by T'_2, is

$$T'_2 = \tau_{rsp} + \tau_{inf} + T_{idle}. \qquad (14)$$

Comparing Equations (13) and (14), we can easily see that using a small N_{inq} for the worm propagation in the mobile network not only shortens the inquiry phase but also eliminates the time spent on failed attempts of establishing connections to those neighbors that have moved out of the radio range.

5. RELATED WORK

There have been substantial efforts on understanding how mobility models affect the wireless network protocols. In [8], Camp et al. found that typical performance metrics of an ad hoc network protocol, such as data packet delivery ratio and end-to-end delays vary significantly with different mobility models. In [4], Bai et al. also observed that mobility models have noticeable impact on network performance metrics such as throughput and routing overhead. In our work, we investigate the impact of mobility models in a different application domain, which is Bluetooth worm propagation. As the Bluetooth worm spreads through neighboring devices, path-oriented metrics such as path duration and path availability are not important. Moreover, to the best of our knowledge, some metrics discovered in this paper, such as the burstiness of successive links, have never been investigated before.

Malware propagation in mobile networks has also been studied in a few papers. Mickens et al. [14] developed a probabilistic queueing system for predicting the steady-state infection level in a mobile network. In [12], Khayam et al. proposed a topologically aware worm propagation model for stationary wireless sensor networks. Anderson et al. [3] simulated mobile contagion using mobility traces collected from a campus wireless network. In [6], Bose et al. gave a comprehensive survey on existing mobile viruses and worms exploiting Bluetooth services. Hoh et al. presented in [10] approaches to estimating the front wave speed of the mobile worm propagation, which can be used in quarantine defense.

6. CONCLUSIONS

In this paper, we have investigated how mobility patterns affect Bluetooth worm propagation. In the future, we will apply more realistic mobility models and explore other unknown factors that affect Bluetooth worm propagation. We also plan to build a model for Bluetooth worm propagation which incorporates the statistical properties of mobility patterns that have been discussed in this paper.

7. REFERENCES

[1] The Network Simulator - ns-2. http://www.isi.edu/nsnam/ns/index.html.

[2] UCBT - Bluetooth Extention for NS2 at the University of Cincinnati. http://www.ececs.uc.edu/ cdmc/ucbt/ucbt.html.

[3] E. Anderson, K. Eustice, S. Markstrum, M. Hansen, and P. Reiher. Mobile contagion: Simulation of infection and defense. In *Proceedings of the 19th Workshop on Principles of Advanced and Distributed Simulation*, pages 80–87, 2005.

[4] F. Bai, N. Sadagopan, and A. Helmy. IMPORTANT: A framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks. In *Proceedings of IEEE Infocom'03*, 2003.

[5] C. Bettstetter. Smooth is better than sharp: A random mobility model for simulation of wireless networks. In *Proceedings of ACM Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pages 19–27, 2001.

[6] A. Bose and K. G. Shin. On mobile viruses exploiting messaging and bluetooth services. In *Proceedings of the Second International Conference on Security and Privacy in Communication Networks*, 2006.

[7] J. Bray and C. Sturman. *Bluetooth: Connect Without Cables.* Prentice Hall, December 2000.

[8] T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. *Wireless Communication & Mobile Computing (WCMC): Special Issue on Mobile Ad*

Hoc Networking: Research, Trends and Applications, 2(5):483–502, 2002.

[9] P. Ferrie, P. Szor, R. Stanev, and R. Mouritzen. Security responses: Symbos.cabir. Symantec Corporation, 2004.

[10] B. Hoh and M. Gruteser. Computer ecology: Responding to mobile worms with location-based quarantine boundaries. In *Proceedings of International Workshop on Research Challenges in Security and Privacy for Mobile and Wireless Networks*, 2006.

[11] J. P. Lynch Jr. Co-channel interference in bluetooth piconets. Master's thesis, Virginia Polytechnic Institute and State University, 2002.

[12] S. A. Khayam and H. Radha. A topologically-aware worm propagation model for wireless sensor networks. In *Proceedings of The 2nd International Workshop on Security in Distributed Computing Systems (SDCS-2005)*, June 2005.

[13] M. Lactaotao. Security information: Virus encyclopedia: Symbos_comwar.a: Technical details. Trend Micro Incorporated, 2005.

[14] J. W. Mickens and B. D. Noble. Modeling epidemic spreading in mobile environments. In *Proceedings of the 4th ACM workshop on Wireless security*, pages 77–86, September 2005.

[15] S. Souissi and E. F. Meihofer. Performance evaluation of a bluetooth network in the presence of adjacent and co-channel interference. In *Proceedings of IEEE Emerging Technologies Symposium on Broadband Wireless Internet Access*, 2000.

[16] C. Taylor and N. Mawston. Bluetooth market doubles: CSR still gaining momentum. http://www.strategyanalytics.net/, December 2005.

[17] G. Yan and S. Eidenbenz. Bluetooth worms: Models, dynamics, and defense implications. Proceedings of the 22nd Annual Computer Security Applications Conference, 2006.

[18] J. Yoon, M. Liu, and B. Noble. Random waypoint considered harmful. In *Proceedings of IEEE Infocom'03*, 2003.

[19] Q. Zheng, X. Hong, and S. Ray. Recent advances in mobility modeling for mobile ad hoc network research. In *Proceedings of the 42nd annual Southeast regional conference*, 2004.

[20] J. Zyren. Intersil corporation - prism products. White Paper, June 1999.

