WORM INFECTION DYNAMICS ON A LOCAL AREA NETWORK

by

Kübra Doğan

B.S., Electrical and Electronics Engineering, Bilkent University, 2016

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Electrical and Electronics Engineering
Boğaziçi University
2018

ACKNOWLEDGEMENTS

Foremost, I would like to thank my dear supervisor, Professor Yagmur Denizhan, for her invaluable guidance, continuous support, encouragement and enlighting contributions. It is a great honour for me to take her excellent advices and to study with her. It would not be possible for me to write this thesis without her.

I would like to thank to Bilgin Metin, who provided an illuminating introduction to the subject area of my thesis and was always there for further support, to Tuna Tugcu, who generously offered his very limited spare time to answer my questions, and to my committee member Leyla Gören for her clarifying comments.

It is a great pleasure for me to be the member of Dynamical Systems Lab and I would like to thank all DS lab members. Thanks to Özgür and Emre for sharing their experiences, friendship and support. This lab provided us an inspriring working environment thanks to my supervisor Yagmur Denizhan.

Last but not least, I wish to express a sincere thank to my family members, particularly my mother Hacer and my father Veli for their love, support and encouragement. Behind the scenes, my parents are real heroes through this process and this thesis is dedicated to them.

ABSTRACT

WORM INFECTION DYNAMICS ON A LOCAL AREA NETWORK

The aim of this thesis is to investigate the spreading behaviour of worms on a Local Area Network that is supervised by a central administrator who is authorized to enforce the updates on individual computers, but leaves their activation to the users.

For this purpose, the processes involved in a specific worm attack have been modelled with a Markovian approach, and simulated in the Matlab programming environment. Most parameters in the model are set to realistic estimates of the average values of the corresponding variables, while some other parameters are varied in reasonable ranges to investigate their impact on performance of the computers on the LAN. The performance and its deterioration are evaluated in terms of the number of computers that are infected and have to be repaired, as well as in terms of the average percentage of active computers on the LAN.

The proposed model can be used to develop security policies to reduce the average number of infected computers on a LAN. Here, the effects of some parameters related to infection and security on the percentage of infected computers have been investigated.

ÖZET

BİR YEREL AĞDA SOLUCAN ENFEKSİYONUNUN YAYILMA DİNAMİĞİ

Bu tezin amacı, bireysel bilgisayarları güncellemeleri yüklemeye zorlama yetkisine sahip, fakat aktivasyonu kullanıcıya bırakan merkezi bir yöneticinin idaresindeki bir yerel ağda solucanların yayılma davranışını araştırmaktır.

Bu amaçla, spesifik bir solucan saldırısındaki süreçler Markov yaklaşımı ile modellenmiş ve Matlab programlama ortamında simüle edilmiştir. Modeldeki birçok parametre gerçekçi tahminlerle seçilmiş, diğer bazı parametreler ise makul aralıklar içerisinde değiştirilerek yerel ağa bağlı bilgisayarların performansı üzerindeki etkileri incelenmistir. Modelin performansı ve bozunumu, solucanla enfekte olup tamire gönderilen bilgisayar sayısı ve yerel ağ içerisinde aktif bilgisayarların ortalama yüzdesi cinsinden değerlendirilmiştir.

Önerilen model, solucan bulaşmış ortalama bilgisayar yüzdesini düşürecek güvenlik politikaları geliştirmek için kullanılabilir. Burada, enfeksiyon ve güvenlik ile ilgili bazı parametrelerin enfekte bilgisayar yüzdesi üzerindeki etkileri incelenmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

H	Hibernating (computer state)
I	Infected (infection state)
Inact. P	Downloaded, but not activated patch (patch state)
N	Total number of hosts
no P	No downloaded patch (patch state)
not I	Not infected (infection state)
ON-DL	On and downloading the Worm patch (computer state)
P	Active patch (patch state)
R	Under recovery (not working, being cleaned from the Worm) (computer state)
R_0	Basic Reproduction number in KM models
α	Recovery rate
β	infection rate
β_{ext}	External infection rate from the Internet
β_{int}	Probability of an infectious computer infecting another
γ	Probability of infection discovery
δ_{dl}	1/expected value of the patch downloading duration
μ_0	Turning off probability
μ_{rb}	Rate of a computer with downloaded patch being rebooted
ν	1/expected value of the duration of recovery
χ	Rate of hibernation

LIST OF ACRONYMS/ABBREVIATIONS

ABM	Agent Based Modelling
LAN	Local Area Network
ODE	Ordinary Differential Equation
PC	Personal computer
SEIR	Susceptible, Exposed, Infected, Recovered
SEIRS	Susceptible, Exposed, Infected, Recovered, Susceptible
SIR	Susceptible, Infected, Recovered
SIRS	Susceptible, Infected, Recovered, Susceptible
WLAN	Wide Area Network

1. INTRODUCTION

With the developments in Information Technology, communication between computers has become so inevitable that most organizations cannot do their tasks without it. It cannot even be imagined if all of these computers were to collapse suddenly. On the other hand, there is a huge threat by malwares such as worms, viruses, Trojans etc. which cause loss of data, or time, steal confidential information, or infect other hosts in computer networks. Among the various types of malwares, worms differ from the others in terms of their spreading behavior [1]. Worm is a computer program which can easily propagate itself by exploiting the vulnerabilities of the operating systems (OS) without the interference of any other program or human help [1]. Therefore, this thesis focuses on worm infection dynamics because of their distinguishing propagation characteristics and because they are one of the most common types of malware. Especially today, worms can be very destructive for network security and can cause financially disastrous losses. The latest incident of WannaCry which was the most detrimental malware attack in 2017 [2], constitutes an illustrative example. Security experts estimate that 230.000 systems all over the world have been infected by the WannaCry ransomware attack. Another worm attack known as Code Red infected 360.000 computer in minutes all over the world [1]. The severity of these issues has motivated this thesis that aims to model the infection dynamics on a Local Area Network.

Before proposing a simulation model for the spread of infections, we need to explain the meaning of the word 'epidemic'. Originally, the notion of epidemic is used to represent a rapid spread of contagious diseases in populations of living beings. The same notion can also be used for the spread of malware in the community of computers that are in contact via a network.

The aim of this thesis is to investigate the spreading behavior of computer worms and to develop a simulation based model for their propagation on a Local Area Network. This also requires the modelling of the operating behavior of the computers, i.e., how they are turned on or off, or enter hibernation. In this thesis, all events involved in

the dynamics of computers and worms have been modelled with a stochastic approach based on Markovian assumptions. The model is designed to investigate the spread of a single worm, starting from the moment when the patch that disables this worm has been received by the LAN administrator. To account for the possibility of infection until then, different scenarios are considered, that assume different number of infected computers at the beginning of simulations.

1.1. Literature Survey

Epidemic modelling of diseases in populations and epidemic modelling of an attack of worm attack in computer networks are closely related. For this reason, a brief literature survey for each of them is given below.

1.1.1. Biological Infections and Mathematical Epidemiology

Throughout the human history, there have been numerous epidemic diseases causing millions of deaths and recurring periodically in time. To be more specific, "The Black Deaths" in 1346 have spread from Asia to Europe and recurred periodically over the course of 300 years [3]. Instigated by the immense damages caused by various epidemics, some public health physicians set up the foundation of compartmental models and started to develop a mathematical approach to epidemiology. Their main goals were to investigate the causes of a disease, to model its spreading process and based on these models to control the spreading behavior of the disease. First recordings of data about an epidemic date back to John Graunt (1620-1674) who published the number of deaths caused by an infectious disease in London [4]. Based on these continuous records, he estimated the cause of deaths and came to statistical conclusions. The first mathematical epidemiological model was developed by Danielle Bernoulli who worked on modelling an infectious disease (smallpox) in 1760 [5] and he was the first to relate the number of susceptibles in an infected population to life expectancy and the severity of the disease. Bernoulli's goal was to estimate the life expectancy of individuals in the absence of smallpox as a cause of death [5]. In 1850s, cholera outbreak in London was believed to be miasma in the air [6]. However, John Snow asserted

that cholera is caused by contaminated water and showed the number of deaths as evidence in Table-1 [7]. In this respect, John Snow who changed the commonly believed

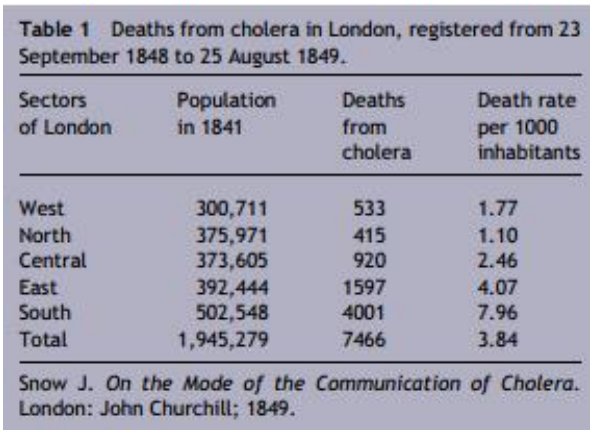

Table 1 Deaths from cholera in London, registered from 23 September 1848 to 25 August 1849.

Sectors of London	Population in 1841	Deaths from cholera	Death rate per 1000 inhabitants
West	300,711	533	1.77
North	375,971	415	1.10
Central	373,605	920	2.46
East	392,444	1597	4.07
South	502,548	4001	7.96
Total	1,945,279	7466	3.84

Snow J. *On the Mode of the Communication of Cholera.* London: John Churchill; 1849.

