

Некоммерческое акционерное общество  
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ»

Кафедра «Телекоммуникационные сети и системы»

Специальность: 6М071900 «Радиотехника, электроника и телекоммуникации»

ДОПУЩЕН К ЗАЩИТЕ

Зав. кафедрой

к.т.н., профессор Байкенов А.С.

(ученая степень, звание, ФИО)

\_\_\_\_\_  
(подпись)

« \_\_\_\_\_ » \_\_\_\_\_ 2019 г.

**МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ**  
**пояснительная записка**

на тему: «Исследование вопросов идентификации, прогнозирования трафика с использованием технологий AI»

Магистрант: Амангалиева Ж.А. \_\_\_\_\_ группа МРЭТн-17-2  
(Ф.И.О.) \_\_\_\_\_ (подпись)

Руководитель: к.т.н., профессор \_\_\_\_\_ Байкенов А.С.  
(ученая степень, звание) \_\_\_\_\_ (подпись) (Ф.И.О.)

Рецензент: к.т.н., доцент \_\_\_\_\_ Халижанов В.С.  
(ученая степень, звание) \_\_\_\_\_ (подпись) (Ф.И.О.)

Консультант по ВТ к.т.н., профессор \_\_\_\_\_ Байкенов А.С.  
(ученая степень, звание) \_\_\_\_\_ (подпись) (Ф.И.О.)

Нормоконтроль: ст. пр-16 \_\_\_\_\_ Демурова Т.Д.  
(ученая степень, звание) \_\_\_\_\_ (подпись) (Ф.И.О.)

Алматы 2019

Некоммерческое акционерное общество  
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ»

Институт Космической Инженерии и Телекоммуникаций

Специальность: 6M071900 «Радиотехника, электроника и телекоммуникации»

Кафедра: «Телекоммуникационные сети и системы»

**ЗАДАНИЕ**

на выполнение магистерской диссертации

Магистранту Амангалиевой Ж.А.

(фамилия, имя, отчество)

Тема диссертации «Исследование вопросов идентификации, прогнозирования трафика с использованием технологий AI»

утверждена Ученым советом университета №     от «     »    

Срок сдачи законченной диссертации «     »    

Цель исследования состоит в изучении вопросов идентификации, прогнозирования трафика с использованием технологий искусственного интеллекта, а также в построении имитационной модели.

Перечень подлежащих разработке в магистерской диссертации вопросов или краткое содержание магистерской диссертации:

1. Исследование характеристик интернет-трафика.
2. Обзор и анализ существующих моделей для прогнозирования интернет трафика, их преимущества и недостатки.
3. Создание модели на основе искусственного интеллекта.

Перечень графического материала (с точным указанием обязательных чертежей)

- Рисунок 3.4 - Топология маршрутизации через перерезы облучивающие
- Рисунок 3.5 - Топология маршрутизации после перереза облучивающие
- Рисунок 3.6 - Зависимость скорости от поперечной пропускной способности
- Рисунок 3.7 - Зависимость скорости от задержки
- Рисунок 3.8 - Зависимость скорости от энергии
- Рисунок 3.9 - Зависимость скорости от количества добавки

Выпуск 3.10 - Зависимость потребления в компьютерной сети от топологии  
 Выпуск 3.11 - Зависимость потребления от загрузки  
 Выпуск 3.12 - Зависимость потребления от потребляемой энергии  
 Выпуск 3.13 - Зависимость потребления от потребляющей доставки

Рекомендуемая основная литература

1. Melinda Barabas, Georgeta Boanca, Andrei B. Rus, Virgil Dobrota, Jordi Domingo-Pascual, Evaluation of Network Traffic prediction based on Neural Network with Multi-task Learning and Multiresolution Decomposition, 2011  
 2. Network Simulator, <http://www.isi.edu/nsnam/ns>

ГРАФИК  
 подготовки магистерской диссертации

| Наименование разделов, перечень разрабатываемых вопросов             | Сроки представления научному руководителю | Примечание |
|--|---|------------|
| 1. Информационный обзор согласно теме                                | 13.11.2018                                |            |
| 2. Анализ существующих моделей по прогнозированию интернет-трафика   | 24.12.2018                                |            |
| 3. Сравнение имитационной модели на основе искусственного интеллекта | 8.04.2019                                 |            |
|  |   |            |
|  |   |            |

Дата выдачи задания \_\_\_\_\_

Заведующий кафедрой \_\_\_\_\_ (подпись) (Байкенов А.С.) (Ф.И.О.)

Руководитель диссертации \_\_\_\_\_ (подпись) (Байкенов А.С.) (Ф.И.О.)

Задание принял к исполнению магистрант \_\_\_\_\_ (подпись) (Амангалиева Ж.А.) (Ф.И.О.)

## **Аңдатпа**

Бұл диссертациялық жұмыс жасанды интеллект негізіндегі интернет-трафикті болжау тақырыбына арналған.

Жұмыста қолданыстағы модельдердің егжей-тегжейлі сипаттамасы және таңдалған модельге негіздеме қолданылады.

Интернет-трафикті болжау үшін жасанды интеллект негізінде модель жасаланды. Болжау міндеттерін орындау үшін NS-2 модельдеу құралы пайдаланылды.

Жұмыста 13 формула, 38 сурет, 6 кесте және 30 кеңсе қолданылған.

## **Аннотация**

Данная диссертационная работа посвящена теме прогнозирования Интернет-трафика на основе искусственного интеллекта.

В работе представлено подробное описание существующих моделей и обоснование выбранной модели.

Создана модель на основе искусственного интеллекта, прогнозирующая данные интернет-трафика. Для выполнения задачи прогнозирования был использован инструмент моделирования NS-2.

В работе использовано 13 формул, 38 рисунка, 6 таблиц и 30 источника.

## **Annotation**

This dissertation is devoted to the topic of predicting Internet traffic based on artificial intelligence.

The work uses a detailed description of the existing models and the rationale for the selected model.

Created a model based on artificial intelligence, forecasting data of Internet traffic. To perform the tasks of forecasting, the NS-2 modeling tool was used.

The work uses 13 formulas, 38 figures, 6 tables and 30 sources.

## Содержание

|   |    |
|---|----|
| Введение.....   | 6  |
| 1 Основные характеристики интернет-трафика. Существующие модели по прогнозированию интернет-трафика ..... | 9  |
| 1.1 Характеристики интернет-трафика.....  | 9  |
| 1.2 Моделирование и прогнозирование трафика.....  | 10 |
| 2 Анализ существующих моделей. Обоснование выбора модели для прогнозирования интернет-трафика.....        | 11 |
| 2.1 Методы моделирования.....   | 14 |
| 3 Предлагаемая методология прогнозирования на основе искусственных нейронных сетей.....                   | 52 |
| 3.1 Решение для высокой мобильности сети .....  | 53 |
| 3.2 Этап планирования задач.....  | 56 |
| 3.3 Этап оценки пропускной способности.....   | 58 |
| 3.4 Имитационная модель и параметры .....   | 59 |
| Заключение .....  | 64 |
| Список литературы .....   | 65 |

## Введение

Прогнозирование сетевого трафика играет очень важную роль во многих областях связи, таких как управление перегрузкой сети, адаптивные приложения, управление сетью и моделирование трафика. Характеристика трафика и моделирование необходимы для эффективного функционирования сети.

Хорошая модель по прогнозированию трафика должна обладать способностью улавливать важные характеристики трафика. В течение последних 10 лет изучение поведения сетевого трафика набирало обороты в трудах различных исследователей, которые характеризовали сетевой трафик как дальнедействующий, самоподобный и демонстрирующий распределения с тяжелыми хвостами.

Прогнозирование трафика является одним из основных научных интересов для многих сетевых инженеров. Чтобы точно спрогнозировать трафик, требуется хорошая модель, которая может представлять врожденные характеристики трафика. Благодаря хорошей модели трафика и точному методу прогнозирования, проектирование трафика можно сделать более эффективным для динамического распределения полосы пропускания, а также для инструментов обнаружения аномалий. Основываясь на методологии прогнозирования движения трафика, сетевые инженеры могут предусмотреть средства проектирования трафика, которые могут автоматически адаптироваться к любым будущим неожиданным условиям. Алгоритм долгосрочного прогноза, безусловно, может играть очень важную роль в планировании сети и обеспечении пропускной способности.

*Актуальность:* Прогноз динамики трафика является обязательным для интеллектуального контроля перегрузки и управления сетью. Высокоточное прогнозирование трафика помогает максимально использовать полосу пропускания и способствует к экономическим выгодам провайдеров.

*Цель:* разработка модели, прогнозирующей интернет-трафик на основе искусственного интеллекта.

*Объект исследования:* трафик сети казахстанского провайдера.

*Научная новизна:* разработка модели краткосрочного прогнозирования интернет-трафика на базе искусственного интеллекта.

# **1 Основные характеристики интернет-трафика. Существующие модели по прогнозированию интернет-трафика**

## **1.1 Характеристики интернет-трафика**

Однако при первоначальном изучении голосового трафика, а затем и Интернет-трафика данных в первые годы модель Маркова была необходимой парадигмой. Но после новаторских исследований Леланда, Такку, Виллингера и Уилсона, зависимость от дальнего расстояния и самоподобие считались неотъемлемыми характеристиками интернет-трафика. Это привело к нескольким исследованиям сетевого трафика, таких как моделирование, анализ производительности сети и т. д. Было обнаружено, что времена между пакетами имеют незначительное распределение, то есть скорее распределение с сильным хвостом чем экспоненциальное. Также было обнаружено, что совокупный трафик следует корреляционной схеме на больших временных масштабах, что привело к характеристике зависимости на большие расстояния.

Самоподобие и фракталы - это явления, в которых определенное свойство объекта не изменяется при масштабировании во времени и пространстве. Например, в случае фракталов, объект является геометрически похожим во всех пространственных масштабах, а в случае временных рядов он является статистически похожим во всех диапазонах временных масштабов. Некоторые из самоподобных процессов, таких как интернет-трафик, демонстрируют дальнедействующая зависимость. ДЗ - это состояние, при котором скорость затухания статистической зависимости намного медленнее, чем экспоненциальный спад. В процессах ДЗ функция автокорреляции затухает медленно. Одним из основных критериев использования свойства Маркова является его простота и аналитическая управляемость. Очень часто предполагается, что процессы взаимодействия и обслуживания связаны экспоненциальным распределением. Тем не менее, с прогрессом в области сбора данных и аналитических процессов предположение о теории очередей в тонком хвосте оказалось неуместным. Распределение тяжелого хвоста было связано с огромной передачей файлов.

ДЗ и самоподобное поведение сетевого трафика лишают законной силы традиционное пуассоновское предположение о прибытии пакетов. Хотя было обнаружено, что интернет-трафик характеризуется самоподобием, дальнедействующей зависимостью и явлениями тяжелого хвоста, в более позднем исследовании точка зрения Пуассона была вновь рассмотрена. Обнаружено, что более поздние данные о трафика следуют образцу пуассоновского процесса с точностью до секунды. Интересно отметить, что траекторное поведение можно охарактеризовать как зависящий от времени Пуассон, проявляющий дальние зависимости на больших временных масштабах. Также было обнаружено, что Интернет-трафик демонстрирует сложные характеристики масштабирования и мультифрактальности, что вызвано задержкой приема-передачи (ЗПП).

