электронный журнал
МОЛОДЕЖНЫЙ НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК
Издатель ФГБОУ ВПО "МГТУ им. Н.Э. Баумана". Эл No. ФС77-51038.

**УДК 519.17 + 004.02**

# Анализ распространения вирусов в компьютерных сетях на основе цепи Маркова

***Емельянов А.А.,*** *студент*
*Россия, 105005, г. Москва, МГТУ им. Н. Э. Баумана,*
*кафедра «Программное обеспечение ЭВМ и информационные технологии»*

*Научный руководитель: Рудаков И.В., к.т.н., доцент*
*Россия, 105005, г. Москва, МГТУ им. Н. Э. Баумана,*
*кафедра «Программное обеспечение ЭВМ и информационные технологии»*
*irudakov@bmstu.ru*

## Введение

В современном мире невозможно представить жизнь без компьютеров. Сейчас компьютеры имеются в каждом доме, в каждом учебном заведении, в каждой организации. На компьютерах установлено различное ПО, которое помогает в работе, учебе, управляет работой различных систем. Но также существует и вредоносное ПО называемое вирусами.

Компьютерный вирус – это вид вредоносного ПО, способного создавать копии самого себя и внедряться в код других программ, распространять свои копии по разным каналам связи с целью нарушения работы ПК, удаления файлов, блокировки работы пользователей.

Компьютерные вирусы на сегодняшний день являются постоянной угрозой, представляющей опасность как для отдельных пользователей ПК, так и для предприятий. Актуальной является задача проектирования локальных сетей таким образом, чтобы сеть была максимально защищена от вирусов своей структурой. Для точной оценки защищенности сети от вирусов необходимо иметь возможность смоделировать развитие вирусной эпидемии на выбранной конфигурации сети.

Существующие модели распространения вирусов в компьютерных сетях (SI, SIR[3], AAWP[4], PSIDR[5]) не учитывают топологию сети. В этом заключается их основной недостаток. Хотя модели AAWP[4] и PSIDR[5] разработаны с учетом особенностей компьютерных вирусов. Для наиболее приближенного к реальности моделирования воспользуемся цепями Маркова. В данной статье будет задача разработать

метод распространения вредоносного программного обеспечения одного типа по компьютерным сетям с различной топологией, без защиты на основе цепей Маркова.

**Понятие цепи Маркова**

Пусть $(\Omega, F, P_0)$ – вероятностное пространство и $X = \{X_n\}_{n\geq 0}$ – определенная на $\Omega$ последовательность случайных величин, принимающих значения из E. Пусть $F_n = \sigma(X_m, m \leq n)$ и $Y_n$ – возрастающая последовательность σ-алгебр, такая, что $Y_n \supset F_n$ для любого n.

Последовательность $X = \{X_n\}_{n\geq 0}$ называется цепью Маркова по отношению к σ-алгебрам $Y_n$, если для любого n σ-алгебры $Y_n$ и $\sigma(X_m, m \geq n)$ условно независимы относительно $X_n$, т.е. для любых $A \in Y_n$ и $B \in \sigma(X_m, m \geq n)$:

$$P_0[A \cap B|X_n] = P_0[A|X_n]P_0[B|X_n]$$

Если $Y_n = F_n$, то $X_n$ является цепью Маркова. Тогда «прошлое» $\sigma(X_m, m \leq n)$ и «будущее» $\sigma(X_m, m \geq n)$ в этом определении будут играть абсолютно симметричную роль, что интуитивно понятно: если задано настоящее и прошлое независимы друг от друга.

Последовательность $X = \{X_n\}_{n\geq 0}$ случайных величин называется однородной цепью Маркова по отношению к σ-алгебрам $Y_n$ с переходной вероятностью P, если для любых целых m и n, таких, что m<n, и любой функции $f \in bE$

$$E_0[f(X_m)|Y_n] = P_{n-m}f(X_m)$$

**Анализ**

Рассмотрим виртуальную сеть, состоящую из N компьютеров. У каждого компьютера имеется свое состояние: зараженный или незараженный.

Сеть можно представить в виде графа, узлы которого являются компьютеры, а дугами являются каналы связи между ними, по которым может распространяться вредоносное программное обеспечение. Вес связи $w_{ij}$ означает вероятность перехода вируса по каналу связи между компьютерами i и j за единицу времени.