Figure 1.1: Deaths caused by cholera in London from 1848 to 1849.

cause of infectious diseases until then, is accepted as one of the founders of modern epidemiology [6]. Similarly, in 1873, William Budd studied on the spread of typhoid [6].

In 1927, Kermack and McKendrick [8] developed a compartmental mathematical model of epidemics where the population is divided into different compartments of individuals by making some assumptions. The assumptions are that the population size is fixed (i.e. the number of deaths by natural cause and the number of births are equal.) and that the population is homogeneous with regards to age. This model subdivides to population into three subgroups: (i) Susceptibles (S) who are healthy but not immune to the disease, and thus open to infection, (ii) Infected (I) who are infected and have the potential of infecting others, and (iii) Recovered (R), who have been infected and have recovered from the disease such that they are immune against infection and cannot infect others, either. This model which is called the Kermack-McKendrick model or SIR model consists of three ordinary differential equations governing the dynamics of the sizes of the three compartments.

In 1957, MacDonald defined a threshold value which is called "Basic Reproduction Number (R_0)" as the expected number of a disease by an infectious person in a susceptible population [9]. This threshold plays a decisive role in whether a disease

will spread further or disappear over time.

In 1979, Roy M. Anderson and Robert M. May [10] developed mathematical models to propose protection strategies against common infections for controlling public health by assuming the population size is not fixed, but it can change through time.

In 2004, Wendi Wang and Xiao-Qiang Zhao developed an epidemic model which describes the dynamics of disease propagation if some individuals in a population migrate to another country [11]. Besides, they have used one of the common mathematical model of KM.

In 2011, Zhisheng Shuai, P. van den Driessche developed a compartmental model which explains the transmission ways of Cholera and the states of pathogen [12]. Their model is improved by using Lyapunov functions and basic reproduction number (R0) to estimate whether Cholera will die out or not.

1.1.2. Computer Infections and the Distinguishing Features of Worms

Malwares are the malicious software programs, such as Trojans, viruses, or worms which infect computers via emails, attached files, or malicious websites [1]. The users usually realize the infection on their PCs if the operating systems of their PC perform its tasks very slowly. Another symptom of such an infection may be the existence of some non-erasable files on their PCs. Among all malware types, viruses and worms are the most common types that threaten the network security severely [1]. Viruses can only spread if a user open or run a malicious software program or file [1]. On the other hand, worms can spread easily without any human help since they are self replicating codes [1]. This ability to spread by themselves makes very similar to biological infections. Besides, worms affect all operations of the host; for instance, they can disable its communication, can damage most of the files or other programs. While viruses can only affect specific files [1]. Therefore, the focus of this thesis has been chosen as worm infection dynamics in a Local Area Network (LAN).

The history of worm attacks goes back to early 1970's. Some important worms, their features and the damage caused by them are summarized in [1, 11, 13–16] as follows: The first worm attack is known as Morris by Robert Tappan Morris (1988). It affected computers connected to the Internet and its damage was over 10 million USD. After that, one of the first computer worms known as Creeper by Bob Thomas was released. Interestingly, its goal was not to damage the computers, but it was designed as an experimental program to see the power of coding. When it is copied to a new machine, it wrote a message on to the screen as: "I'm the creeper, catch me if you can!". Another important worm known as ExploreZip by an unknown author was released in 1999, and spread through Microsoft Outlook. It was transmitted as zipped files by mailing itself to the people who are in the contact list of Outlook. In 2000, Irene and Onel de Guzman Reomel Lamores have released the worm called ILOVEYOU which spread through Outlook mails and copied itself to the contact list. Its particular characteristic was to steal the password of credit cards. Its damage is estimated as 5 to 10 billion USD. Then, in 2001, Code Red and Code Red II have been released by an unknown author, damaging IIS servers and falsifing some websites by using buffer overflow. Their damage is estimated as 1.2 billion USD and 2 billion USD respectively. After that again in 2001, an unknown author author has released the worm known as Nimda, which propagated via different media, such as LAN, websites, executables and emails. It caused DoS (Denial of Service) all over the World causing an estimated damage of 8.75 billion USD. Last but not least, the worm known as WannaCry was released in 2017, harming more than 200.000 people over 150 countries. This type of worm freezes all data on the PC and asks the user to make some payment before the PC can become functional again. It propagated via emails or websites.

The historical examples of computer worms show that the intention behind releasing worms is generally to steal personal information, to have DoS on the host, and therefore, to cause loss of working time and data. The unique feature of worm is to be self-replicating on the host, and to scan other vulnerable hosts to propagate further. Besides, the authors of worms can delete or damage the files on the victim computer remotely by creating back doors [1].

1.1.3. Concepts related to Computer Worms and Local Area Network (LAN)

This section presents commonly used definitions, terminologies, and concepts related to computer networks, LAN, and worms. Here also, the environment, in which computer worms spread and infect hosts are explained.

- Host: Each computer is a host and hosts are connected to each other in a network.
- Network: Computers connected to each other in order to share data and communicate constitute a network. There are two main types of networks which are Local Area Network (LAN) and Wide Area Network (WLAN) [17].
- Local Area Network (LAN): LAN is a type of network for a small area. For instance; a network in a school building or in an office building can be called as a LAN. In other words, this type of network is confined in a specific area [18].
- Servers: A computer connected to a network can act as a server. A computer is called as a server, if its job is to provide services and resources for the other computers in the network [18]. Besides, data storage, messaging, providing security for the network are also among the duties of a server [18].
- Malware: It is a malicious software which penetrates and damages the working system of a computer [18]. A malware abuses the resources of computers or networks, thus slowing down its communication speed and reducing its performance.
- Computer Worm: A harmful, self-replicating and malicious software is called a computer worm which can propagate on a network without any human interference [19]. As explained in [19], a worm can penetrate into the system of a computer via the network connection, as a downloaded file or as an email. Once it penetrates the contact list of an email account, it can rapidly reproduce itself and spread across all the contact list. Hence, if a computer worm cannot be detected on time, it can quickly destruct whole networks. A computer worm can delete some files on the victim machine, can steal personal information, built a backdoor listener or can result in denial of service [20].
- Patches: Patches are software programs intended for fixing the bugs, usually security bugs, of an operating system (OS) such as Microsoft, Mac or Linux [18].

In order to repair security vulnerabilities, it is important that the patches and updates are downloaded and installed, as soon as the operating system releases them for its users. If downloading and installation is not performed right after the patch and/or update release, the attackers can quickly and easily abuse the vulnerability of the operating system because through the release of the patch, the existence of open points of the security system becomes openly admitted. For instance; Microsoft offers a service called Microsoft Updates, which offers four options for the downloading and installation of the patches and updates to its users as explained in [21]. These options are summarized as below [21]:

(i) Install patches/updates automatically (recommended): This option is the recommended one for protection against worm attacks. If the user selects this option, updates will be automatically downloaded and installed. Hence, there will be no missing updates and the computer will be kept up-to-date.

(ii) Download patches/updates but let the user choose whether to install them: This option downloads updates automatically, but asks the user for the timing of installations. Choosing this option avoids the automatic interruption of the user's job. After downloading, the user can select to install updates after 10 minutes, 2 hours or 4 hours.

(iii) Check for patches/updates but let the user choose whether to download and install them: This option warns the user but also asks for permission both for downloading and installation. Compared to the first and second options, it is riskier in terms of security of the operating system.

(iv) Never check for patches/updates (not recommended): This option is the most dangerous one because it is possible that the user can miss most of the updates. Without any check or warning about new updates, users who do not follow the updates are put to great risk.

2. COMMON EPIDEMIC MODELS IN THE LITERATURE

As mentioned before, the history of compartmental epidemic models goes back to a study by Kermack and McKendrick (KM) in 1927 [22]. Their aim was to analyze the rapid changes in the number of infected individuals as observed in the notorious epidemics like cholera (1865) and plague (1665) in London [20].

Epidemic models can be largely divided into two categories: (i) Kermack and McKendrick (KM) models, which are deterministic models represented in terms a coupled set of differential equations that govern the dynamics of the various compartments defined on the population, (ii) Stochastic epidemic models which depend on probabilistic representations of occurances related to the epidemics. KM models are also used for modeling worm dynamics on computer networks. To be compatible with the content of this thesis, compartmental epidemic models will be explained by using computer related terminology rather than a biological one. In this section, brief explanations of commonly used compartmental models and the corresponding ordinary differential equations (ODEs) will be presented.

2.1. Compartmental Models

Kermack-McKendrick (KM) model [22] constitutes a compartmental KM model because it subdivides the population (of computers in a network) into the following subgroups:

- Susceptible Hosts: The class of hosts which are not infected, but are susceptible to worm attacks. The size of this class is denoted by S.
- Infected Hosts: The class of hosts which are infected by a worm. The size of this class is represented by I.
- Resistant (Immune) Hosts: The class of hosts that are recovered from a worm

infection, and thus has gained immunity. The size of this class is denoted by R.

Table 2.1: List of parameters and notations for KM models.

Parameter	**Definition**
β	Infection rate
α	Recovery rate
μ	Per capita birth and death rate
γ	Loss of immunity
N	Total number of hosts
S(t)	Total number of susceptible hosts
I(t)	Total number of infected hosts
R(t)	Total number of recovered hosts
E(t)	Total number of exposed hosts
R_0	Basic Reproduction Number

2.1.1. SIR Model

The transition diagram of a typical SIR (Susceptible, Infected, Resistant) model is shown in figure 2.1. In terms of computer terminology, births in a population can be considered as an increase in the number of hosts in a network and deaths in a population can be considered as a host becomes useless after a worm attack.