## 1.2 Моделирование и прогнозирование трафика

Моделирование трафика и его анализ играют важную роль в определении производительности сети. Модель, которая может точно интерпретировать важные характеристики трафика движения, необходима для эффективного анализа и моделирования. Это, в свою очередь, приводит к лучшему знанию динамики сети, что существенно облегчает проектирование сети и контроль потерь пропускной способности. Моделирование трафика возникло в результате изучения телефонной сети с пуассоновским предположением о процессе прибытия трафиков. Тем не менее, с появлением современных технологий, сетевой трафик эволюционировал, становясь все более сложным и взрывным, чем предыдущий голосовой трафик. Это привело к появлению нескольких сложных стохастических моделей. Но для точного прогнозирования трафика существует необходимость в моделях трафика, которые могут фиксировать характеристики.

Сетевое проектирование и управление во многом полагаются на подходящую модель для измерений трафика. В большинстве предыдущих исследований, проводившихся более двух десятилетий, наиболее вероятным выбором модели является случайный процесс. Стохастические процессы, которые, как правило, используются при моделировании, - это процессы с меньшим числом параметров. И в большинстве случаев параметры, снабженные измеренной статистикой фактических данных о трафике, имеют тенденцию создавать модели, которые имеют схожие статистические свойства с фактическими данными о трафике. Модель такого типа сможет обеспечить методику прогнозирования, сопоставимую с фактическим трафиком. Таким образом, поведение реального трафика может быть предсказано с использованием случайных процессов. В идеале такие процессы должны быть в состоянии точно представлять статистические свойства реального трафика, что не всегда возможно из-за нескольких проблем сложности. Но одна из главных целей - убедиться, что статистика первого и второго порядка соответствует статистике фактического трафика.

Моделирование и прогнозирование трафика также играют важную роль в достижении оптимального распределения ресурсов путем надлежащего выделения полосы пропускания и одновременного поддержания максимального использования сети. Таким образом, стоит иметь хорошую модель для проектирования будущей емкости сети. Динамический характер сетевого трафика влияет на необходимость динамического распределения полосы пропускания. Динамическое распределение может быть достигнуто только при наличии точной модели. Существует проблема не только недооценки, но и переоценки. Прогнозирование приводит к ограничению пропускной способности и, следовательно, к потере информации при превышении прогнозирования приводит к потере пропускной способности. Таким образом, точное моделирование и прогнозирование трафика



необходимы для достижения лучшего качества обслуживания и для лучшего будущего сетевого проектирования.

Модели временных рядов, как линейные, так и нелинейные, широко использовались для моделирования трафика сети многими исследователями. Некоторые из популярных моделей временных рядов: Авторегрессивное скользящее среднее (АСС), Авто регрессивное интегрированное скользящее среднее (АИСС), Авто регрессивная условная гетероскедастичность (АУГ) и т. д. Это не ограничивает использование альтернативных моделей, таких как нечеткая логика или нейронный сетевой подход. У каждого были свои недостатки, а также преимущества с точки зрения точности и сложности. Учитывая постоянно меняющуюся природу интернет-трафика, любой вид моделирования и прогнозирования является лишь приблизительным, а не точным. Поэтому разработка модели и алгоритма прогнозирования, который может наилучшим образом представить трафик в Интернете, должна стать основной целью сетевых инженеров.

## **2 Анализ существующих моделей. Обоснование выбора модели для прогнозирования интернет-трафика**

Принимая во внимание недостатки и преимущества предыдущих моделей, использовавшихся до сих пор при моделировании межсетевого трафика, мы рассмотрели две основные проблемы в качестве критериев моделирования и прогнозирования. Первый - это основные характеристики интернет-трафика, а второй - эффективность прогноза. Принимая во внимание взрывную природу интернет-трафика, мы выбрали нелинейную модель временных рядов - модель обобщенного авторегрессионного условного гетероскедастика (ОАРУГ). Прелесть этой модели заключается в ее условной дисперсии, где дисперсия зависит от прошлых дисперсий. Используя это свойство, мы включили взрывную природу интернет-трафика.

Мы использовали реальные данные интернет-трафика для моделирования. Проверка пригодности, график Q-Q (t-распределенный) и график автокорреляционной функции ясно показывают, что наша модель очень хорошо соответствовала данным. Алгоритм прогнозирования разработан на основе нашей модели, и мы использовали рекурсивное простое одношаговое прогнозирование для прогнозирования трафика. Сравнение фактического и прогнозируемого трафика выполняется для расчета ошибки прогноза. Ошибка прогноза оказалась минимальной. Мы также провели сравнительное исследование с другими моделями, чтобы оценить производительность нашей модели и алгоритм ее прогнозирования. Во всех случаях наша модель оказалась лучше, чем другие модели. Когда речь идет о сложности модели, наша модель представляет собой очень простую модель с простым методом прогнозирования.

Первым шагом к созданию хорошего алгоритма прогнозирования для оптимизации размещения полосы пропускания с целью поддержания

использования сети является точная модель сетевого трафика. Основная цель моделирования интернет-трафика состоит в том, чтобы определить модель, которая может наилучшим образом представить поведение трафика, и включить эту модель для разработки алгоритма, позволяющего эффективно распределять пропускную способность динамически, несмотря на то, что интернет-трафик является сложным по своей природе. Мотивация для этого исследования основана на предыдущих работах, в которых исследователи использовали несколько моделей, основанных на характеристиках данных трафика, а также принимая во внимание динамическое распределение полосы пропускания. В следующих разделах мы рассмотрим некоторые из проделанных работ, которые мотивировали это исследование.

В исследованиях, проведенных в течение последнего десятилетия, различные аспекты поведения сети показали достаточные свидетельства долгосрочной зависимости (ДЗ) и распределения с тяжелыми хвостами. Недавние исследования показали, что фактический сетевой трафик самоподобен или фрактален по своей природе, то есть взрывается в широком диапазоне временных масштабов. Общей характеристикой самоподобных явлений является то, что их пространственно-временная динамика определяется степенным распределением и гиперболически затухающими автокорреляциями. Напротив, фрактальное явление опирается на высокопараметризованные многоуровневые иерархии традиционных моделей, которые в свою очередь характеризуются функциями распределения и автокорреляции, которые затухают экспоненциально быстро. С другой стороны, зависимость от дальнего расстояния измеряет память процесса, а интуитивно отдаленные события во времени демонстрируют корреляцию.

Понимание характеристик интернет-трафика является непростой темой исследований уже более десяти лет. Глубокое знание основной динамики интернет-трафика играет важную роль для улучшения качества обслуживания. Процессы Пуассона использовались для моделирования прибытий в сеть из-за ее аналитической простоты, однако общее предположение о взаимных поступлениях пакетов как экспоненциально распределенных больше не действует<sup>4</sup>. Леланд и др. [5] предложили автомодельные процессы Ethernet-трафика в локальной сети, и эти процессы отличаются от процесса Пуассона в отношении его теоретических свойств. Исследования, проведенные на более ранних данных Ethernet, показали, что трафик сети обладает свойствами, подобными самоподобным процессам второго порядка, альтернативно определяемым стационарными процессами в широком смысле. Эти результаты привели к многочисленным исследованиям импульсного характера отдельных ПТС, которые являются составной частью совокупного трафика. В их анализе было обнаружено, что трафик демонстрирует корреляцию в различных временных масштабах, которые варьируются от секунд до часов. Большой объем работы был проделан, чтобы объяснить самоподобную природу интернет-трафика. Их гипотеза и результаты были подтверждены обширным исследованием нескольких

измерений Ethernet. Поведение интернет-трафика склоняется к существованию ДЗ, самоподобия и распределений с тяжелыми хвостами. Сообщалось, что присутствие ДЗ в трафике может быть распределением файлов, передаваемых в сети в [7] с тяжелыми хвостами.

В более ранние годы использование системы массового обслуживания для моделирования трафика было оправдано из-за меньшего объема трафика. Тем не менее, с постоянно растущим использованием Интернета и необходимостью повышения качества обслуживания привели к поиску моделей, которые могут работать даже для будущего Интернета. В [8] был предложен процесс всплеска Пуассона Парето (PPBP), и было обнаружено, что он точно моделирует интернет-трафик. PPBP точно предсказал производительность очереди агрегированной трассировки выборки. Эта модель была предложена с учетом постоянно растущего объема хоста, использующего Интернет. Процесс PPBP - это модель, тесно связанная с такими моделями, как  $M/G/\infty$ , и иногда упоминаемая как процесс  $M$ /Парето. Процесс PPBP был основан на распределении тяжелых пакетов с большими хвостами, особенно для нескольких пакетов, которые происходят одновременно.

В [4] авторы показали, что процессы Пуассона были действительны только при моделировании управляющих соединений Teletype network (TELNET) и протокола передачи файлов (FTP), и им не удалось точно смоделировать процессы поступления Wide Area Network (WAN). Было установлено, что процессы прибытия пакетов WAN более точны при моделировании с помощью автомодельных процессов. Одной из причин является тот факт, что пуассоновский процесс недооценил как разрывность, так и изменчивость интернет-трафика. В то же время крупномасштабные корреляции характеризовали следы WAN. Истоки самоподобия в интернет-трафике были в основном связаны с распределением файлов по размерам.

Несмотря на неопровержимые доказательства присутствия ДЗ в интернет-трафике, некоторые результаты показывают, что модели Пуассона все еще могут быть применимыми, так как увеличивается число источников в быстрых магистральных каналах, которые несут огромное количество различных потоков, что приводит к большим объемам трафика мультиплексирования. В [2] авторы обсуждали возможность процесса Пуассона, учитывая тот факт, что объем подключенного к Интернету хоста значительно увеличился за последнее десятилетие с момента сбора исходных данных. Таким образом, допущение Пуассона было пересмотрено с использованием новых измерений трафика, а также некоторых исторических следов трасс магистральных каналов ISP уровня 1. Исследование показало, что в подсекундных временных масштабах текущий трафик сети может быть смоделирован как процесс Пуассона. Но они также сообщили, что в мультисекундном масштабе времени сеть вела себя как кусочно-линейный нестационарный трафик с наличием ДЗ. Такое поведение сообщалось как зависящий от времени процесс Пуассона сетевого трафика, рассматриваемый

в разных временных масштабах. Изучение и анализ сосуществования распределений Пуассона и ДЗ в сетевом трафике будет ключом к моделированию интернет-трафика, что, в свою очередь, будет способствовать созданию схем управления трафиком, обеспечивающих желаемое QoS как можно более эффективным и экономически эффективным.