**Модель на основе цепи Маркова для всей сети**

Общее состояние сети является совокупностью состояний всех компьютеров сети, которое можно описать вектором из N элементов, где значение i-го элемента соответствует состоянию i-го компьютера: I (infected), если компьютер заражен, и S (suspected), если компьютер не заражен. Состояние сети в следующий момент времени

зависит только от текущего состояния сети, и не зависит от предыдущих. Поэтому процесс распространения вируса в сети можно представить как цепь Маркова [1,2].

Сеть перейдет из состояния $s^i$ в состояние $s^j$ при условии, если каждый компьютер в сети перейдет из состояния $s_n^i$ в состояние $s_n^j$ , где n – номер компьютера в сети.

Вероятность перехода n-го компьютера из состояния $s_n^i$ в состояние $s_n^j$ можно вычислить следующим образом. Необходимо рассмотреть четыре варианта для различных состояний компьютера на предыдущем и следующем шаге: переход из состояния S в состояние I, из S в S, I в S, и I в I.

S → I. Пусть вероятность заражения незараженного k-го компьютера из состояния сети $s^i$ равна:

$$P_{зар}(n, s^i)$$

S → S. Поскольку событие перехода компьютера в зараженное состояние и событие, при котором незараженный компьютер останется незараженным, образуют полную группу событий, то вероятность будет равна:

$$1 - P_{зар}(n, s^i)$$

I → S. Так как модель не учитывает излечения компьютера от вирусов, то переход из состояния 1 в состояние 0 невозможен, т.е. имеет нулевую вероятность.

I → I. Так как компьютер в данной ситуации уже заражен, то вероятность перехода из состояния 1 в состояние 1 равна единице.

Вероятность передачи вируса от узла k узлу n при состоянии сети $s_k^i$ можно вычислить следующим образом: если компьютер m заражен, вероятность равна $w_{kn}$; если компьютер k не заражен, то вероятность передачи вируса с него равна нулю.

$$P_{передачи}(k, n, s_k^i) = \begin{cases} w_{kn}, & s_k^i = I \\ 0, & s_k^i = S \end{cases}$$

Узел n перейдет из незараженного состояния в зараженное за единицу времени в том случае, если вирус к нему поступит хотя бы с одного другого узла. Поскольку события заражения n-го узла от различных источников являются независимыми, то вероятность заражения незараженного n-го узла будет равна:

$$P_{зар}(n, s^i) = 1 - \prod_{k-1}^{N}(1 - P_{передачи}(k, n, s_k^i)).$$

**Модель на основе цепи Маркова для отдельных узлов**

Если рассматривать в виде марковской цепи не процесс распространения вируса по всей сети, а строить отдельную марковскую цепь для каждого узла, можно значительно сократить объем вычислений.

В каждый момент времени каждый компьютер с определенной вероятностью может быть незараженным (S) либо зараженным (I). Вектор состояния в данном случае состоит из двух элементов – вероятности того, что компьютер не заражен, и вероятности того, что компьютер заражен. Сумма состояний равна 1. Следовательно в итоге получим следующую матрицу переходов:

$$P = \begin{bmatrix} 1 - P_{зар}(n) & P_{зар}(n) \\ 0 & 1 \end{bmatrix},$$

где $P_{зар}(n)$ – это вероятность заражения n-го узла, которая, как было показано выше, вычисляется по формуле:

$$P_{зар}(n) = 1 - \prod_{k=1}^{N}(1 - P_{передачи}(k, n)).$$

Передача вируса от узла k узлу n произойдет при одновременном наступлении следующих событий: если компьютер k заражен на предыдущем шаге, вероятность этого события равна $P_n^{t-1}(I)$, если вирус пройдет по связи k – n, вероятность этого события равна $w_{kn}$.

Следовательно, вероятность передачи вируса от узла k узлу n равна произведению вероятности зараженности компьютера n на предыдущем шаге, умноженной на вероятность перехода вируса по связи k – n:

$$P_{передачи}(k, n) = P_n^{t-1}(I) \cdot w_{kn},$$

где t - номер шага.

В результате подстановки получаем следующую матрицу переходов для отдельного узла:

$$P = \begin{bmatrix} \prod_{k=1}^{N}(1 - P_{передачи}(k, n)) & 1 - \prod_{k=1}^{N}(1 - P_{передачи}(k, n)) \\ 0 & 1 \end{bmatrix}.$$

Можно заметить, что матрица переходов зависит от состояния узлов сети на предыдущем шаге, – следовательно, цепь Маркова для каждого узла является неоднородной.