Figure 2.1: Transition diagram of SIR epidemic model.

In this model, β and α are the transition rates, from Susceptible to Infected and Infected to Recovered respectively. A susceptible host can be infected with a

probability of β. As a result of this transition, susceptible population will decrease and the population of infectious class will increase at the same amount. Likewise, an infectious host can be a recovered host with a certain probability . Therefore, changes in the number of hosts for each state can be represented by a set of nonlinear ordinary differential equations (ODE) which are deterministic.

While setting up those differential equations, there is an assumption that the total number of hosts remains constant (N) over time, such that S(t)+I(t)+R(t)=N.

The ODEs, that make up the SIR model show how the number of individuals in each class changes [17].

$$\frac{dS}{dt} = -\frac{\beta SI}{N} \tag{2.1}$$

$$\frac{dI}{dt} = \frac{\beta SI}{N} - \gamma I \tag{2.2}$$

$$\frac{dR}{dt} = \gamma I \tag{2.3}$$

If the change in the total number of computers is taken into account, the differential equations need to be modified as follows [17].

$$\frac{dS}{dt} = \mu N - \frac{\beta SI}{N} - \nu S \tag{2.4}$$

$$\frac{dI}{dt} = \frac{\beta SI}{N} - \gamma I - \nu I \tag{2.5}$$

$$\frac{dR}{dt} = \gamma I - \nu R \tag{2.6}$$

This model is known as SIR epidemic model with the above ODEs. An important parameter by using these rates is the basic reproduction number R0 which is represented by the ratio of infection rate (β) and recovery rate. (α).

By using the basic reproduction number, endemic and epidemic infections can be analyzed as R_0 has a threshold value, 1 and accordingly [22];

For $R_0 > 1$:If the rate of infection is greater than recovery rate, then infection is called as an endemic infection which means that it will remain in the local area network for any case.

For $R_0 < 1$:If the rate of infection is less than the rate of recovery, then infection is called as an epidemic infection which means it will not remain permanently and it will be eradicated after a while on LAN.

2.1.2. SIRS Model

Another KM model known as SIRS model [17] explains that the recovered individuals (in our case computers) can move to the class of susceptibles by losing their immunity. Accordingly, a new parameter is defined to represent the rate of transition from Recovered to Susceptible.

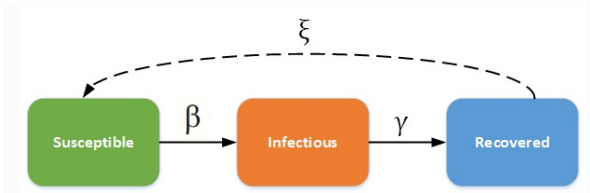

Figure 2.2: Transition diagram for SIRS epidemic model.

The transition is taken as proportional to the size of the recovered class [17].

Hence, the equations in SIRS epidemic model with a constant population can be written as follows:

$$\frac{dS}{dt} = -\frac{\beta SI}{N} + \xi R \tag{2.7}$$

$$\frac{dI}{dt} = \frac{\beta SI}{N} + \gamma I \tag{2.8}$$

$$\frac{dR}{dt} = \gamma I - \xi R \tag{2.9}$$

Taking also the change of the size of the total population into account, the equations of the model are obtained as in equations 2.10-2.12. Here, μ, the rate of new hosts connected to the network and , the rate of computers leaving the network are assumed to be equal to keep the population constant.

$$\frac{dS}{dt} = \mu N - \frac{\beta SI}{N} - \nu S \tag{2.10}$$

$$\frac{dI}{dt} = \frac{\beta SI}{N} - \gamma I - \nu I \tag{2.11}$$

$$\frac{dR}{dt} = \gamma I - \nu R \tag{2.12}$$

2.1.3. SEIR Model

SEIR model has four states namely: susceptible, exposed, infected and resistant [17]. Unlike the previous models, there is a new state, exposed (E) which denotes the class of hosts that had a contact with the infected ones, and they carry the disease. This feature leads to a “latency period”. Thus, exposed state should be between the

susceptible and infected state as a transition state. To be more specific, SEIR model can be shown as below diagram:

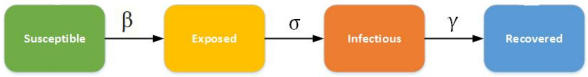

Figure 2.3: Transition diagram for SEIR epidemic model.

Hence; S+E+I+R=N. ODEs for SEIR epidemic model are defined accordingly as below [17].

$$\frac{dS}{dt} = -\frac{\beta SI}{N} \tag{2.13}$$

$$\frac{dE}{dt} = \frac{\beta SI}{N} - \sigma E \tag{2.14}$$

$$\frac{dI}{dt} = \sigma E - \gamma I - \nu I \tag{2.15}$$

$$\frac{dR}{dt} = \gamma I \tag{2.16}$$

As in the previous models, if the change in the total number of computers is taken into account, the differential equations need to be modified as follows [17].

$$\frac{dS}{dt} = \mu N - \frac{\beta SI}{N} - \nu S \tag{2.17}$$

$$\frac{dE}{dt} = \frac{\beta SI}{N} - \sigma I - \nu E \tag{2.18}$$

$$\frac{dI}{dt} = \sigma E - \gamma I - \nu I \tag{2.19}$$

$$\frac{dR}{dt} = \gamma I - \nu R \tag{2.20}$$

2.1.4. SEIRS Model

As in the case of SIR and SIRS epidemic models, loss of immunity brings about a new type of epidemic model called as SEIRS (Susceptible-Exposed-Infected-Susceptible) and a new parameter for loss of immunity [17].

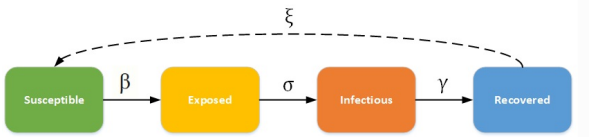

Figure 2.4: Transition diagram for SEIRS epidemic model.

$$\frac{dS}{dt} = \frac{\beta SI}{N} + \xi R \tag{2.21}$$

$$\frac{dE}{dt} = \frac{\beta SI}{N} - \sigma E \tag{2.22}$$

$$\frac{dI}{dt} = \sigma E - \gamma I \tag{2.23}$$

$$\frac{dR}{dt} = \gamma I - \xi R \tag{2.24}$$

With the same reasoning, if the change in the total number of computers is taken into account, the differential equations need to be modified as follows [17].

$$\frac{dS}{dt} = \mu N - \frac{\beta SI}{N} + \xi R - \nu S \tag{2.25}$$

$$\frac{dE}{dt} = \frac{\beta SI}{N} - \sigma E - \nu E \tag{2.26}$$

$$\frac{dI}{dt} = \sigma E - \gamma I - \nu I \tag{2.27}$$

$$\frac{dR}{dt} = \gamma I - \xi R - \nu R \tag{2.28}$$

2.1.5. SIS Model

If there is no permanent immunity against an worms, infected hosts can be susceptible again. This type of epidemic model is called as Susceptible, Infected, Susceptible. The set of ordinary differential equations are described as below.

$$\frac{dS}{dt} = -\frac{\beta SI}{N} + \gamma I \tag{2.29}$$

$$\frac{dI}{dt} = \frac{\beta SI}{N} - \gamma I \tag{2.30}$$

2.2. Stochastic Approaches to Epidemic Modelling

In addition to the deterministic epidemic models, which analytically represent the dynamics of average (infected, recovered, etc.) population sizes, there are also agent

based stochastic epidemic models in the literature, which allow a closer look to the behaviour of individuals and their interactions.

As reviewed in [23], some examples of stochastic epidemic models in the literature are as follows: The first stochastic epidemic model has been proposed by McKendrick who developed the stochastic version of his previous deterministic models. Then, Reed-Frost model which is a SIR epidemic model, that is to say that the population is divided into three categories as defined in the previous section. This model represents state changes by using a Markovian approach. Accordingly, each state depends on the previous state and all state transitions are represented as dependent Poisson processes. In 1949, Bartlett defined the general stochastic epidemic model by defining the state transitions as a probabilistic event. There were also another class of stochastic epidemic models called "Coupling methods". In this approach, the coupling of two different stochastic processes is accounted for as explained by Lindvall (1992).

3. PROPOSED MODEL FOR WORM INFECTION DYNAMICS ON A LOCAL AREA NETWORK

The main concern of this thesis is the modelling of the worm infection dynamics in a LAN that is supervised by a central administrator who has limited authorization. For that purpose, a stochastic model has been developed that uses Markovian assumptions and an agent based approach. In early studies, most of the researchers use deterministic epidemic models by deriving ordinary differential equations (ODEs). However, using ODEs can bring about general solutions and insufficient results. To illustrate this point, although reproduction number can be greater than one and the number of hosts are large but the infection is initiated by a few infectives, it is possible that the epidemic may never be disappeared in the LAN. Like this scenario, there can be many more cases by chance, so to be more realistic, our motivation for this thesis is to address and solve such cases by developing a stochastic epidemic model rather than using general ODEs. In the previous parts, these general ODEs for a population of biological creatures are presented by using computer related terminology to give a framework on epidemic modelling.