## **2.1 Методы моделирования**

### **2.1.1 Модели временных рядов**

Модели временных рядов использовались для моделирования интернет-трафика во многих предыдущих работах. Модель линейного временного ряда Модель авторегрессионного скользящего среднего (ARMA) была использована в качестве модели прогнозирования сетевого трафика. В [10] ARMA использовался для моделирования последовательности данных трафика из трафика Ethernet, двух колец FDDI кампуса, точек входа / выхода NSFNET и т. Д. Модель ARMA, использованная автором, была подобрана для различных данных. Однако оказалось, что вариации процесса ARMA негауссовы. В [11] было обнаружено, что дробные процессы AutoRegressive Integrated Moving Average (f-ARIMA) фиксируют ДЗ и характеристики ближней зависимости (SRD) трафика. Сезонный ARIMA был использован для размещения данных беспроводного трафика в [12].

Трафик данных показал значительно большую изменчивость, чем процессы Пуассона. Было показано, что трафик данных обладает временной корреляцией, которая сохраняется в масштабах времени, которые варьируются от миллисекунд до сотен секунд. Нестационарное поведение может возникнуть в результате смещения средних уровней, изменения параметров базовой структурной модели или того и другого. Если принять во внимание последовательность зависимых от скорости передачи байтов преобразований, обычно выполняемых потоком исходного трафика из приложения на сетевой уровень, может быть применен нелинейный временной ряд. Протоколы управления транспортировкой, такие как TCP, накладывают ограничения на скорость потока данных из источника. В результате отклики источника изменяются как функция мгновенной нагрузки в транзитных сетях, и изменение наблюдается после конечной временной задержки в информации обратной связи.

Некоторые из используемых моделей нелинейных временных рядов: пороговая авторегрессия (TAR), модель на основе ARCH и ARIMA / GARCH. В [14] модель на основе ARCH использовалась для определения эффективной структуры обеспечения динамической пропускной способности в нестационарной среде трафика, которая может адаптироваться к кратковременным колебаниям трафика, соблюдая ограничения по потере данных, их использованию и стоимости сигнализации. Некоторыми из ключевых вопросов, которые рассматривали авторы, были соответствующая модель трафика временных рядов, подходящая граница для предсказанной полосы пропускания, чтобы не было переоценки или недооценки, избегать

высоких затрат на сигнализацию, учитывая правильные обновления полосы пропускания, а также стохастический характер движения.

Чтобы правильно охарактеризовать динамику скорости передачи данных в собранных наборах данных, была рассмотрена модель, основанная на использовании ARCH, с инновационным процессом, обобщенным на класс распределения с тяжелыми хвостами, что является отклонением от традиционного распределения Гаусса, обычно с допущением в большинстве предыдущих работ. Схема обеспечения полосы пропускания в этой статье принимает в качестве входных данных алгоритм прогнозирования измененного вероятностного скачка при распределении полосы пропускания. Основное предположение схем обеспечения состоит в том, что полоса пропускания может быть выделена или освобождена только в дискретных единицах, называемых квантом полосы пропускания. Квант полосы пропускания выражается в виде доли от максимальной доступной полосы пропускания на ссылке. Попытка уменьшить кратковременную потерю данных и увеличить использование полосы пропускания может вызвать частые обновления полосы пропускания, приводящие к колебательному / нестабильному поведению. Чтобы решить эту проблему, авторы придумали схему обеспечения, нацеленную на снижение накладных расходов на сигнализацию в дополнение к удовлетворению потерь и ограничений использования. Они называли эту схему схемой обеспечения мгновенной пропускной способности (IBP).

Выбор модели был сделан главным образом на основе совокупной скорости передачи данных, но не на одном уровне потока. Собранные наборы данных также имели более грубую временную шкалу каждые 5 минут, агрегированную за каждые 15 минут. Этот выбор более грубого временного интервала был сделан главным образом для уменьшения стоимости сигнализации, вызванной частыми обновлениями полосы пропускания. Но исследование более тонкого временного масштаба и его сложности требует рассмотрения. Прогноз был сделан для последующего единственного периода времени, и на основании этого была обеспечена пропускная способность. Прогноз для нескольких таймфреймов и соответствующий метод обеспечения пропускной способности заслуживают изучения. Результаты и показатели эффективности, представленные в этой статье, оказались многообещающими и убедительными.

В [15] предложена комбинация линейных моделей временных рядов ARIMA и нелинейного OARUG, ARIMA / OARUG. Эта модель использовалась главным образом для соответствия взрывному характеру трафика. Авторы представили процедуры оценки параметров и на основе этого использовали метод адаптивного прогнозирования с учетом нестационарного поведения трафика. Их эффективность прогнозирования была проверена на реальных данных движения с использованием схем прогнозирования на один шаг вперед и на  $k$  шагов вперед в трех различных временных масштабах. В их исследовании было обнаружено, что нелинейная модель временного ряда

является более точной моделью с лучшим прогнозом по сравнению с традиционной моделью линейного временного ряда. Нелинейное поведение очень хорошо уловило взрывную природу интернет-трафика. Однако ARIMA / ОАРУГ использовали сложную методологию прогнозирования. Точность прогноза и шкала прогноза еще не определены и оставлены в качестве будущей работы.

В [13] была предложена модель пороговой авторегрессии (TAR) модели нелинейного временного ряда, которая составлена из набора моделей линейной AR в действительных несвязанных субрегионах для трафика данных Интернета и кодируемой переменной скорости передачи битов (VBR) H.261. Эти процессы AR были сгруппированы в соответствии с указанным диапазоном амплитуды. По сути, процесс AR регулирует амплитуду движения, оцениваемую в конкретном субрегионе. В данный момент выбранный субрегион будет зависеть от амплитуд, наблюдаемых по запаздывающим значениям времени. Хотя динамика внутри каждого порога определялась локально линейными процессами AR, совокупный процесс был глобально нелинейным. Авторы разработали интегрированную схему прогнозирования, а также предложили метод для одновременного запуска нескольких предикторов и прогнозирования с минимальной ошибкой прогнозирования. Модель TAR была применена для моделирования данных временных рядов, демонстрирующих циклическое поведение и особенности ДЗ.

### **2.1.2 Альтернативные модели**

Есть и другие модели, используемые в дорожном моделировании. В [16] был предложен пуассоновский процесс дробового шума для моделирования трафика на уровне потока, предполагая, что характеристики трафика могут быть зафиксированы на уровне потока. Они использовали реальные следы трафика, полученные из магистральной сети Sprint IP. Было обнаружено, что модель очень похожа на реальный магистральный трафик, и, кроме того, она была названа простой моделью. Нечеткая логика использовалась для моделирования движения, прогнозирования и контроля заторов в [17]. Авторы предложили модель нечеткой авторегрессии (fuzzy-AR) для анализа характеристик трафика в высокоскоростных сетях. Метод нечеткой кластеризации использовался для объединения нескольких линейных локальных процессов AR, чтобы смоделировать модель нечеткой AR, которая может близко напоминать нестационарный нелинейный процесс. Авторы использовали фактический пакет Ethernet-LAN для проверки предложенного метода. В [18] была предложена адаптивная архитектура нейронной сети для онлайн-моделирования, а также для моделирования отдельных трафиков. Вейвлет-модели использовались для изучения глобальных сетей, и было обнаружено, что интернет-трафик имеет сложное масштабирование и обладает мультифрактальными характеристиками.

Сравнение и оценка М/М/1 модели. В этом эксперименте мы сравниваем точность подобранной модели М/М/1 и однослойной нейронной сети.

Для каждой тренировки оптимальные параметры для экспериментальной модели М/М/1 (количество звеньев М и размер шага) и для нейронной сети (параметр регуляризации и количество нейронов в скрытом слое) выбираются с использованием методологии перекрестной проверки. Мы повторяем процесс обучения и валидации достаточное количество раз в каждом случае, чтобы гарантировать, что результаты обеспечивают бета-достоверность не менее 5%.

После выбора перекрестной проверки окончательные результаты, отображаемые на графиках, получают с использованием независимого набора тестов. Различные параметры, выбранные для перекрестной проверки моделей, ограничены следующим:

Размер шага = (0,00001, 0,0001, 0,001, 0,01, 0,1, 1,0). М = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10). Параметры регуляризации = (0,0001, 0,001, 0,01, 0,1, 1,0, 1,0, 10). Количество нейронов в скрытом слое = (2, 3, 4, 5, 7, 10, 15, 20, 30, 40, 50, 60, 80, 100).

Результаты были оценены с использованием нормализованной относительной ошибки:

$$e = \sqrt{\frac{1}{T} \cdot \sum_{t=0}^T \left( \frac{\hat{Y}_t - Y_t}{Y_t} \right)^2}, \quad (1)$$

где Т указывает число примеров;

$\hat{Y}_t$  - ожидаемый результат;

$Y_t$  - ожидаемый.

Первый эксперимент сравнивает обучающую способность подобранной модели с нейронной сетью, соблюдая необходимое количество примеров в каждом конкретном случае для достижения определенной частоты ошибок. В этом первом эксперименте и для практических целей мы ограничили моделирование только тремя активными передающими станциями в сети.

На рисунке 2.1 показан наилучший коэффициент ошибок при тестировании, которого могла достичь каждая модель, в зависимости от размера обучающего набора. Выбранные параметры после выбора перекрестной проверки в этом случае М=8 и размер шага 0,01 для подогнанного М/М/1, в то время как 50 скрытых нейронов и стоимостный параметр 1,0 оказались лучшими для нейронного выбора. Результаты ясно показывают, как нейронная сеть узнает лучше и быстрее паттерны, свойственные данным.

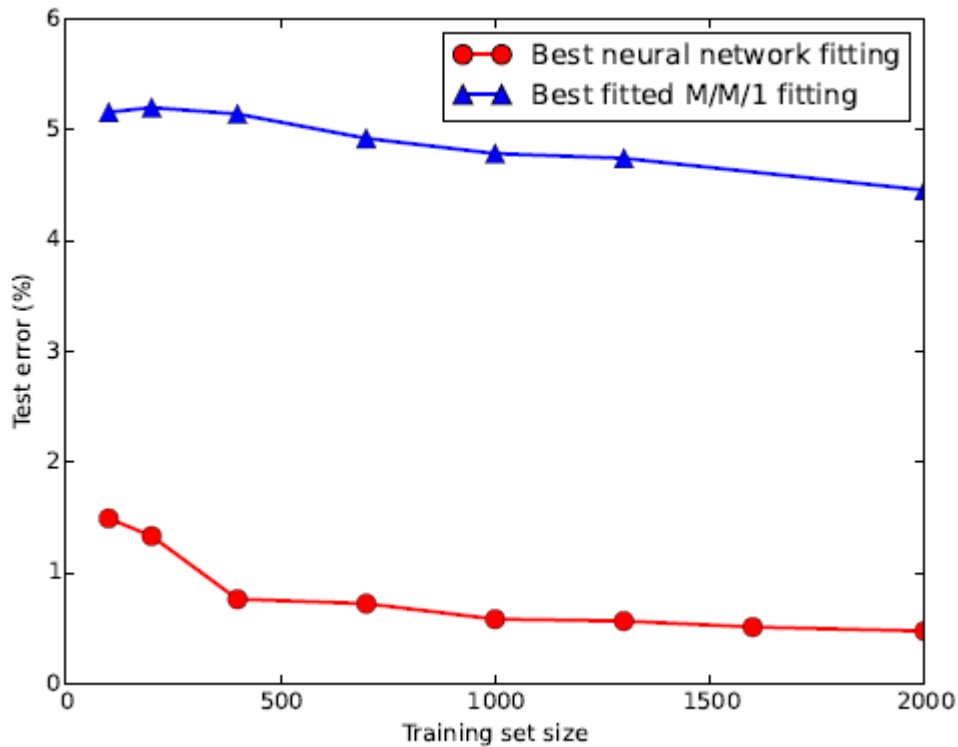


Рисунок 2.1 - Эволюция относительной ошибки теста в зависимости от размера тренировочного набора

В следующих экспериментах мы проверяем, как насыщение в каналах влияет на прогнозы и насколько надежны результаты для увеличения количества активных (передающих) станций в нашей сети. На рисунке 2.2 и 2.3 показывают соответственно, как изменяется наилучшая частота ошибок тестирования, когда мы увеличиваем количество активных станций и уровень насыщения в линиях связи. Для второго случая мы устанавливаем ссылку на третий уровень, где в этот момент ссылки (в среднем), как ожидается, начнут насыщаться. Моделирование было разработано в соответствии с тем же подходом, который описан в первом эксперименте.