Для использования модели на основе цепи Маркова необходимо задать начальное распределение $g_0$– вектор вероятностей нахождения сети в том или ином состоянии в начальный момент времени. Выбирается единственное начальное состояние сети $f_0$, для которого вероятность принимается равной единице, для остальных – нулю.

Исходя из теории марковских цепей, распределение на шаге t будет равно:

$$g_t = g_{t-1}P.$$

Математическое ожидание среднего числа зараженных компьютеров на шаге q можно вычислить следующим образом.

Для каждого состояния сети $s_j$ легко определить количество зараженных компьютеров: поскольку $s_j$ представляет собой вектор, состоящий из N элементов, то количество зараженных компьютеров для состояния $s_j$ определяется как

$$N_I(s_j) = \sum_{i=1}^{N} \begin{cases} 1, s_{ij} = \mathrm{I} \\ 0, s_{ij} \neq \mathrm{I} \end{cases}.$$

Поскольку сумма всех элементов вектора состояния на шаге t всегда равно единице, то математическое ожидание количества зараженных компьютеров будет равно

$$M[N_I^t] = \sum_{j=1}^{2^N} p_j^{(t)} \cdot N_I(s_j).$$

Проводя вычисления для t = 0, ..., $t_{max}$, получим зависимость среднего количества зараженных компьютеров от номера шага t.

Использование модели для отдельных узлов значительно проще. Начальное распределение задается для каждого узла сети: $g_j^0 = \{P_j^0(S), \quad P_j^0(I)\}$, где j – номер узла. Наиболее удобный способ построения начальных распределений – выбрать в сети зараженные компьютеры, для которых вероятность нахождения в зараженном состоянии равна 1, т.е. $g_j^0 = \{0, \quad 1\}$, а для остальных, – наоборот $g_j^0 = \{1, \quad 0\}$.

Далее для каждого шага t производятся следующие действия:

для каждого j-го узла строится матрица переходов; вектор начального на предыдущем шаге умножается на полученную матрицу переходов, в результате получается вектор распределения для первого шага:

$$g_t = g_{t-1}P.$$

Имея множество векторов распределения на шаге t (равное $\{g_1^t, g_2^t, \dots, g_N^t\}$), а следовательно, зная вероятность нахождения каждого узла в зараженном состоянии в момент времени t ($P_j^t(I)$), можно вычислить математическое ожидание количества зараженных компьютеров на этом шаге:

$$M[N_I^t] = \sum_{j=1}^{N} P_j^t(I).$$

Проводя вычисления для t = 0, ..., $t_{max}$, получим зависимость среднего количества зараженных компьютеров от номера шага t.

**Описание программного комплекса**

Для моделирования распространения вирусов в компьютерных сетях на основе данных методах была разработана программа на языке C#. Опишем ее входные и выходные данные.

Входные данные:

- количество узлов в сети;
- связи между узлами сети (информационные связи, по которым могут распространяться вирусы);
- веса связей (вероятность передачи вируса с зараженного узла на незараженный за один шаг через данную связь);
- начальное состояние сети (зараженные узлы);
- метод моделирования.

Выходные данные:

- зависимость количества зараженных узлов от номера шага.

В результате можно будет сравнить время полного заражения сети в зависимости от ее топологии, а также сравнить методы моделирования.

Рассмотрим результаты выполнения программы на графе из 5 компьютеров и четырех различных структурах для получения скорости распространения вредоносного ПО. Вероятность передачи вредоносного ПО от одного компьютера к другому при наличии связи между ними равна 0.25. На первом этапе заражение присвоим компьютеру

1. Программа проводила моделирование по двум методам: марковской цепи для всей сети; марковской цепи для отдельных узлов.

Используя первую модель, строится матрица переходных вероятностей для сети из 5 компьютеров размером 32x32 (таблица 1). Из-за того, что вероятность, заданная 0.25, перехода вредоносного ПО из одного компьютера в другой мала, то вероятность того, что состояние сети за одни шаг не изменится велика. Наша модель не учитывает лечения, поэтому элементы ниже главной диагонали матрицы равны 0.