There were also a few stochastic epidemic models as explained in Section 2 and our model makes use of Markovian approach and presents more realistic scenarios and simulation results to model the spreading behavior of worms in a network environment.

The assumptions of this model are listed below:

(i) A simple worm attack is considered which from now onward will be referred to as the Worm.
(ii) A uniform LAN without any subgroups.
(iii) The LAN under consideration is maintained by a central administrator who has the authorization to push patch updates on each computer but leaves their activation to the individual users. (recommended update such that of Windows,

refer to [21])

(iv) The model accounts for the worm spread dynamics starting from the instance when the patch or vaccine related to the Worm has been received by the administrator. (This initial time has been preferred in order to avoid the complications of modelling the worm dynamics on the global network.)

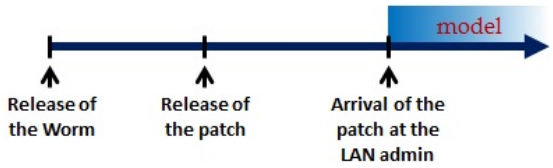

Figure 3.1: Time sequence of the events "worm release", "patch release" and "patch arrival to the network center".

(v) In order to account for the possibility that some computers on the LAN may already be infected by the time when the patch/vaccine is received by the administrator, the simulation is started with a small number of infected and not yet detected computers.

(vi) All processes including the turning on/off and hibernation of computers, individual's activation behavior as well as the infection dynamics are modelled as Markovian processes represented in terms of their rate (=1/expected transition time).

(vii) These stochastic processes are run many times and averages of the variables of interest are presented.

(viii) The variables of interest are: (i)$N_{F\%}$: the percentage of the computers on the LAN that are infected and become subject to loss of working time and data during the recovery, and (ii) $W_{\%}$: the percentage of the average number of computers active on the LAN.

3.1. States and Parameters Defined in the Proposed Model

3.1.1. States

In the proposed model, there are three state categories concerning each computer: Operational state, infection state and patch state.

At any time instance, any computer can be in one of the following operational states:

- off (OFF),
- hibernating (H),
- on and downloading the Worm patch (ON-DL),
- under recovery (R) (not working, being cleaned from the Worm)

At any time instance, any computer can be in one of the following infection states:

- Infected (I),
- Not infected (not I)

At any time instance, any computer can be in one of the following patch states:

- no downloaded patch (no P),
- downloaded, but not activated patch (inact. P),
- active patch (P).

The computers can switch between these states according to continuous transition Markovian dynamics characterized by transition rates.

3.1.2. Parameters

The state transition rates in the model are set to realistic estimates of the corresponding variables by empirical observations while some other parameters are investigated in a reasonable range to evaluate their impacts on the model. Operational state transition rates (λ_0, μ_0, χ) are tuned to obtain an acceptably high network occupancy.

Table 3.1: List of parameters and their nominal values.

Symbol	**Parameter**	**Nominal values**
λ_0	rate of transition from OFF to ON	0.007 min^{-1}
μ_0	rate of transition from ON to OFF	0.0001 min^{-1}
χ	rate of transition from ON to H	0.0001 min^{-1}
β_{ext}	arrival rate of external infection to each host	0.002 min^{-1}
μ_{rb}	1/(expected delay until rebooting)	0.033 min^{-1}
γ	rate of discovery of infected computers	0.004 min^{-1}
δ_{dl}	1/(expected download duration)	0.5 min^{-1}
ν	1/(expected repair duration)	0.0007 min^{-1}
β_{int}	rate of an infected host infecting another	0.7 min^{-1}
m_{ini}	number of infected computers until patch arrival	0; 1; 2; 3; 4
N	number of computers on the LAN	8; 64
R	number of simulation runs	100
Δt	sampling time	1 min

3.2. State Transitions

The proposed approach to model infection dynamics of worms in a Local Area Network is mainly based on a probabilistic simulation in Matlab environment via Random Number Generator which generates uniformly distributed random numbers and used for modelling any probabilistic event.

- All state transitions of any computer are determined by using Markovian transition probabilities.
- To decide on any event in the state transition diagram, a random number is generated by using Random Number Generator.
- This random number is compared with the corresponding Markovian transition probability. In this way, the simulation performs its tasks by following the decided path.

In this simulation, the variables of interest are $NF_{\%}$ and $W_{\%}$. The nominal parameter values shown in Table 3.1. have been adjusted to achieve reasonably high network operation $W_{\%}$ and reasonably low percentage of repaired computers $NF_{\%}$.

For this purpose, the algorithm of the state transition diagram is determined by checking the operational states, patch states and infection states of each agent.

λ_0: Rate for a closed computer until it is opened. It is calculated as an inverse of duration that a closed computer is opened. If a computer is on, it can be closed with a Poisson rate of λ_0.

μ_0: Rate for an opened computer until it is closed. It is calculated as an inverse of average duration that an opened computer is closed. If a computer is off or in a hibernation mode, it can be opened with a Poisson rate of μ_0.

χ_0: Rate for an on computer until its state changes to hibernate. It is calculated as an inverse of average duration that an on computer is turned out a computer being hibernate.

β_{ext}: Infection rate from the Internet for a computer connected to a LAN. It is calculated as an inverse of average duration that a computer connected to a LAN can be infected. If a computer is on and has no active patch or has an inactive patch, it can be infected with a Poisson rate of β_{ext}. Besides, while downloading a patch, it can get an external infection with a probability of β_{ext}

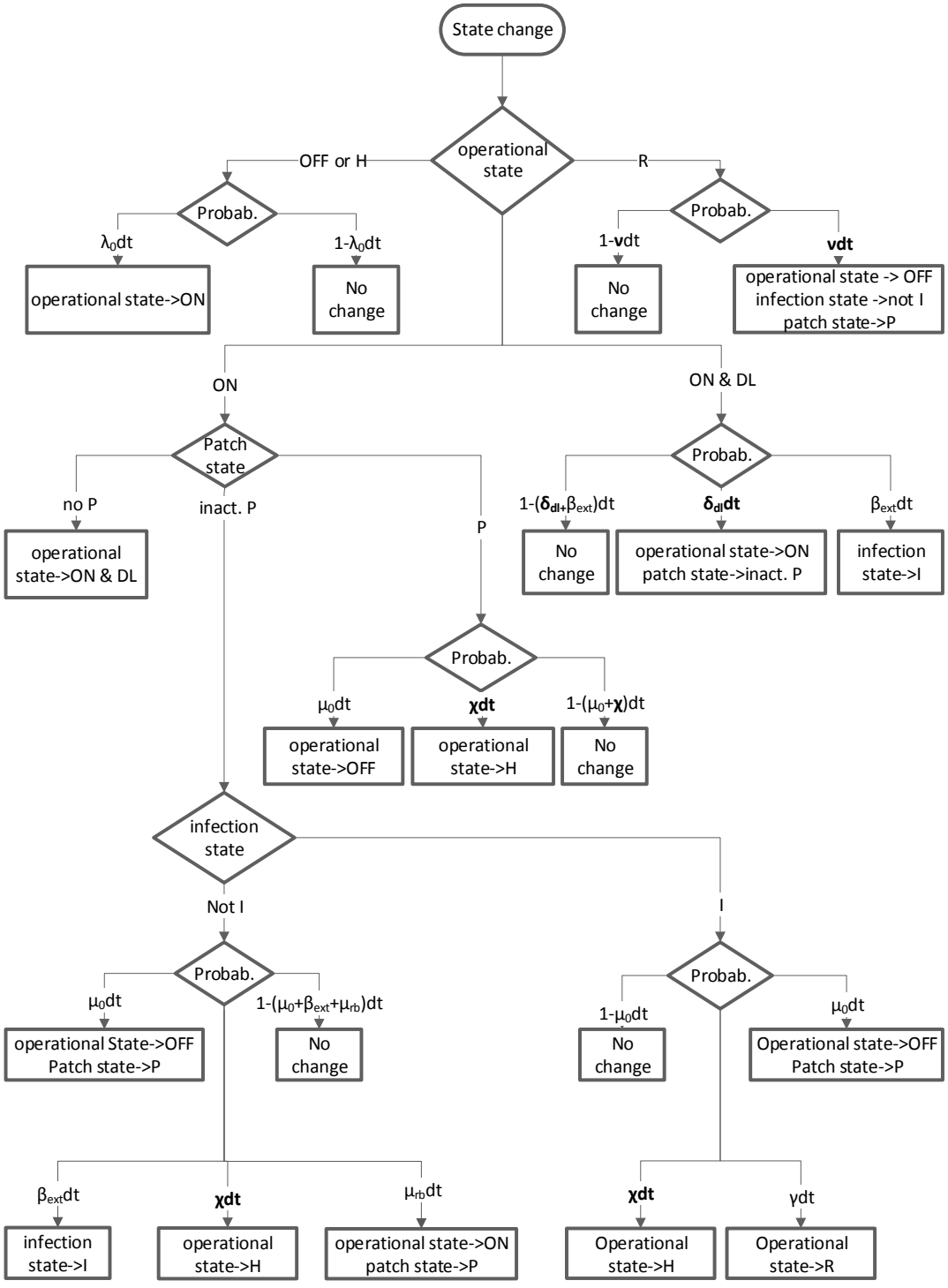

Figure 3.2: State Transition Diagram of the Proposed Model

β_{int}: Internal infection rate. It is the probability of an infected computer infecting another. If some computers have infected files, they can infect others with a Poisson rate of β_{int}.