Опять же, экспериментальная модель M/M/1 работает плохо по сравнению с нейронной сетью, с дополнительным значительно плохим допуском к увеличению нагрузки в сети.

Низкая производительность экспериментальной модели M/M/1 может объясняться отсутствием в нашей сети допущений Джексона, таких как общая независимость поступлений или экспоненциальное распределение длин пакетов во всех последовательных маршрутизаторах.



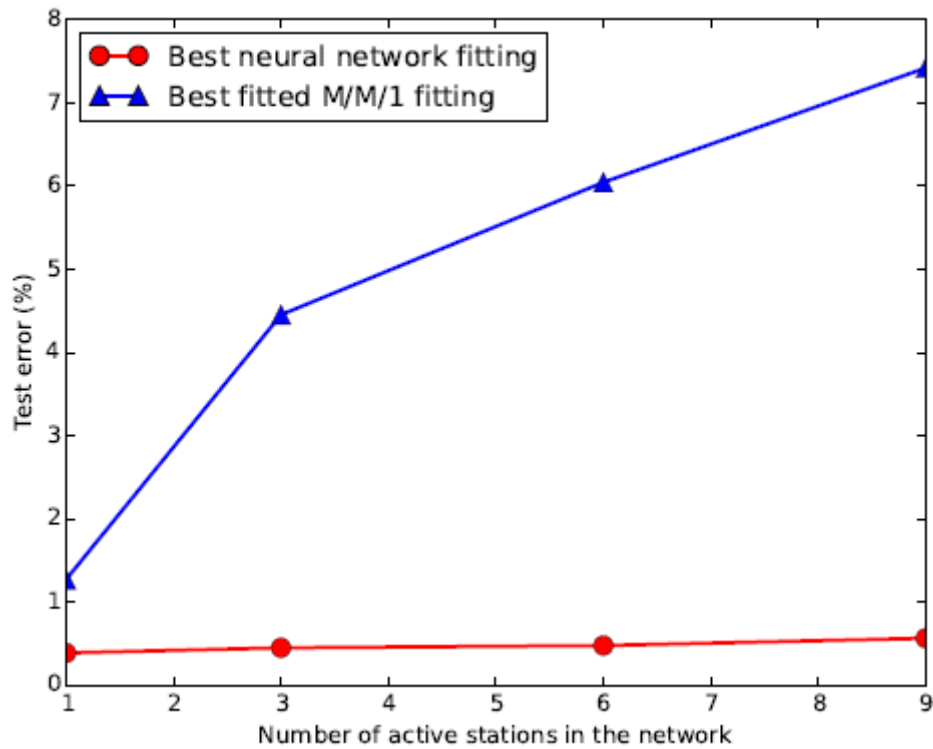


Рисунок 2.2 - Эволюция относительной погрешности теста в зависимости от количества активно передающих станций в сети

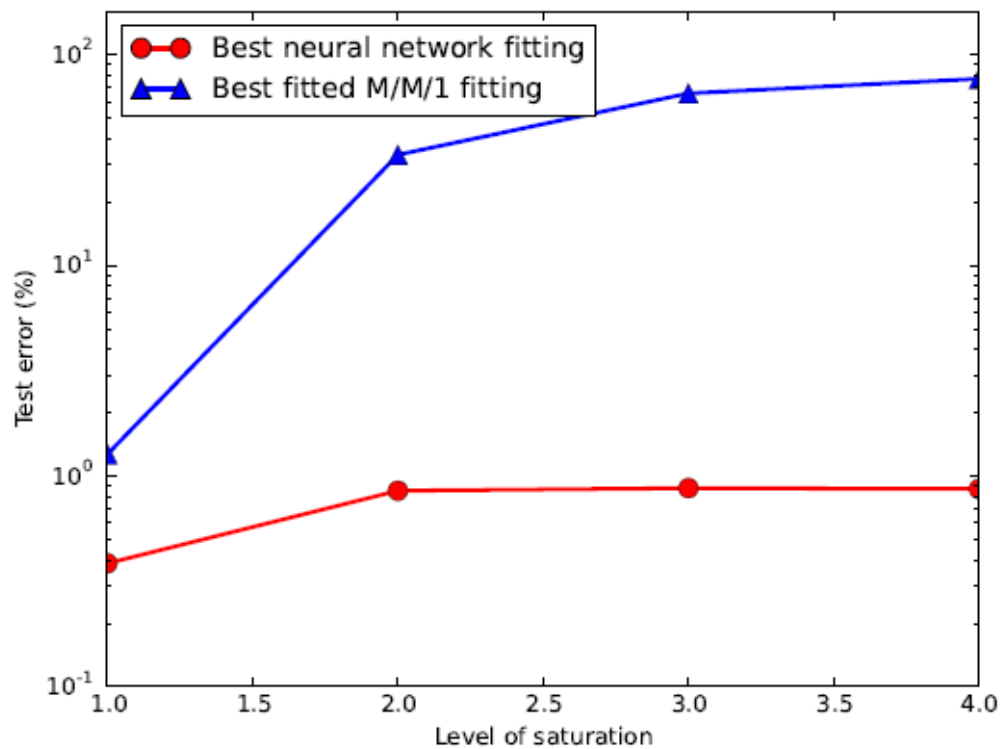


Рисунок 2.3 - Эволюция относительной погрешности теста как функции среднего уровня насыщения в ссылках (логарифмическая шкала)

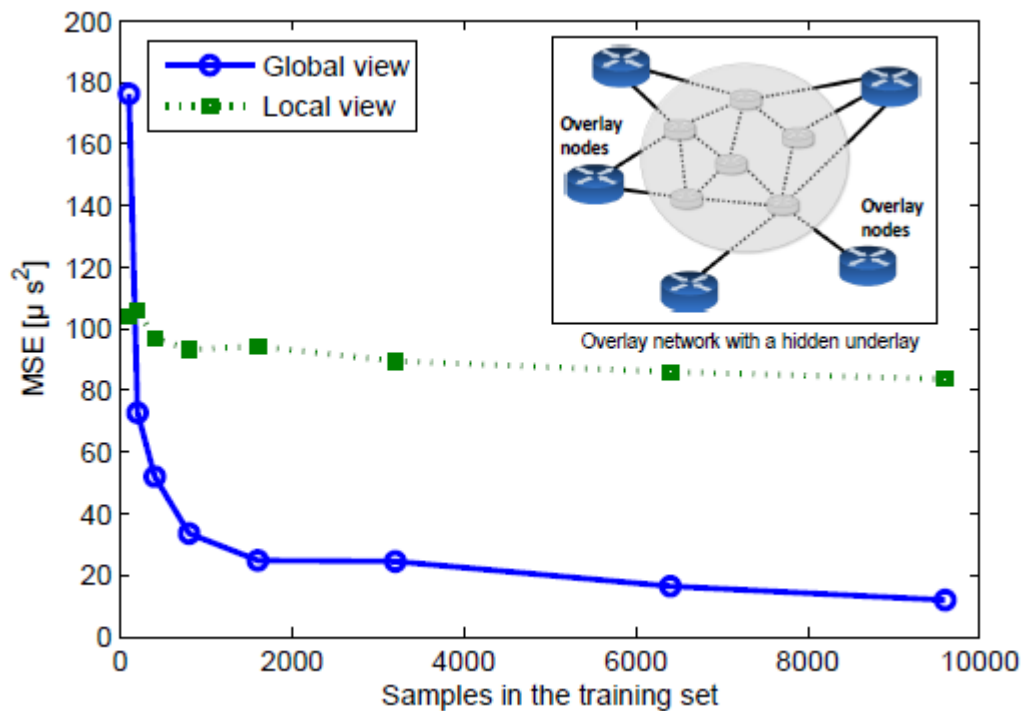


Рисунок 2.4 - Ошибка прогнозирования (среднеквадратическая ошибка) как функция размера обучающего набора в сценарии с наложением-наложением, как для глобального, так и для локального просмотра

После того, как мы показали, что модели нейронных сетей превосходят регрессор М/М/1, мы хотим убедиться, что использование глобальной информации сети необходимо для лучшего моделирования сквозной задержки сети. Мы также хотим проверить размер тренировочного набора, необходимого для обучения модели.

Для этого мы обучаем две разные модели ANN. Модель локального представления использует только информацию, полученную от одного узла, для прогнозирования задержек, наблюдаемых от этого узла. Модель глобального представления имеет ту же структуру, что и используемая в предыдущем эксперименте. Мы обучаем эти модели разному количеству выборок, что позволяет охарактеризовать влияние размера набора данных на тренировочный процесс.

На рисунке 2.4 показана погрешность (точность) модели в зависимости от размера тренировочного набора (сплошная линия). Эта ошибка показывает, насколько точно модель прогнозирует задержку при известной маршрутизации и трафике, но не топологию. Как показано на рисунке, относительная ошибка составляет примерно 1% при использовании 6400 обучающих выборок, что эквивалентно среднеквадратичной ошибке  $20 \text{ мс}^2$ . В дополнение к этому, рисунок 2.4 также показывает (пунктирная линия)

когда модель обучается только с использованием местной информации. Основная причина этого эксперимента состоит в том, что мы стремимся проверить основную гипотезу, изложенную в этой статье: ML, примененный к

глобальному представлению, дает лучшие результаты, чем когда доступна только локальная информация. Для этого каждый оверлейный узел обучается только локальному трафику, маршрутизации и задержке, и, как показывают результаты, точность в этом случае сильно ухудшается. Это связано с тем, что задержка между двумя узлами зависит от состояния очередей подстилающей сети, которое, в свою очередь, зависит от общего трафика сети.

Мы изучили производительность двух разных моделей, чтобы охарактеризовать задержки в сети с учетом нагрузки трафика. Мы продемонстрировали, что простые оценщики на основе нейронных сетей работают замечательно. Однако эти оценщики моделируют сеть как черный ящик и не предоставляют внутреннюю информацию о поведении сети. Для сравнения мы внедрили оценщик, основанный на M/M/1, который мог бы предоставить больше информации о сети, но это привело к плохим результатам. Кроме того, мы продемонстрировали, что модель, обученная с полной информацией о сети, превосходит модели, обученные с локальным представлением о сети. Наконец, мы также убедились, что нам нужны тысячи образцов для обучения модели для сети, используемой в этой главе.