*Таблица 1*

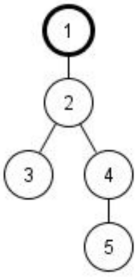
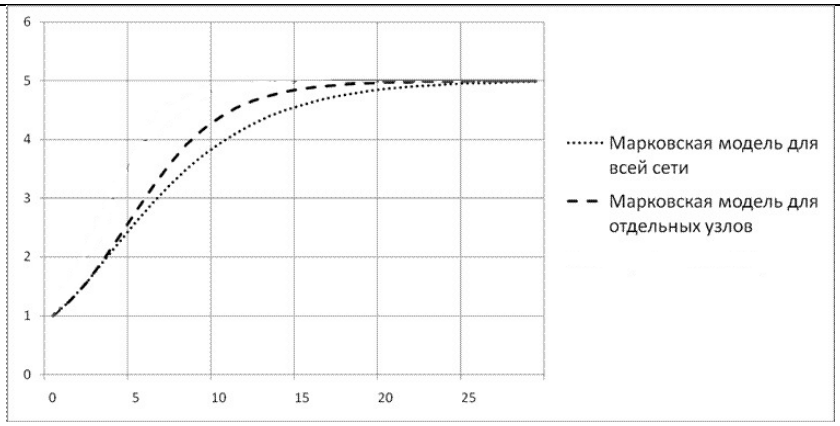
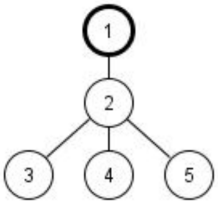
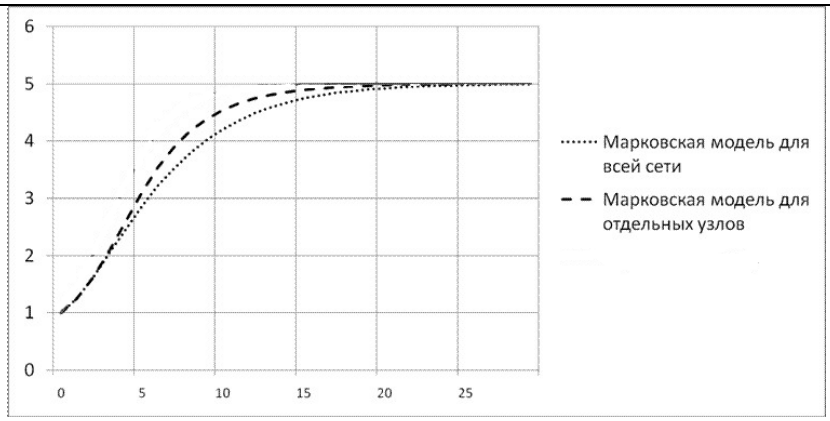
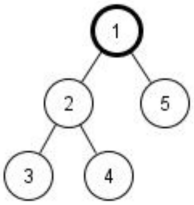
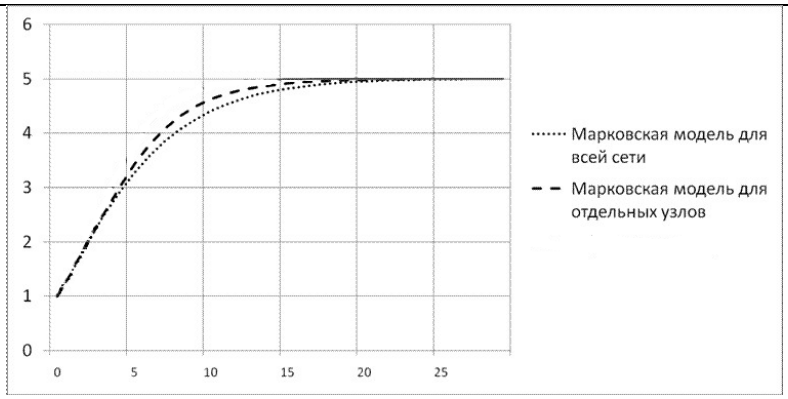
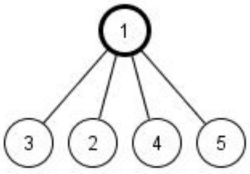
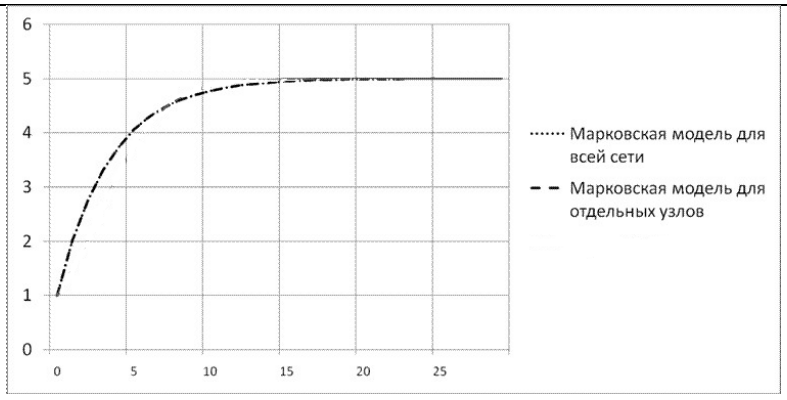
Матрица переходных состояний для сети из 5 компьютеров