δ_{dl}: Transition rate from On and downloading state to On state. 1/ expected duration for downloading patch. The nominal value of the patch download duration has been taken as approximately 2 minutes.

γ: Rate of discovery of infected computers. 1/expected duration for an infection discovery.

ν: 1/expected duration for cleaning a computer from an infection. Its nominal value has been taken as approximately 1 day.

μ_{rb}: Rate of rebooting a computer with inactive patch. 1/(expected delay until rebooting after patch download).

3.2.1. Tuning the Nominal Values of Operational State Parameters

In order to obtain an acceptable network occupancy, the operational state transition parameters have been adjusted heuristically. The acceptable network occupancy has been set to approximately $70 - 75\%$. In table 3.1 and table 3.2 some of the trials are shown.

For this purpose, firstly, operational state parameters of the model which represent the rates of turning on/off and hibernation, are adjusted by trial and error to achieve a network workload in such a way that percentage of active computers is approximately $70 - 75\%$ approximately.

Table 3.2: Effects of operational state parameters to have an acceptable network occupancy ($m_{ini} = 0$).

Parameter	**Values**	**Values**	**Values**
λ_0	0.002 min^{-1}	0.002 min^{-1}	0.002 min^{-1}
μ_0	0.001 min^{-1}	0.0005 min^{-1}	0.0005 min^{-1}
χ	0.00028 min^{-1}	0.00028 min^{-1}	0.00014 min^{-1}
β_{ext}	0.003 min^{-1}	0.002 min^{-1}	0.002 min^{-1}
μ_{rb}	0.033 min^{-1}	0.033 min^{-1}	0.033 min^{-1}
γ	0.004 min^{-1}	0.002min^{-1}	0.004 min^{-1}
δ_{dl}	0.5 min^{-1}	0.5 min^{-1}	0.5 min^{-1}
ν	0.0007 min^{-1}	0.0007 min^{-1}	0.0007 min^{-1}
β_{int}	0.7	0.7	0.7
m_{ini}	0	0	0
$W_{\%}$	32.6	37	50.65

The last coloumn shows the best selection of the operational state parameters by achieving the network occupancy as approximately $70 - 75\%$ in Table 3.4.

After having a reasonable performance by adjusting operational state parameters, parameters related to infection state transitions (γ, βint, β_{ext}) are varied in some realistic range to investigate their effect on the percentage of the infected computers on the LAN ($N_{F\%}$).

Table 3.3: List of parameters and simulation results to find the values of operational state transition parameters ($m_{ini} = 0$).

Parameter	**Values**	**Values**	**Values**
λ_0	0.004 min^{-1}	0.004 min^{-1}	0.007 min^{-1}
μ_0	0.0005 min^{-1}	0.0002 min^{-1}	0.0001 min^{-1}
χ	0.00014 min^{-1}	0.0001 min^{-1}	0.0001 min^{-1}
β_{ext}	0.003 min^{-1}	0.002 min^{-1}	0.002 min^{-1}
μ_{rb}	0.033 min^{-1}	0.033 min^{-1}	0.033 min^{-1}
γ	0.004 min^{-1}	0.002min^{-1}	0.004 min^{-1}
δ_{dl}	0.5 min^{-1}	0.5 min^{-1}	0.5 min^{-1}
ν	0.0007 min^{-1}	0.0007 min^{-1}	0.0007 min^{-1}
β_{int}	0.7	0.7	0.7
m_{ini}	0	0	0
$W_{\%}$	63.8	71.9	74.3

Table 3.4 shows the effects of different infection discovery rates on the network occupancy. As the value of γ increases, the network occupancy increases and when the value of γ is 0.004, we have reached the desired $W_{\%}$ value.

Besides, β_{ext} is also one of the parameters related to infection state transitions, but its value is fixed at a rate that allows the testing of different security settings. ν which is the rate for recovery duration is also another parameter related to infection state transitions and its value is taken as a realistic estimate.

Table 3.4: List of parameters and simulation results to reach the desired network occupancy for different γ values ($m_{ini} = 0$).

Parameter	**Values**	**Values**	**Values**
λ_0	0.007 min^{-1}	0.004 min^{-1}	0.007 min^{-1}
μ_0	0.0001 min^{-1}	0.0002 min^{-1}	0.0001 min^{-1}
χ	0.0001 min^{-1}	0.0001 min^{-1}	0.0001 min^{-1}
β_{ext}	0.002 min^{-1}	0.002 min^{-1}	0.002 min^{-1}
μ_{rb}	0.033 min^{-1}	0.033 min^{-1}	0.033 min^{-1}
γ	0.001 min^{-1}	0.002min^{-1}	0.004 min^{-1}
δ_{dl}	0.5 min^{-1}	0.5 min^{-1}	0.5 min^{-1}
ν	0.0007 min^{-1}	0.0007 min^{-1}	0.0007 min^{-1}
β_{int}	0.7	0.7	0.7
m_{ini}	0	0	0
$W_{\%}$	71.5	73.9	74.3

Lastly, other parameters related to patch state transitions are δ_{dl} which is the rate of download duration and μ_{rb} which is the rebooting probability. Their values are taken as realistic estimates. The estimate value of duration to download a patch (δ_{dl}) is considered as an average value. Likewise, rebooting duration is taken as an estimate value.

4. SIMULATION RESULTS AND THEIR EVALUATION

The model presented in Section 3 represents worm infection dynamics on a LAN accomplished with a Markovian approach.

In this section, the change in the system behavior is analyzed while varying the parameters related to infection state transitions within a realistic range. Besides, to exemplify the Worm infection dynamics on a local area network, a small LAN consisting of 8 computers is presented in section 4.1, giving a demonstration of how the states of the individual computers change and interact, such that the propagation of a Worm infection can be easily followed. Performance analysis of the proposed model is presented in section 4.2 in a more detailed way.

4.1. Representation of State Changes on a LAN

At the very beginning of the model, a LAN is set up with 8 computers to see the flow of propagation of any worm attack. Using 8 computers enables us to debug the program, and easily track the sources of infection.

The simulation for representation of state changes on a small LAN with 8 computers is observed in figure 4.1 and figure 4.2 which show the propagation of the Worm and infection source for each computer. At the beginning of the simulation, each computer can be in one of the following operational states: (i) off (OFF), (ii) on (ON), (iii) hibernation (H). These initial operational states are randomly determined by Random Number Generator in Matlab programming environment.

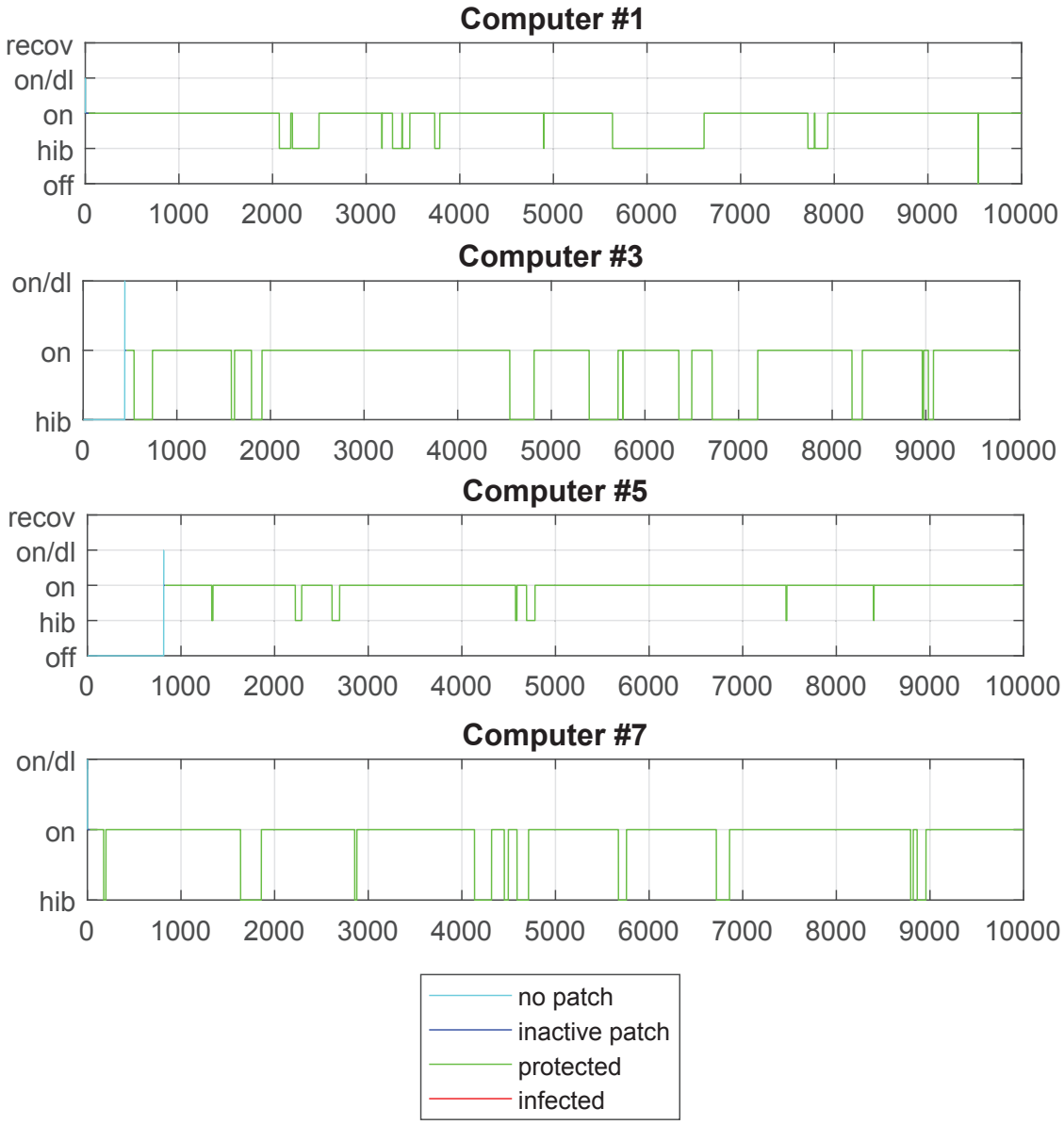

Figure 4.1: Simulation result of worm infection dynamics on a LAN consisting of 8 computers.