Сетевое моделирование с использованием ANN. В контексте KDN, методы обучения могут использоваться для автоматического управления сетью через контроллер SDN благодаря информации о мониторинге сети, полученной через платформу NA. Эта новая сетевая парадигма известна как сеть, основанная на знаниях (KDN).

В рамках парадигмы KDN существует множество вариантов использования технологий ML в компьютерных сетях. Среди всех таких потенциальных вариантов использования в этой главе мы сосредоточимся на одном: моделирование производительности сети с использованием нейронных сетей. Основная причина этого заключается в том, что сетевое моделирование является центральным элементом многих сетевых операций, особенно в области оптимизации сети. Обычно алгоритмы оптимизации сети требуют сетевой модели, в которой используются методы оптимизации, чтобы найти лучший элемент. В этой главе мы стремимся ответить на следующий вопрос: могут ли нейронные сети точно смоделировать задержку компьютерной сети как функцию входного трафика? Для этого мы предполагаем, что сеть представляет собой черный ящик с входным трафиком и задержками на выходе. Следуя этому подходу, мы экспериментально оцениваем точность нейронной сети при оценке сквозной задержки трафика, мы также сравниваем ее с другими известными регрессорами и изучаем влияние основных характеристик сети (топология, маршрутизация и т. д.) в точности оценок задержек.

Вопрос, который мы стремимся рассмотреть в этой главе, является фундаментальным для сетевого моделирования. Действительно, как аналитические (например, модели массового обслуживания), так и вычислительные модели (например, имитаторы) являются хорошо известными методами, используемыми для оценки производительности сети

на основе ее входного трафика. В этой главе мы утверждаем, что нейронные сети могут представлять третью опору в области сетевого моделирования, обеспечивая соответствующие преимущества по сравнению с традиционными методами. Насколько нам известно, это первая попытка смоделировать компьютерную сеть с использованием нейронных сетей.

Действительно, в этой главе мы выступаем за то, чтобы нейронные сети представляли третью опору в области сетевого моделирования. Нейронные сети могут эффективно дополнять существующие аналитические и вычислительные методы, особенно в сценариях оптимизации сети. Основными преимуществами нейронных сетей по отношению к существующим методам являются:

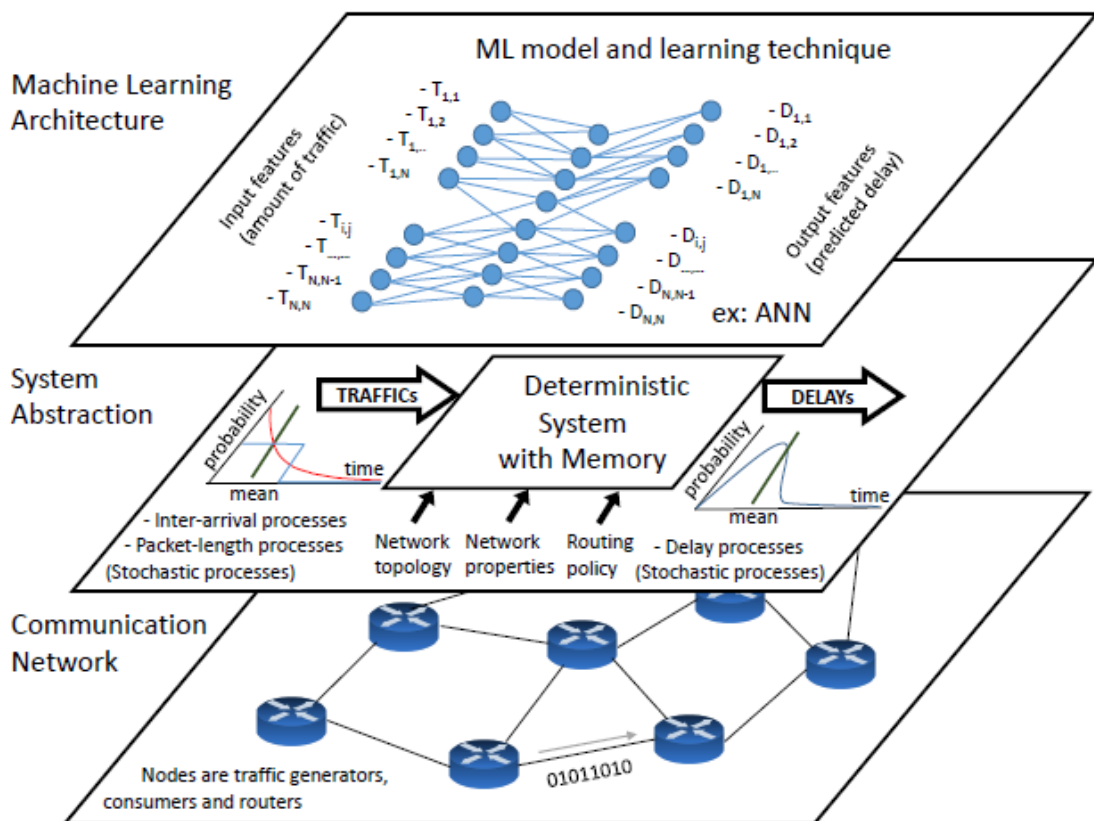


Рисунок 2.5 - Графическое представление постановки задачи, рассматриваемой в этой главе

Следует описать постановку задачи, которую мы стремимся решить. Рисунок 2.5 суммирует постановку задачи, используя три уровня. Нижний уровень представляет реальную физическую сетевую инфраструктуру, которая имеет определенные фундаментальные характеристики, такие как топология, размер, маршрутизация и т. д.

Средний уровень представляет системную абстракцию, где сеть считается черным ящиком, трафик входит в ящик и выходит с определенной средней задержкой. Трафик описывается стохастическими распределениями, как процессом между поступлениями, так и процессом длины пакета. Эти

случайные процессы объединяются в сети, которая является детерминированной сложной системой с определенными свойствами (топология, маршрутизация и т. д.) И памятью, когда случайные процессы, такие как физические ошибки, не принимаются во внимание.

Наконец, верхний уровень представляет нейронную сеть, которая моделирует производительность компьютерной сети. Нейронная сеть способна производить оценки средней сквозной задержки для всех пар узлов, рассматривая входной трафик как матрицу трафика [вход, выход]. Характеристики сети (маршрутизация, топология и т. д.) скрыты от нейронной сети, и, следовательно, нейронная сеть обучается только для одной конкретной конфигурации сетевой инфраструктуры, то есть определенной топологии, маршрутизации и т. д.

Основные вопросы, которые мы стремимся рассмотреть в этой главе:

– можем ли мы обучить нейронную сеть для получения точных оценок средней задержки от конца до конца для всех пар узлов с учетом матрицы входного трафика (вход, выход)?

– какое влияние оказывают основные характеристики сети (топология, маршрутизация, размер, интенсивность трафика) на точность нейронной сети?

– как точность нейронной сети сравнивается с другими известными регрессорами?

Основная цель этой главы - понять, можно ли использовать нейронную сеть для моделирования сети, в частности, для оценки средней задержки компьютерной сети как функции матрицы входного трафика.

Сетевое моделирование является хорошо известной областью, которая предоставляет методы, которые являются центральными для широкого спектра коммуникационных функций, например, при оптимизации сети. Чтобы оптимизировать модели, используемые для оценки производительности конфигураций, алгоритм оптимизации ищет пространство конфигурации, используя эти модели, чтобы найти наилучшую возможную, то есть конфигурацию, для которой модель оценивает наилучшую производительность. Яркими примерами этого являются области Traffic Engineering (TE), где модели сети используются для поиска конфигураций маршрутизации, которые достигают определенной целевой производительности.

В области сетевого моделирования существуют фундаментальные подходы: аналитические и вычислительные модели (имитация). Что касается аналитических методов, теория цепей Маркова широко использовалась в теории очередей для моделирования поведения одиночной очереди, допуская определенные стохастические свойства процессов прибытия и завершения работы (например, M/M/1, M/D/1...). Эти модели были распространены на модельные сети узлов, т. е. Сети массового обслуживания. Примерами этих теорий являются: сети Джексона, теорема Гордона-Ньюэлла, анализ среднего значения, алгоритм Бузена, сеть Келли, сеть G, сеть ВСМР. Вычислительные модели также являются еще одним популярным методом моделирования

поведения сетей. Обычно симуляторы работают либо на уровне пакетов, либо на уровне потока и упрощают имитируемые ими сетевые протоколы.

Механизмы машинного обучения использовались в области связи, такие методы широко использовались в области анализа трафика, безопасности сети и анализа первопричин. Кроме того, в некоторых работах предлагается использовать методы обучения армированию для оптимизации маршрутов. Однако, насколько нам известно, это первая попытка смоделировать сеть как черный ящик с использованием нейронных сетей.

Точность в сложных сценариях. Как правило, аналитические методы основаны на строгих упрощающих допущениях базовой сетевой инфраструктуры: это потому, что они должны быть отслеживаемыми. С другой стороны, симуляции могут моделировать сложное поведение, но это требует больших затрат на разработку и вычислений. Методы ML и особенно нейронные сети очень хорошо работают со сложностью (например, нелинейным поведением) и сценариями высокой размерности. Как таковые, хорошо обученные нейронные сети могут моделировать компьютерные сети, не делая каких-либо упрощающих допущений, обеспечивая при этом точные оценки.

Быстрый и легкий: нейронные сети требуют значительных вычислительных ресурсов для обучения, но после обучения они быстрые и легкие. Действительно, они могут произвести оценку производительности сети за один шаг и потребовать очень мало ресурсов для запуска. Это представляет собой важное преимущество, особенно перед симуляторами, которые требуют значительных вычислительных ресурсов для запуска и могут быть медленными.

Основным недостатком нейронных сетей является то, что они основаны на данных и поэтому требуют больших обучающих наборов, а также вычислительных ресурсов для фазы обучения. Чтобы экспериментально проанализировать точность нейронной сети, мы генерируем различные обучающие наборы с помощью имитаций, в каждом наборе мы меняем различные характеристики сети: распределение трафика, интенсивность трафика, топологию, размер и политику маршрутизации и измеряем среднюю задержку.

После того, как мы сгенерировали набор данных, мы используем их для обучения набора регрессоров, включая нейронную сеть, а затем мы оцениваем его точность, используя метод перекрестной проверки. Мы разделили набор данных на три набора: обучающий набор с 60% выборок, проверочный набор с 20% выборок и тестовый набор с оставшимися 20% выборок. Обучающий набор используется для оптимизации модели ML, проверочный набор используется для оценки модели на этапе обучения, а испытательный набор - для независимой оценки эффективности. При этом мы сравниваем среднюю задержку, оцененную регрессорами, с задержкой, измеренной на симуляторе. В конечном итоге мы хотим понять, как точность нейронной сети, так и ее

связь с основными характеристиками сетей: распределением трафика, интенсивностью трафика, топологией, размером и политикой маршрутизации.