| | SSSSS | SSSSI | SSSIS | SSSII | SSISS | SSISI | SSIIS | SSIII | SISSS | SISSI | SISIS | SISII | SIISS | SIISI | SIIIS | SIIII | ISSSS | ISSSI | ISSIS | ISSII | ISISS | ISISI | ISIIS | ISIII | IISSS | IISSI | IISIS | IISII | IIISS | IIISI | IIIIS | IIIII |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SSSSS | 1.0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SSSSI | 0 | 0.7 | 0 | 0.2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SSSIS | 0 | 0 | 0.4 | 0.1 | 0 | 0 | 0.1 | 0 | 0 | 0 | 0.1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SSSII | 0 | 0 | 0 | 0.5 | 0 | 0 | 0 | 0.1 | 0 | 0 | 0 | 0.1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SSISS | 0 | 0 | 0 | 0 | 0.7 | 0 | 0.2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SSISI | 0 | 0 | 0 | 0 | 0 | 0.5 | 0 | 0.4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SSIIS | 0 | 0 | 0 | 0 | 0 | 0 | 0.5 | 0.1 | 0 | 0 | 0 | 0 | 0 | 0 | 0.1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SSIII | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SISSS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.5 | 0 | 0.1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SISSI | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.4 | 0 | 0.3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.1 | 0 | 0.1 | 0 | 0 | 0 | 0 |
| SISIS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.4 | 0.1 | 0 | 0 | 0.1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.1 | 0 | 0 | 0 | 0 | 0 |
| SISII | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.5 | 0 | 0 | 0 | 0.1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.1 | 0 | 0 | 0 | 0 |
| SIISS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.4 | 0 | 0.3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.1 | 0 | 0.1 | 0 |
| SIISI | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.3 | 0 | 0.4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.1 | 0 | 0.1 |
| SIIIS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.5 | 0.1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.1 | 0 |
| SIIII | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.2 |
| ISSSS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ISSSI | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.5 | 0 | 0.1 | 0 | 0 | 0 | 0 | 0 | 0.1 | 0 | 0 | 0 | 0 | 0 | 0 |
| ISSIS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.3 | 0.1 | 0 | 0 | 0.1 | 0 | 0 | 0 | 0.2 | 0 | 0 | 0 | 0 | 0 |
| ISSII | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.4 | 0 | 0 | 0 | 0.1 | 0 | 0 | 0 | 0.3 | 0 | 0 | 0 | 0.1 |
| ISISS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.5 | 0 | 0.1 | 0 | 0 | 0 | 0 | 0 | 0.1 | 0 | 0 | 0 |
| ISISI | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.4 | 0 | 0.3 | 0 | 0 | 0 | 0 | 0 | 0.1 | 0 | 0.1 |
| ISIIS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.4 | 0.1 | 0 | 0 | 0 | 0 | 0 | 0 | 0.3 | 0.1 |
| ISIII | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.4 |
| IISSS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.7 | 0 | 0.2 | 0 | 0 | 0 | 0 | 0 |
| IISSI | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.5 | 0 | 0.4 | 0 | 0 | 0 | 0 |
| IISIS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.5 | 0.1 | 0 | 0 | 0.1 | 0 |
| IISII | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.7 | 0 | 0 | 0 | 0.2 |
| IIISS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.5 | 0 | 0.4 | 0 |
| IIISI | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.4 | 0 | 0.5 |
| IIIIS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.7 | 0.2 |
| IIIII | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1.0 |

Результатом моделирования будет зависимость среднего количества зараженных компьютеров на каждом шаге. В таблице 2 показано 4 различных топологий сети и их графики зависимостей скорости распространения вредоносного ПО от топологии.

*Таблица 2*

Результаты моделирования 4 различных топологий сети из 5 компьютеров

| Топология сети | Зависимость количества зараженных компьютеров от номера шага |
|---|---|
| №1  |  |
| №2  |  |
| №3  |  |
| №4  |  |

На полученных графиках, на основе моделей цепи Маркова, четко видно, как изменяются зависимости от топологии сети.