The values of the graph are shown with different colors to point out the changes in operational states easily. The vertical axis shows the operational states, while some patch and infection states are shown in color code. The horizontal axis represents the time axis in terms of minutes. In this experiment, firstly, computer 4 and computer 6 are infected externally, and computer 2 and computer 8 is infected by the computer

6, each of them was in "no patch" state when they encountered the Worm. The first four computers, on the other hand, have not infected because they have downloaded and activated the corresponding patch early enough.

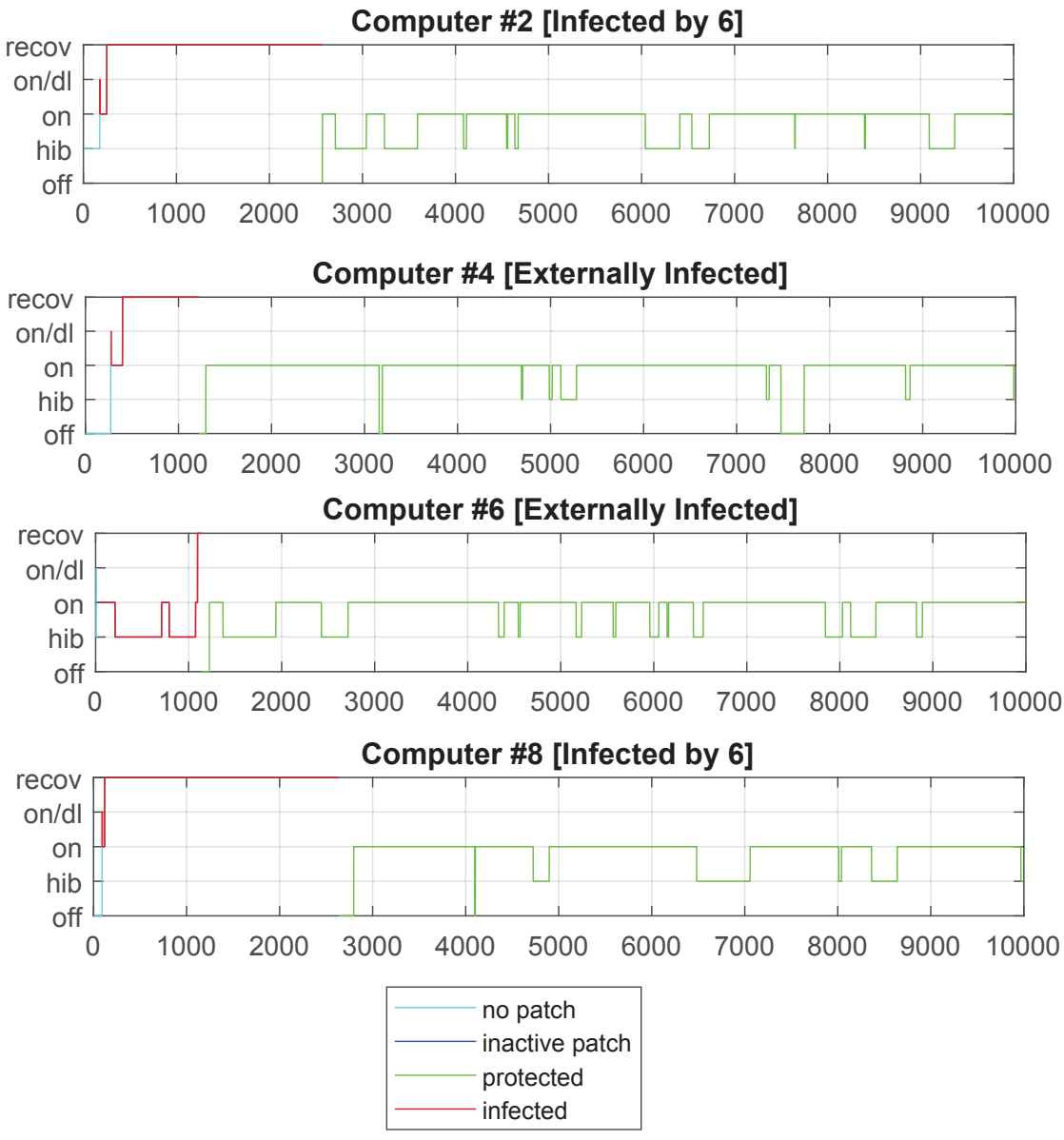

Figure 4.2: Simulation result of worm infection dynamics on a LAN consisting of 8 computers.

4.2. Performance Analysis of the Proposed Model

Since the goal of proposed model is to seek strategies to reduce the effects of worm attacks on a LAN, its performance will be analyzed in terms of the number of computers that are infected at a specific worm attack ($NF_{\%}$), while the average percentage of active computers ($W_{\%}$) is kept at a reasonable level. For this purpose, firstly, the values of the parameters which are μ_0, χ, and λ_0 are determined to have an average percentage of working computers as 70-75 % approximately. After satisfying this reasonable percentage value, the effects of infection state parameters on the average percentage of infected computers that have to be repaired are investigated via some experiments.

4.2.1. Importance of Early Discovery of Infection

One of the parameters related to infection state transitions is the rate of infection discovery (γ), which directly affects the average percentage of infected computers. If the Worm infects any computer on the LAN, it scans the others to propagate more. As long as the infected computers are not diagnosed and sent to repair, the infection will have an opportunity to spread. The simulation results obtained by varying the rate of infection discovery (γ) are presented in figure 4.3.

Results for various initial number of infected computers are represented in different colors in figure 4.3. The simulation results are obtained for a LAN consisting of 64 computers by taking the average of 100 runs. Figure 4.3 presents the effects of the number of initially infected computers on the LAN with respect to the varying values of the rate of infection discovery.

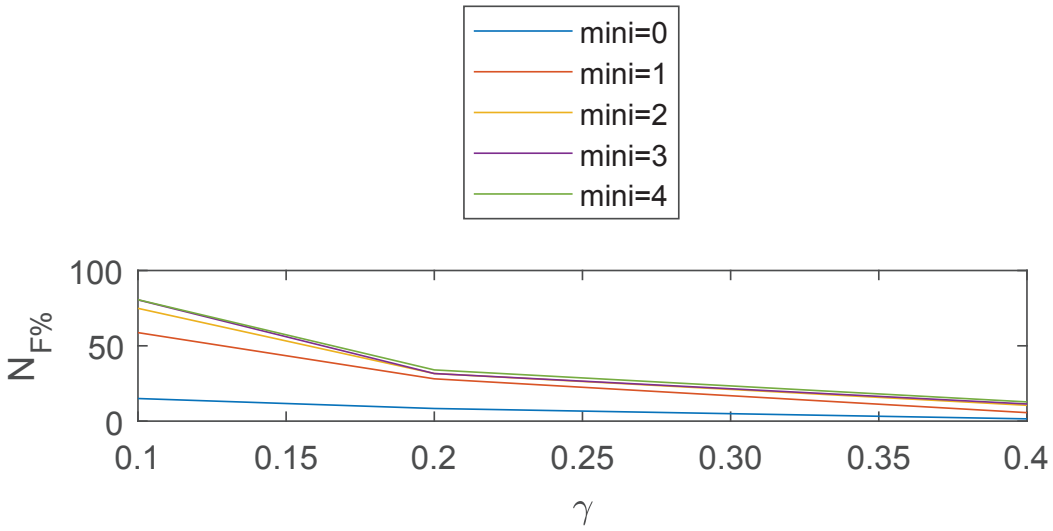

Figure 4.3: The average percentage of infected computers at a worm attack as a fraction the rate of infection discovery.

Different number of initially infected computers (m_{ini}) are taken into consideration to be more realistic because when the simulation starts there can be already infected computers on the LAN. As can be seen from the general trend on the graph, the percentage of infected computers decreases with increasing rate of infection discovery.

Fixed parameters and their values are presented in table 4.1. Actually, these values are the nominal values that are found at the beginning of the simulations. Results obtained in this simulation experiment are summarized in figure 4.4 which clearly shows the values of $N_{F\%}$ versus different γ values and different number of initially infected computers (m_{ini}). As m_{ini} increases, $N_{F\%}$ values increase as expected.

Table 4.1: List of fixed parameters to investigate the effects of γ on $N_{F\%}$ ($m_{ini} = 0$).