Чтобы сгенерировать набор данных, мы используем симулятор Omnet++ (версия 4.6), в каждой симуляции мы измеряем среднюю сквозную задержку в течение 16 тысяч единиц времени для всех пар узлов. Скорость передачи всех каналов в сети установлена на 10 килобит в единицу времени, а средний размер пакетов составляет 1 килобит. Для набора данных мы

Рассмотрим следующие параметры:

- топология: мы исследуем 3 различные топологии сети: однонаправленные кольцевые, звездообразные и без масштабные сети;

- размер сети: мы изучаем сети от 3 до 15 узлов, где все узлы являются активными передатчиками и / или приемниками;

- распределение трафика: мы оцениваем 4 различных распределения длины пакета: детерминированное (постоянное), равномерное, биномиальное и пуассоновское с использованием фиксированной средней длины пакета. Во всех случаях время прибытия является экспоненциальным;

- интенсивность трафика. Мы исследуем различные уровни насыщенности в сети, варьируя интенсивность трафика. Для этого мы передаем среди всех пар узлов случайное значение трафика с максимальным значением (max);

- маршрутизация. Мы исследуем четыре различные конфигурации маршрутизации.

Всего мы сгенерировали 380 различных наборов данных с различными конфигурациями, чтобы оценить точность нейронной сети. Каждый набор данных состоит из 10000 различных симуляций со случайными матрицами трафика между 0 и максимальным.

Набор данных используется для обучения различных регрессоров: глубоких нейронных сетей (DNN) (с использованием одного и двух скрытых слоев) и полиномиальной регрессии. Для DNN мы исследуем следующие гиперпараметры: количество скрытых нейронов, функция активации, скорость обучения и параметр регуляризации. Для случая полиномиального регрессора мы исследуем его степень. Из всех исследованных моделей мы выбираем лучшую, использующую метод перекрестной проверки. Обратите внимание, что мы используем независимый набор тестов для оценки точности модели.

С точки зрения реализации мы используем библиотеку Tensorflow (версия 1.2.1) для реализации моделей DNN. После ручной настройки гиперпараметров мы использовали следующую конфигурацию:

Функция активации: сигмовидная, количество скрытых слоев: равно количеству входных данных, то есть квадрату количества узлов в сети, максимальная эпоха обучения: 7 500 000, алгоритм обучения: Adam Optimizer, функция стоимости: MSE с регуляризацией L2, параметр регуляризации L2: 0,00003.

Для полиномиальной регуляризации мы использовали инструмент LinearRegression в sklearn (версия 0.19). Для степени выше единицы мы

пропустили термины, состоящие из нескольких продуктов, из-за высокой размерности данных. В результатах мы вычисляем точность моделей как «ошибку обучения», выраженную в процентах:

$$\text{error} = 100 \sqrt{\frac{1}{N} \sum_{i=1}^N (d_i^{\wedge} - d_i)^2}, \quad (2)$$

где N - усредненное количество выборок, включая размер набора тестов и другую пару узлов.

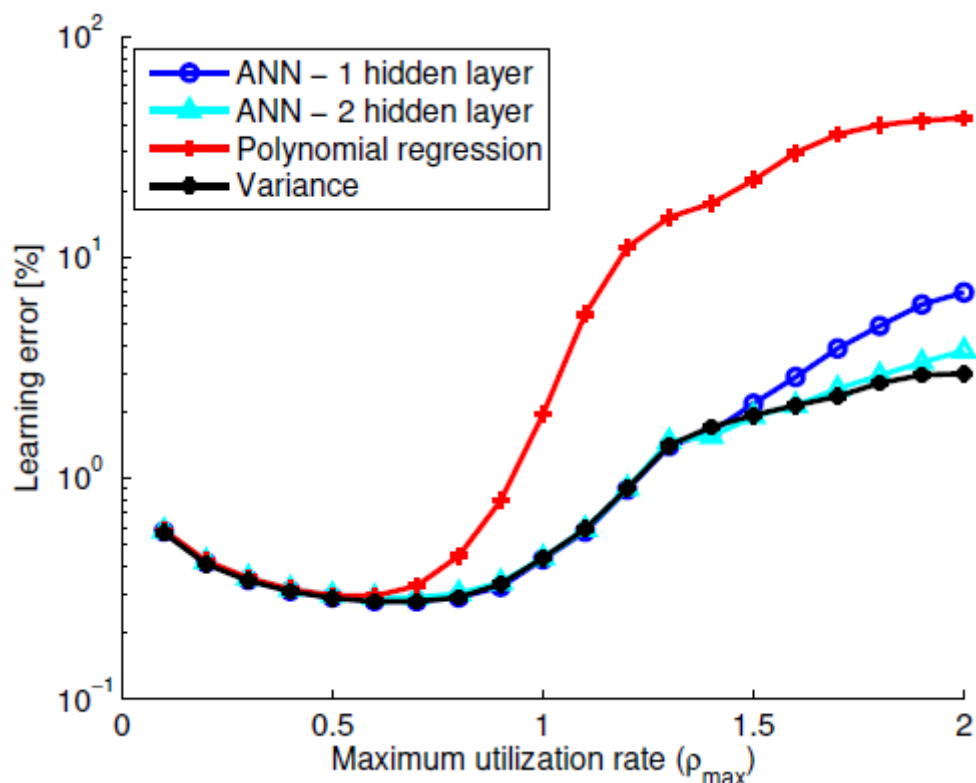


Рисунок 2.6 - Ошибка обучения (логарифмическая шкала) как функция насыщения сети для биномиального трафика

Характеристики и интенсивность движения. Сначала мы сосредоточимся на точности нейронной сети при оценке задержки различной интенсивности трафика и распределения размера пакета. На рисунке 2.6 показана точность разных регрессоров в 10-узловой безмасштабной сети с разной интенсивностью трафика и для распределения биномиального размера пакета.

Обратите внимание, что интенсивность трафика выражается как (макс). Каждая пара узлов генерирует равномерно распределенную случайную полосу пропускания с максимальной (максимальной). В качестве примера (max=2) показано, что для каждого моделирования каждый узел сети для каждого пункта назначения генерирует случайный трафик после указанного



распределения со случайной скоростью в диапазоне от 0 до удвоенной пропускной способности канала, деленной на число направлений.

Как показано на рисунке 2.6, нейронные сети демонстрируют интересное поведение, во-первых, они замечательно работают с ошибкой ниже 5% во всех конфигурациях. Чтобы лучше понять эту ошибку, мы также наносим на график дисперсию набора данных (черная линия на том же рисунке). Дисперсия аппроксимируется путем повторения 1000 раз каждого моделирования, которое используется для оценки «реальной» задержки, и использования этих выборок в приближении дисперсии. Дисперсия также выражается в процентах.

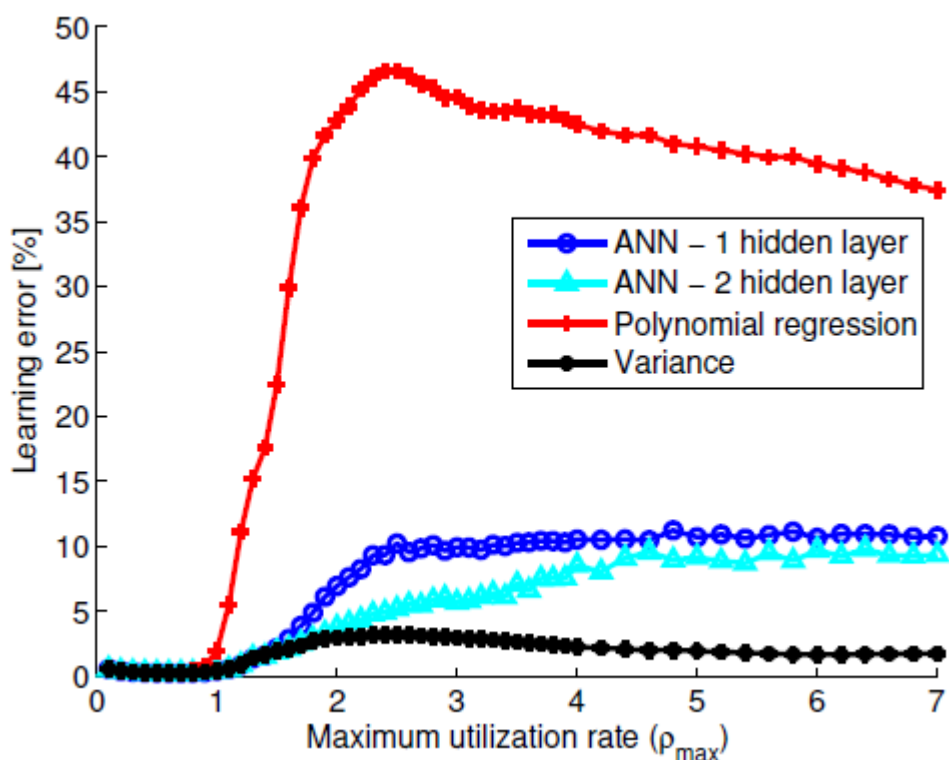


Рисунок 2.7 - Ошибка обучения (линейный масштаб) как функция насыщения сети для биномиального трафика в более насыщенных сетях

Кроме того, на рисунке 2.7 показан тот же эксперимент в линейном масштабе и для более насыщенного трафика. Также можно наблюдать, что производительность полиномиальной регрессии, когда сеть начинает насыщаться, действительно плохая. Однако мы также можем наблюдать, как производительность моделей нейронных сетей начинает ухудшаться для более насыщенного трафика. Это можно объяснить двумя фактами: 1) с ростом насыщенности модель становится более нелинейной и более трудной для изучения, и 2) наши нейронные сети были оптимизированы в области между одним и двумя, и она начинает расходиться по мере того, как мы расстояние от этого региона.

Понимание дисперсии обучающего набора: поскольку на рисунке показано, что наборы данных содержат дисперсию, это является результатом процесса измерения. Задержка на пакет, измеренная в каждом моделировании, следует определенному распределению, которое имеет дисперсию. Затем средняя сквозная задержка оценивается в течение конечного времени: 16 тысяч единиц времени в наших экспериментах. В результате использования конечного числа выборок дисперсия приводит к неустранимой ошибке при обучении регрессоров. Эта ошибка может быть уменьшена только путем увеличения количества выборок.

Время усреднения: чтобы проиллюстрировать этот эффект, мы изображаем на рисунке 2.8 влияние времени усреднения, использованного при моделировании (эквивалентное количеству пакетов), с ошибкой обучения. Для этого эксперимента мы используем 10-узловую топологию без масштаба  $s^2$  Max=0.8. Как и следовало ожидать, на рисунке показано резкое уменьшение ошибки обучения из-за уменьшения дисперсии данных. Стоит отметить, что это не выполняется, когда присутствует самоподобный трафик, поскольку они имеют бесконечную дисперсию. Основным выводом из этого является то, что ошибка измерения набора данных распространяется регрессорами как ошибка обучения.