В первом случае количество зараженных компьютеров увеличивается медленнее всего, чем в остальных случаях. Это из-за того, что в данной топологии, чтобы заразить компьютер 5, вредоносное ПО должно сначала заразить компьютеры 2 и 4.

Во втором случае заражение происходит немного быстрее из-за того, что компьютер 5 был присоединен ближе к зараженному компьютеру 1.

В третьем случае заражение происходит еще быстрее, так как 5 компьютер был связан непосредственно с зараженным компьютером 1.

В четвертом случае каждый компьютер связан непосредственно с зараженным компьютером 1, отчего происходит самое быстрое распространение вредоносного ПО. Поэтому данная архитектура сети является самой небезопасной с точки зрения вирусной безопасности топологии сети.

Достоинство данных методов в том, что модели оперируют однозначными состояниями всей сети. То есть, на основании полученных результатах можно сделать вывод, какие компьютеры в сети будут заражены полностью, а какие частично.

Используя однородную цепь Маркова, получаем возможность активно пользоваться математическим аппаратом теории марковских цепей. Для того, чтобы рассчитать вероятности заражения компьютеров из любого начального состояния нужно возвести матрицу в определенную степень и начальное распределение умножить на полученную матрицу. Построенную матрицу можно не изменять для различных начальных распределений, но если будет изменение топологии сети, то построение матрицы переходов нужно будет провести заново.

Основным недостатком марковской модели для всей цепи является очень большой объем вычислений. Для компьютерной сети, состоящей из N компьютеров, существует $2^N$ разных состояний сети. Для данной модели необходимо строить переходную матрицу, размер которой будет $2^{2N}$ элементов. Так как при вычислении вероятности перехода для каждого элемента матрицы происходит обход всех компьютеров сети, для каждого из которых происходит также обход всех компьютеров сети, то сложность алгоритма построения матрицы переходных вероятностей составляет $O(N^2 2^{2N})$.

Если данную модель попытаться расширить, например, перейти от модели с двумя состояниями к модели с тремя состояниями (модель, которая учитывает лечение – SIR), то сложность алгоритма возрастет до $O(N^2 3^{2N})$.

По сравнению с марковской моделью для всей цепи, марковская модель для отдельных узлов имеет гораздо меньший объем вычислений и это ее достоинство. Так как для каждого компьютера размер матрицы переходов составляет 2x2 и на каждом шаге для каждого компьютера нужно перебирать всех соседних компьютеров, то сложность алгоритма составляет $O(N^2T)$, где N – количество компьютеров в сети; T – количество шагов моделирования.

**Вывод**

Разработанные методы позволяют рассчитывать скорость распространения вредоносного ПО в компьютерных сетях с различной структурой. Был описан механизм их использования для получения информации зависимостей скорости распространения вирусов в сети от топологии сети. После моделирования был проведен анализ и сравнение моделей на основе цепи Маркова для всей цепи и для отдельных узлов, выявлены их достоинства и недостатки. Полученные модели могут помочь оценить защищенность от компьютерных вирусов сетей с разной архитектурой. Также на самом раннем этапе создания сети данные модели позволят помочь найти наиболее безопасные структуры сетей от распространения вредоносного ПО.

**Список литературы**


1. Рудаков И.В. Представление формальной макромодели функционального блока сложной дискретной структуры в виде логической сети // Инженерный вестник. МГТУ им. Н. Э. Баумана. Электрон. журн. 2014. № 10. Режим доступа: http://engbul.bmstu.ru/doc/730829.html (дата обращения 02.04.2015).
2. Рудаков И.В., Шляева А.В. Моделирование входных потоков данных для стохастических моделей дискретных систем // Вестник МГТУ им. Н. Э. Баумана. Сер. Приборостроение. 2008. № 2. С. 65-72.
3. Jeffrey O. Kephart, Steve R. White. Directed-Graph Epidemiological Models of Computer Viruses // IEEE Symposium on Security and Privacy. 1991. P. 343-359. DOI: 10.1109/RISP.1991.130801.
4. Zesheng Chen, Lixin Gao, and Kevin Kwait. Modeling the spread of active worms. Available at: http://infocom2003.ieee-infocom.org/papers/46_03.PDF, accessed 20.03.2015.
5. Williamson M.M., Leveille J. An epidemiological model of virus spread and cleanup. Available at: http://www.hpl.hp.com/techreports/2003/HPL-2003-39.pdf, accessed 20.03.2015.