Parameters	**Fixed Values**
λ_0	0.007 min^{-1}
μ_0	0.0001 min^{-1}
χ	0.0001 min^{-1}
β_{ext}	0.002 min^{-1}
μ_{rb}	0.033 min^{-1}
δ_{dl}	0.5 min^{-1}
ν	0.0007 min^{-1}
β_{int}	0.7

	N_F %	$m_{ini} = 0$	$m_{ini} = 1$	$m_{ini} = 2$	$m_{ini} = 3$
Infection discovery rate	$\gamma = 0.001$ min^{-1}	15	58.75	74.88	79.12
	$\gamma = 0.002$ min^{-1}	8.75	28.08	31.51	34
	$\gamma = 0.004$ min^{-1}	1.5	5.63	11.38	12.75

Figure 4.4: Simulation results for $N_{F\%}$ versus various infection discovery rates (γ).

4.3. Importance of Security Precautions within the LAN

The spread of infection across a LAN critically depends on the probability of individual computers infecting each other. To investigate this dependence, the parameter called as β_{int}, the rate of an infected computer infecting another on the LAN, has been varied in a reasonable range to investigate its effects on $N_{F\%}$, while keeping the other parameters' values constant as shown in figure 4.5. In this simulation experiment, in order to easily observe the dependence on β_{int}, it is assumed that just one computer is

initially infected ($m_{ini} = 1$) and there is no more external infection arrival ($\beta_{ext} = 0$).

Para-meters	Fixed values
λ_o	0.007 min^{-1}
μ_o	0.0001 min^{-1}
χ	0.0001 min^{-1}
β_{ext}	0 min^{-1}
γ	0.004 min^{-1}
ν	0.0007 min^{-1}
δ_{dl}	0.5 min^{-1}
μ_{rb}	0.033 min^{-1}
m_{ini}	1

Figure 4.5: The list of fixed parameters to observe the effects of β_{int} on average percentage of repaired computers ($N_{F\%}$)

In this simulation, again, a LAN with 64 computers is considered and the results are obtained as the average of 100 runs. Figure 4.6 presents how the average percentage of infected computers depends on the internal infection rate β_{int}. This empirical dependence can be used by the LAN administrator to develop strategies for reducing $N_{F\%}$. To reduce this percentage, the infection status of the computers needs to be checked more frequently by the administrator. After detecting any infected computer on the LAN, data sharing of this computer has to be prevented by the administrator.

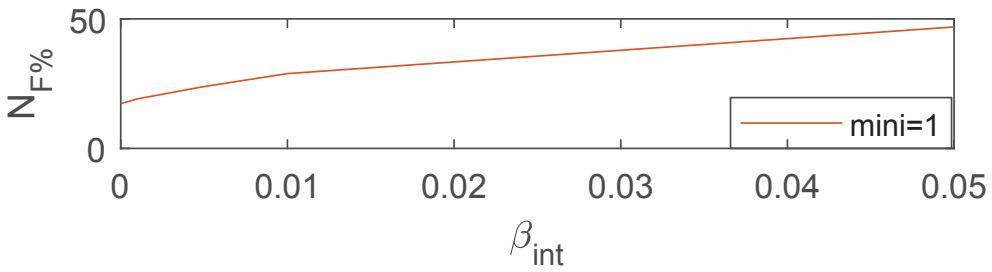

Figure 4.6: The average percentage of repaired computers ($N_{F\%}$) versus the rate of internal infection (β_{int}).

Another way is that if there is a control mechanism to check the infection status of the files that are shared on the LAN, this percentage can be reduced more. This experiment is performed to point out the importance of security precautions within the LAN.

β_{int}	N_F %
0.1 min^{-1}	22.12
0.5 min^{-1}	38.34
0.7 min^{-1}	51.01

Figure 4.7: The average percentage of repaired computers ($N_{F\%}$) versus the rate of internal infection (β_{int}).

Detailed simulation results taken from figure 4.6 can be found in figure 4.7. Therefore, the observation of this result is that if there is a regulation on the LAN while the internal computers are communicating with each other, propagation of the Worm will be harder. In this way, the values of $N_{F\%}$ will decrease a lot.

4.4. Negligence of a few

In our model, the LAN administrator automatically pushes all hosts that are in ON-state to download a new patch, as soon as it arrives at the LAN. However, the activation of the patch is at the disposal of the host. If the user, preoccupied with other tasks, neglects to reboot the computer and activate the patch, he will not only run the risk of being infected, but also endanger the others.

In order to investigate how the negligence of a few affects the security of the overall LAN, a simulation experiment has been conducted where k negligent users reboot their computers at a low rate ($\mu_{rb} = 0.01min^{-1}$), while the rest operate at the nominal value ($\mu_{rb} = 0.033min^{-1}$). For the sake of simplicity, a LAN of 8 computers is used in this experiment. Figure 4.8 presents the results of this simulation.

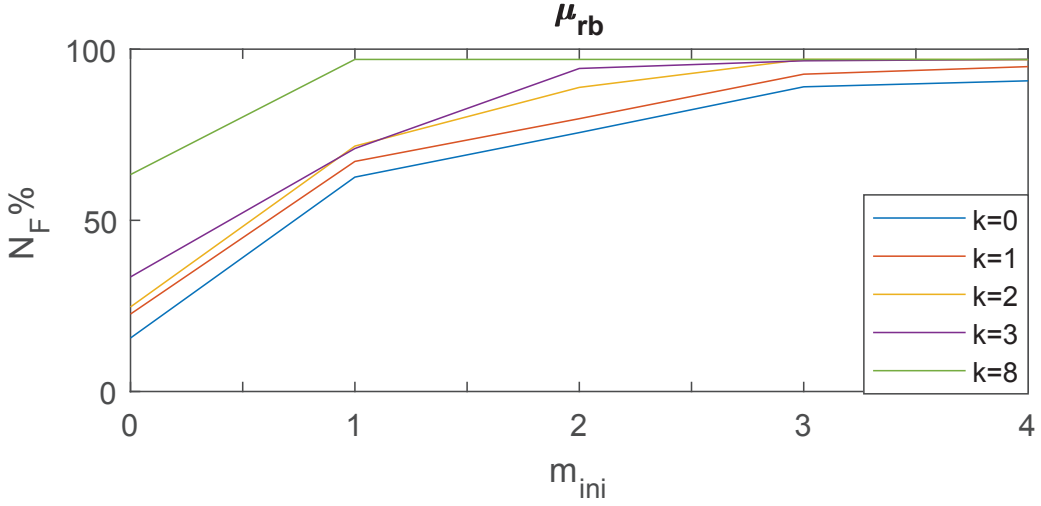

Figure 4.8: k computers with low $\mu_{rb}(0.01min^{-1})$ (number of incautious hosts). N-k computers with nominal $\mu_{rb}(0.033min^{-1})$ (number of cautious hosts).

Table 4.2: List of fixed parameters to investigate the effects of μ_{rb} on $N_{F\%}$ ($m_{ini} = 0$).

Parameters	**Fixed Values**
λ_0	0.007 min^{-1}
μ_0	0.0001 min^{-1}
χ	0.0001 min^{-1}
β_{ext}	0.002 min^{-1}
γ	0.004 min^{-1}
δ_{dl}	0.5 min^{-1}
ν	0.0007 min^{-1}
β_{int}	0.7

N_F %	$m_{ini} = 0$	$m_{ini} = 1$	$m_{ini} = 2$	$m_{ini} = 3$
k = 0	18.3	52.1	65.21	84.12
k = 1	22.2	54.3	66.7	86
k = 2	22.6	55.2	75.1	89.4
k = 8	63.1	95.5	96.2	98

Figure 4.9: k computers with low $\mu_{rb}(0.01min^{-1})$ (number of incautious hosts). N-k computers with nominal $\mu_{rb}(0.033min^{-1})$ (number of cautious hosts).

In this simulation, k shows the number of negligent users who activate the patch at a low rate ($\mu_{rb} = 0.01min^{-1}$) while the rest reboots at the nominal rate ($\mu_{rb} = 0.033min^{-1}$). Detailed simulation results depicted in figure 4.9 show that the higher the number of computers that are rebooted relatively late, the higher is the percentage

of infected computers. But it is worth noting that as long as these negligent users are few, the resulting percentage of infected computers remains relatively close to the levels obtained when all users are cautious. This is similar to the way the vaccination of the majority in a population is enough to constrain the spread of a disease even though there is an unvaccinated minority.

5. CONCLUSION

The main purpose of this thesis was to investigate the propagation of the Worm in a Local Area Network which is maintained by a central administrator who has the authorization to enforce the corresponding patch updates on each computer, but leaves their activation to the individual users.

For this purpose, a novel model is designed to account for the Worm spread dynamics starting from the instance when the patch related to the Worm has been received by the administrator. The model accounts for all state transitions in case of any worm attack. All processes of such a model are based on Markovian approach. By designing the model, our goal is to minimize the percentage of infected computers that have to be recovered and to maximize the average percentage of network workload. The presented Markovian model is useful for investigating the effects of different parameters and evaluating the feasibility of different security policies.

Simulation results show that if the administrator checks the infection status of each computer more frequently, the infection can be detected in a short time. As a result, the average percentage of repaired computers can be decreased and the whole LAN can be kept in a safer mode. To implement such a policy, the administrator can check and warn the corresponding users according to the type of symptoms of the corresponding computers. Another factor that affects the average percentage of repaired computers is the internal infection probability (β_{int}). Simulation results point out the necessity of regulations for the communication of computers while sharing files, data, etc on the same LAN. Moreover, rebooting of the computers on time can keep the LAN in a safer mode as shown in figure 4.3. Hence, the users should give priority to the activation of patches when not occupied.