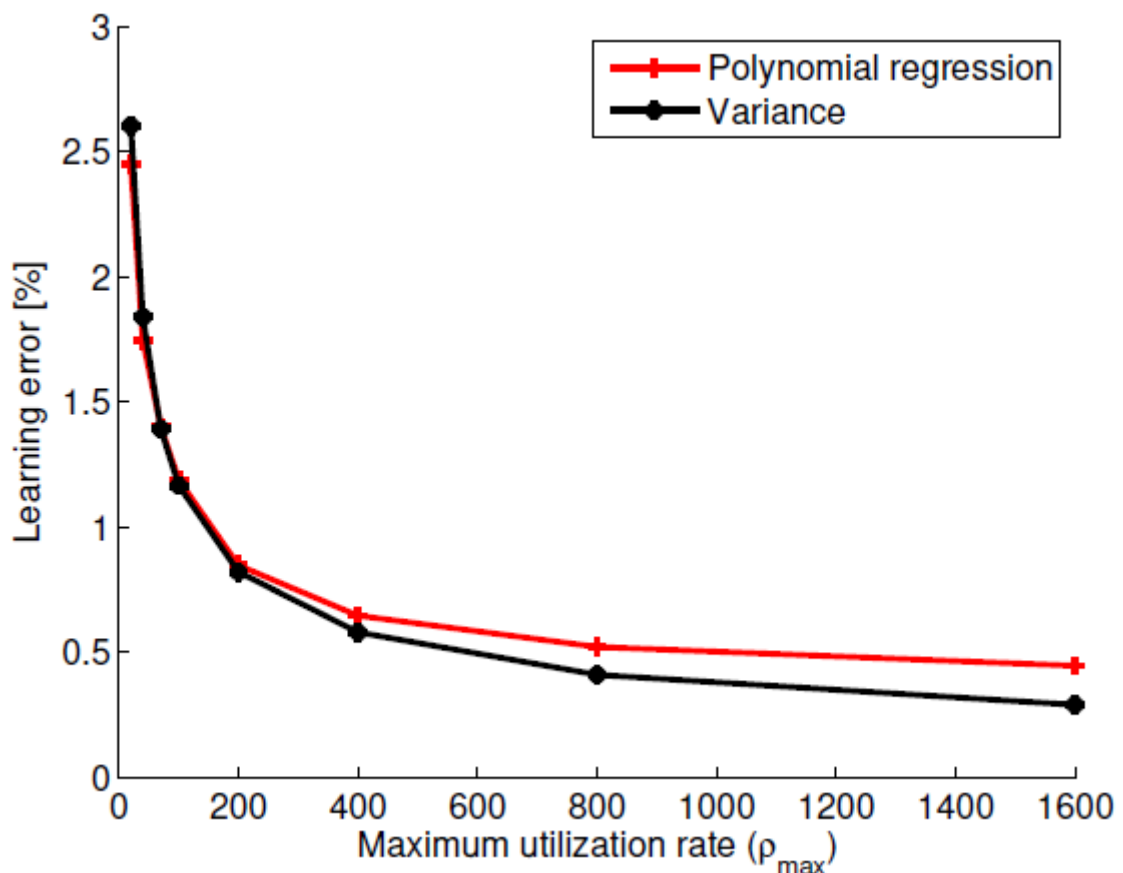


Рисунок 2.8 - Ошибка обучения как функция времени усреднения

Дисперсия и интенсивность трафика. Мы установили, что набор данных содержит дисперсию в результате процесса измерения, но рисунок 2.6 также показывает, что дисперсия зависит от интенсивности трафика, почему? Эта разница в задержке пакетов вызвана двумя факторами: (i) разным временем передачи пакетов из-за разного размера и (ii) разным временем очереди. Поскольку сеть становится более насыщенной ( $\rho_{max} > 1$ ), пакеты испытывают различные задержки в очереди, и в результате задержка имеет большую дисперсию. Как только сеть начинает приближаться к полному насыщению ( $\rho_{max} > 2$ ), большой процент пакетов находит очереди заполненными, и в результате задержка становится более постоянной, а дисперсия - меньшей.

Другие распределения трафика:

На рисунках 2.9, 2.10 и 2.11 показана точность разных регрессоров в одной и той же 10-узловой безмасштабной сети с разной интенсивностью трафика и для другого распределения размера пакета: распределение Пуассона, детерминированное распределение и постоянная длина пакета.

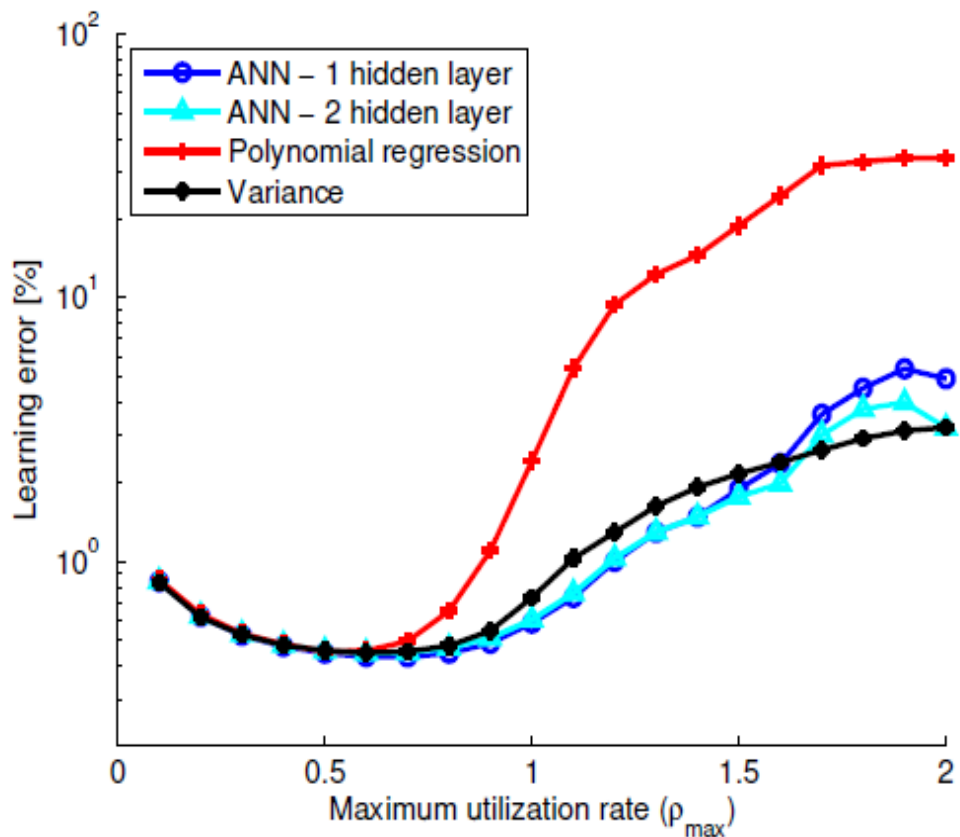


Рисунок 2.9 - Ошибка обучения (масштаб журнала) как функция насыщения сети для пуассоновского трафика

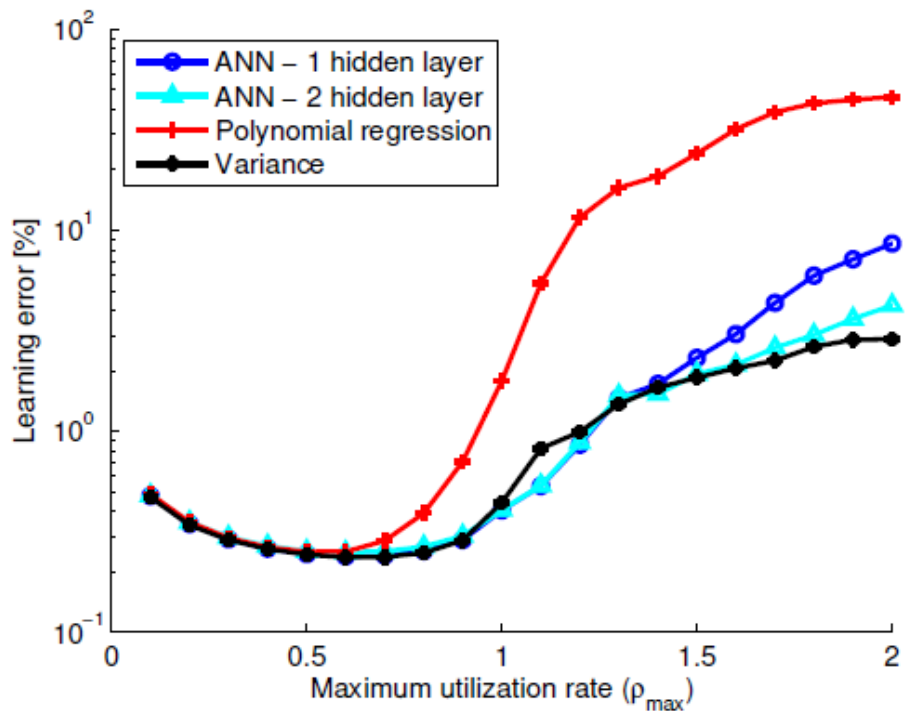


Рисунок 2.10 - Ошибка обучения (масштаб журнала) как функция насыщенности сети для равномерного трафика

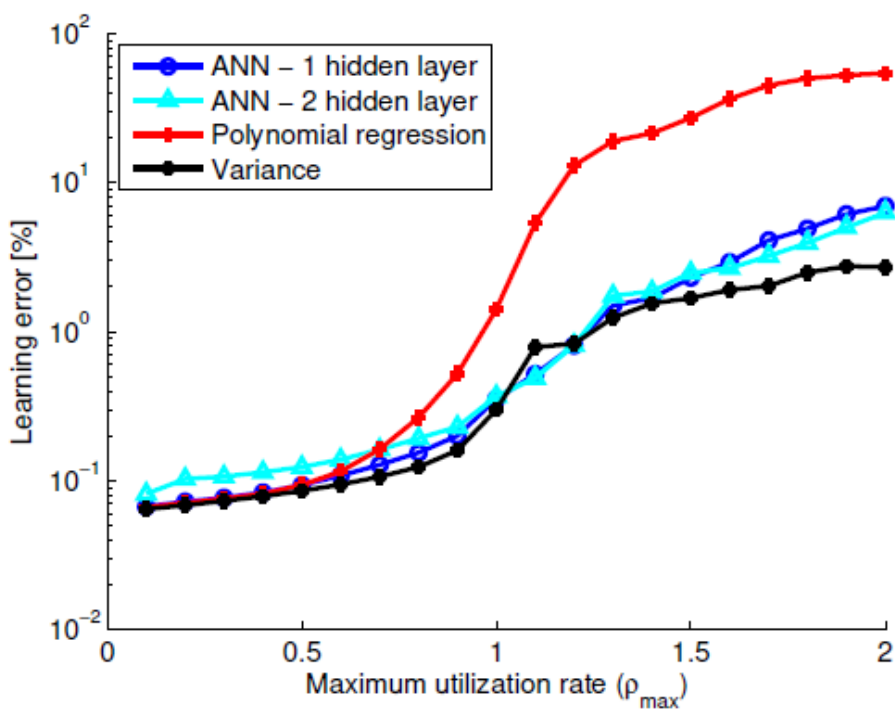


Рисунок 2.11 - Ошибка обучения (масштаб журнала) как функция насыщения сети для детерминированного трафика

Рисунки 2.10 и 2.11, которые представляют Пуассона и равномерно распределенную длину пакета, представляют поведение, аналогичное биномиальному случаю. Обратите внимание, что в некоторых точках может наблюдаться ошибка обучения ниже, чем дисперсия. Это потому, что

дисперсия является приблизительной, а в некоторых случаях она может быть завышена. В обоих случаях сеть из 2 скрытых слоев всегда обеспечивает лучшую производительность.