Appendix A

Proof of Theorem 1. We first note that since the link durations T_i are exchangeable, the order of infections π is irrelevant when taking expectation. So

$$\begin{aligned}\mathbb{E}\,M &= \mathbb{E}\sum_{k=1}^{N} 1\{T_{\pi(k)} \geq ck\}(1 - X_{\pi(k)}) && (15)\\ &= \mathbb{E}\sum_{k=1}^{N} 1\{T_k/c \geq k\}(1 - X_k) && (16)\\ &= \mathbb{E}\sum_{k=1}^{\infty} 1\{N \geq k, T_k/c \geq k\}(1 - X_k) && (17)\\ &= \sum_{k=1}^{\infty} \mathbb{E}\left[1\{N \geq k, T_k/c \geq k\}(1 - X_k)\right], && (18)\end{aligned}$$

where we have used Fubini's Theorem to interchange the expectation and infinite summation. Since X_k is independent of T_k and N,

$$\sum_{k=1}^{\infty} \mathbb{E}\,1\{N \geq k, T_k/c \geq k\}(1 - X_k)$$
$$= (1-p)\textstyle\sum_{k=1}^{\infty} \mathbb{P}(N \geq k, T/c \geq k) \qquad (19)$$

Now for integer-valued k, $T/c \geq k$ is equivalent to $\lfloor T/c \rfloor \geq k$. So applying Fubini's Theorem once more, we have

$$\begin{aligned}&(1-p)\sum_{k=1}^{\infty} \mathbb{P}\{N \geq k, T/c \geq k\}\\ &= (1-p)\,\mathbb{E}\sum_{k=1}^{\infty} 1\{N \geq k, \lfloor T/c \rfloor \geq k\} && (20)\\ &= (1-p)\,\mathbb{E}\sum_{k=1}^{\min(N,\lfloor T/c \rfloor)} 1 && (21)\\ &= (1-p)\,\mathbb{E}\min(N, \lfloor T/c \rfloor). && (22)\end{aligned}$$

Appendix B

Proof of Corollary 1. Clearly,

$$\mathbb{E}\min(N, \lfloor T/c \rfloor) \leq \mathbb{E}\,N \qquad (23)$$

and

$$\mathbb{E}\min(N, \lfloor T/c \rfloor) \leq \mathbb{E}\lfloor T/c \rfloor. \qquad (24)$$

So

$$\mathbb{E}\min(N, \lfloor T/c \rfloor) \leq \min(\mathbb{E}\,N, \mathbb{E}\lfloor T/c \rfloor) \leq \min(\mathbb{E}\,N, \mathbb{E}\,T/c). \qquad (25)$$

Appendix C

Proof of Theorem 2. Under assumption A_1, we can derive the expectation on T_2 as follows. Let *state* i, $0 \leq i \leq m$, denote the state in which there are i devices that have been infected by device 0. Given that the network is at state i when a new infection cycle starts, after the infection cycle, the network stays at state i with probability i/m and it goes to state $i+1$ with probability $(m-i)/m$. We can thus construct a Markov chain, which is illustrated in Figure 16.

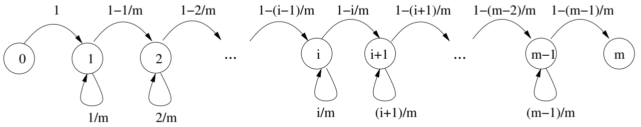


Figure 16: The Markov chain under assumption A_1

From Figure 16, we know that in order for device 0 to infect all m susceptible devices, there must be exactly m forward state transitions. Each forward state transition corresponds to an infection cycle that has a new device infected. The duration of such an infection cycle is $\tau_{rsp} + \tau_{inf} + T_{idle}$. On the other hand, an infection cycle that has no device infected corresponds to a recurrence in the Markov chain. Because no device is infected, the duration of such an infection cycle is $\tau_{rsp} + T_{idle}$. To compute the expected time to infect all m devices, we need to calculate the expected number of recurrences from state 0 to state m. Let p_i denote the recurrence probability at state i, where $1 \leq i \leq m-1$. Obviously, p_i is i/m. We also use R_i to denote the number of recurrences that occur at state i. R_i is actually a geometric process. Hence, the expectation on R_i is

$$\mathrm{E}(R_i) = \sum_{k=1}^{+\infty} k \cdot p_i^k \cdot (1 - p_i) = \frac{1}{1 - p_i} = \frac{m}{m - i} \qquad (26)$$

Thus, the expectation on T_2 under assumption A_1, $\mathrm{E}(T_2|A_1)$, is

$$\mathrm{E}(T_2|A_1) = m(\tau_{rsp} + \tau_{inf} + T_{idle}) + \Delta, \qquad (27)$$

where

$$\begin{aligned}\Delta &= (\tau_{rsp} + T_{idle})\sum_{i=1}^{m-1} \mathrm{E}(R_i)\\ &= (\tau_{rsp} + T_{idle})(m\sum_{i=1}^{m}\frac{1}{i} - 1)\\ &\approx (\tau_{rsp} + T_{idle})(m \cdot \ln(m) + \gamma m - 1) && (28)\end{aligned}$$

where $\gamma \approx 0.577$ is the Euler-Mascheroni constant. In Equation (28), $\sum_{i=1}^{m} \frac{1}{i}$, often called the finite Harmonic series, can be approximated by $\ln(m) + \gamma$. □