The average dynamics resulting from the presented Markovian model can also be captured by a compartmental model, but it would require at least 12 different compartments and involve rather long differential equations.

As a future work of this thesis, within the LAN, which kind of security measures can be taken to reduce the possibility of getting infection needs to be investigated. Besides, the values of the internal infection rate (β_{int}) are heuristically determined. In the future, their values can be determined more realistically by the type of the security measures within the LAN. In addition to that, the working profile of the individual computers can be represented more explicitly by the turning on/off and hibernation rates. After having more realistic parameter values, security measures can be applied to a real LAN to observe more implicit effects of the parameters.

REFERENCES

1. Syed, F., "Understanding worms, their behaviour and containing them", *Project Report*, pp. i277–i285, 2009.

2. Wikipedia, *WannaCry*, 2017, `https://tr.wikipedia.org/wiki/WannaCry`, accessed at May 2018.

3. Benedictow, O. J., *The Black Death, 1346-1353: the complete history*, Boydell & Brewer, 2004.

4. Caldwell, J. C., "Demographers and the Study of Mortality", *Annals of the New York Academy of Sciences*, Vol. 954, No. 1, pp. 19–34, 2001.

5. Bernoulli, D., "Essai d'une nouvelle analyse de la mortalité causée par la petite vérole, et des avantages de l'inoculation pour la prévenir", *Histoire de l'Acad., Roy. Sci.(Paris) avec Mem*, pp. 1–45, 1760.

6. Brauer, F., "Mathematical epidemiology: Past, present, and future", *Infectious Disease Modelling*, Vol. 2, No. 2, pp. 113–127, 2017.

7. Bingham, P., N. Verlander and M. Cheal, "John Snow, William Farr and the 1849 outbreak of cholera that affected London: a reworking of the data highlights the importance of the water supply", *Public health*, Vol. 118, No. 6, pp. 387–394, 2004.

8. Weisstein, E. W., *Kermack-McKendrick Model*, 2018, `http://mathworld.wolfram.com/Kermack-McKendrickModel.html`, accessed at May 2018.

9. Macdonald, G. *et al.*, "The epidemiology and control of malaria.", *The Epidemiology and Control of Malaria.*, 1957.

10. Anderson, R. M. and R. M. May, "Population biology of infectious diseases: Part I", *Nature*, Vol. 280, No. 5721, p. 361, 1979.

11. Kienzle, D. M. and M. C. Elder, "Recent worms: a survey and trends", *Proceedings of the 2003 ACM workshop on Rapid malcode*, pp. 1–10, ACM, 2003.

12. Shuai, Z. and P. Van den Driessche, "Global dynamics of cholera models with differential infectivity", *Mathematical biosciences*, Vol. 234, No. 2, pp. 118–126, 2011.

13. Wikipedia, *Timeline of computer viruses and worms*, 2018, `https://en.wikipedia.org/wiki/Timeline_of_computer_viruses_and_worms`, accessed at May 2018.

14. Chen, T. M. and J.-M. Robert, "Worm epidemics in high-speed networks", *Computer*, Vol. 37, No. 6, pp. 48–53, 2004.

15. Zou, C. C., W. Gong and D. Towsley, "Code red worm propagation modeling and analysis", *Proceedings of the 9th ACM conference on Computer and communications security*, pp. 138–147, ACM, 2002.

16. Chen, Z., L. Gao and K. Kwiat, "Modeling the spread of active worms", *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, Vol. 3, pp. 1890–1900, IEEE, 2003.

17. IDM, *SIR and SIRS models*, 2018, `https://institutefordiseasemodeling.github.io/Documentation/general/model-sir.html#sir-with-vital-dynamics`, accessed at May 2018.

18. Guevara, P., G. Delgado, J. Audelo, J. Valdez and H. Pérez, "Basic definitions for discrete modeling of computer worms epidemics", *Ingeniería e Investigación*, Vol. 35, No. 1, pp. 79–85, 2015.

19. Li, P., M. Salour and X. Su, "A survey of internet worm detection and containment", *IEEE Communications Surveys & Tutorials*, Vol. 10, No. 1, 2008.

20. Toutonji, O. and S.-M. Yoo, "An Approach against a Computer Worm Attack", *International Journal of Communication Networks and Information Security*, Vol. 1, No. 2, p. 47, 2009.

21. Lifewire, *How to Change Windows Update Settings*, 2018, `https://www.lifewire.com/how-to-change-windows-update-settings-2625778`, accessed at May 2018.

22. Martcheva, M., *An introduction to mathematical epidemiology*, Vol. 61, Springer, 2015.

23. Andersson, H. and T. Britton, *Stochastic epidemic models and their statistical analysis*, Vol. 151, Springer Science & Business Media, 2012.

24. MathWorks, *Markov Chains*, 2018, `https://www.mathworks.com/help/stats/markov-chains.html`, accessed at May 2018.

APPENDIX A: MARKOV CHAINS

Markov processes are a special kind of stochastic processes, where the system can make transitions between different states according to certain probabilities. Markov processes are classified as memoryless because their next state depends only on their current state [24].

Markov processes are divided into two: (i) discrete-transition and (ii) continuous transition Markov processes.

(i) Discrete-transition Markov processes occur at discrete time steps with constant state transition probabilities such that $S_i(n+1)$, denoting the probability of being in state S_i, right after, the (n+1)st time step can be written as:

$$S_i(n+1) = \sum_{j=1}^{N} p_{ij} S_j(n) \tag{A.1}$$

where N is the number of all states. Given S_i, the waiting time for transition to S_j is a geometrically distributed random variable with expected value $1/p_{ij}$.
A Markov model can be represented by a state diagram as below. The states are S_1, S_2, S_3, S_4, S_5. The arrows represent the transitions between the states. The labels on the arrows show the probability of transition between the corresponding states. At each step of the model, the system generates an output, depending on its current states according to these transition probabilities. For example, in figure A.1, a state transition from S_1 to S_2 occurs with a probability of p_{12} and from S_1 to S_5 with a probability of p_{15}.

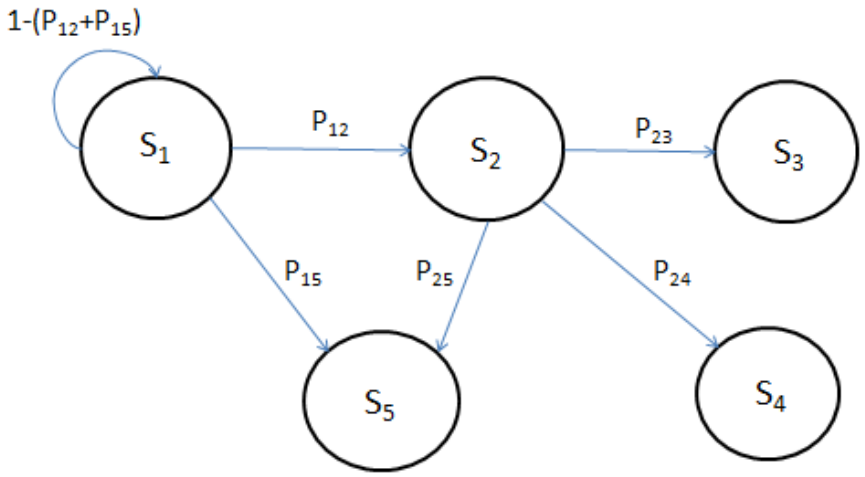

Figure A.1: State Diagram for a Discrete-transition Markov model.

(ii) Continuous transition Markov processes can occur any time and the probability of transition from S_i to S_j at any infinitesimal time interval dt is $\lambda_{ij}dt$, where λ_{ij} is the transition rate. Thus, the probability of being found in S_i at time (t+dt) can be written as:

$$S_i(t+dt) = \sum_{j=1}^{N}(\lambda_{ji}dt)S_j(t) + S_i(t)[1 - \sum_{j=1}^{N} \lambda_{ji}dt] \tag{A.2}$$

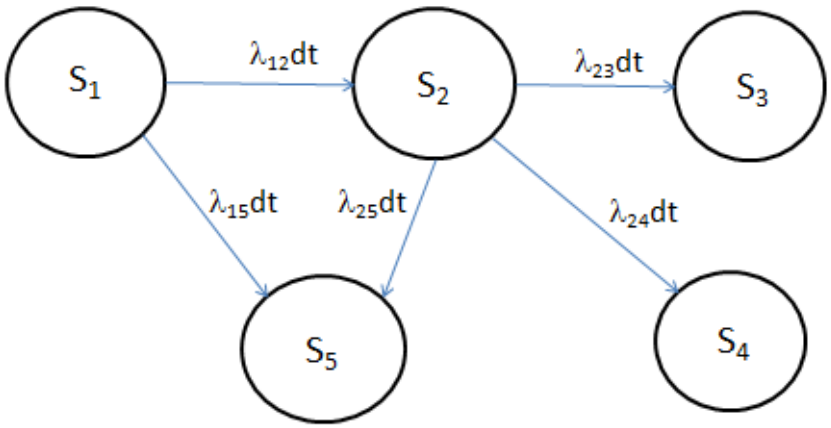

Figure A.2: State Diagram for a Continuous-transition Markov model.

Given S_i, the waiting time for a transition to S_j is an exponentially distributed random variable with expected value $1/\lambda_{ji}$.

As in figure A.1, the same states are used in figure A.2, but the state transition probabilities are represented by multiplying the state transition rates with the time interval for a continuous transition Markov process.