Наконец, результаты, полученные с использованием постоянной длины пакета, представленные на рисунке 2.12, немного отличаются. Поведение, когда сеть не насыщена, отличается. Причина в том, что в этом случае дисперсия вызвана только временем очереди, тогда как в других случаях она также обусловлена различным размером каждого пакета. Другое отличие состоит в том, что априори может наблюдаться более низкая производительность двух слоев сеть. Это может быть объяснено тем фактом, что изученная функция в детерминированном случае является более крутой. Резюме: с этим мы установили, что:

- обучающий набор содержит дисперсию, которая является результатом использования конечного числа пакетов для измерения задержки;
- эта разница приводит к неустраняемой ошибке при обучении регрессоров;
- дисперсия задержки зависит главным образом от времени ожидания.

Нейронные сети обеспечивают точные оценки задержки, а более глубокие сети (с 2 уровнями) могут привести к незначительной ошибке, почти совпадающей с неснижаемой ошибкой, возникающей в результате дисперсии. Тот факт, что более глубокие сети работают лучше, предполагает, что модель задержки является достаточно сложной, чтобы оправдать использование DNN, и что в более глубоких сетях эта ошибка может быть уменьшена.

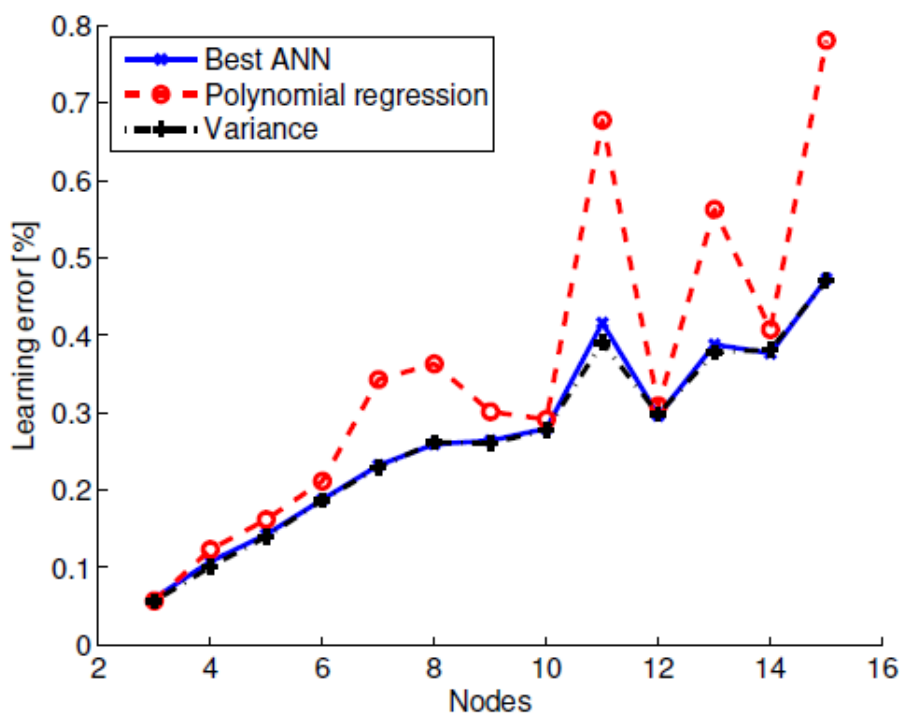


Рисунок 2.12 - Топология без масштаба

Топологии и размер сети. В этом разделе мы исследуем точность нейронной сети при оценке задержки при разных топологиях и размерах сети.

На рисунках 2.13 и 2.14 показана точность регрессоров при оценке задержки в кольцевой и безмасштабной топологии с различными размерами, в диапазоне от 3 до 15 узлов. В этом сценарии интенсивность трафика установлена на (максимум>0:6). Как показано на рисунке, оба регрессора (полиномиальная и нейронная сеть) способны давать точные оценки с ошибками обучения, которые соответствуют дисперсии. Основная причина этого заключается в том, что при такой интенсивности и топологиях трафика трафик не подвергается серьезным очередям. Это также видно по низкой дисперсии обоих сценариев. Исключение составляют некоторые безмасштабные сети, в которых некоторая связь может быть насыщенной, что снижает производительность полиномиальной регрессии.

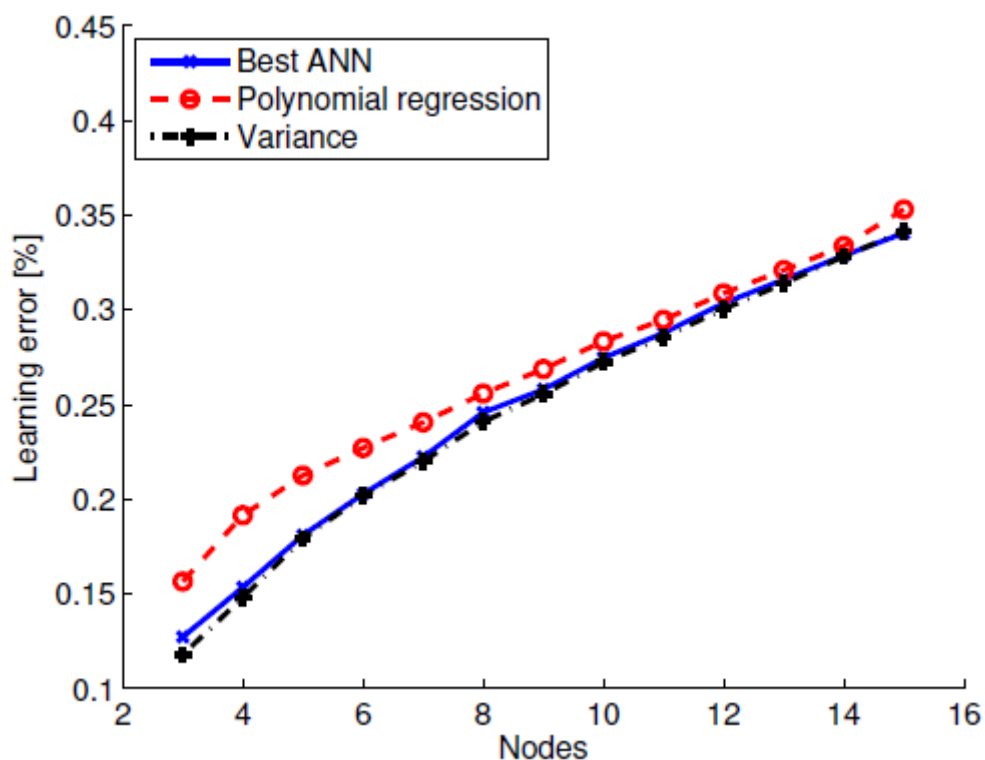


Рисунок 2.13 – Топология звезда

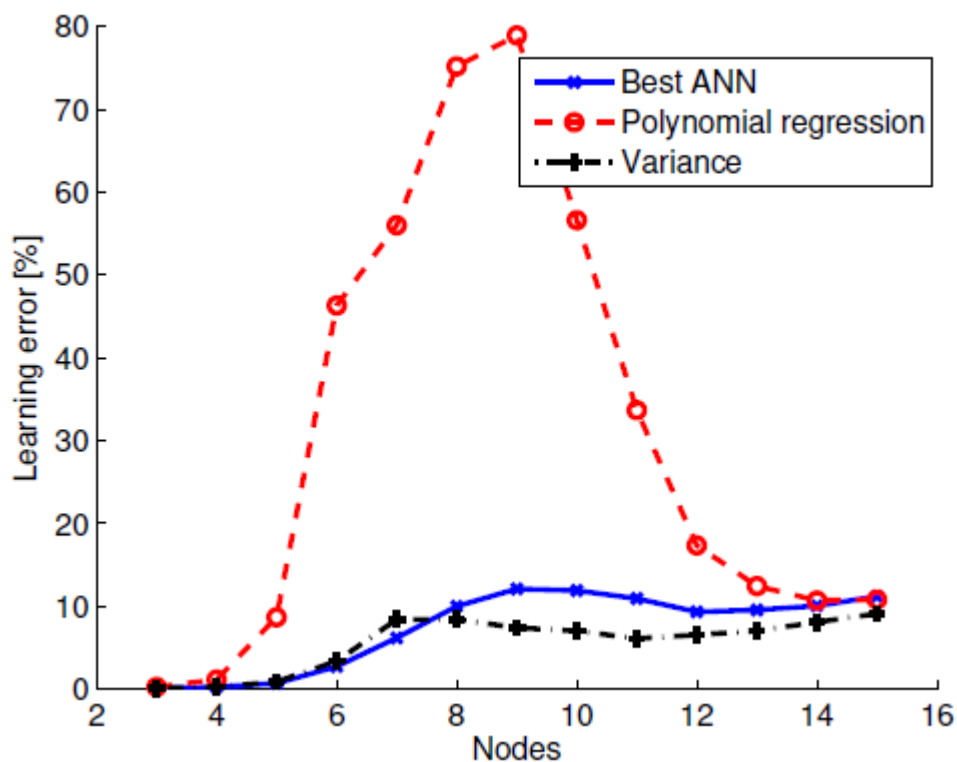


Рисунок 2.14 - Топология кольца

Таблица 2.1 – Ошибка обучения

|      | NN     | Poly   | Variance |
|------|--------|--------|----------|
| SP1  | 0.686% | 0.722% | 0.694%   |
| SP2  | 0.683% | 0.689% | 0.674%   |
| MAN  | 0.707% | 0.703% | 0.696%   |
| POOR | 2.241% | 6.751% | 2.314%   |

На рисунке 2.14 показана точность для кольцевой топологии с различными размерами сети, снова в диапазоне от 3 до 15. Как и в предыдущем случае, интенсивность трафика установлена на ( $\max > 0: 6$ ). Как показано на рисунке, в этом случае полиномиальные регрессоры работают плохо с высокими ошибками обучения, с другой стороны, нейронная сеть показывает превосходную точность, снова совпадая с дисперсией. Причиной этого является то, что кольцевая топология легко насыщается при различных конфигурациях, что приводит к более высокой дисперсии и более сложным функциям для характеристики.

На обоих графиках показано, что для рассматриваемых вариантов использования топология или размер не влияют на точность оценок нейронной сети. Единственное влияние таких основных характеристик сети состоит в том, что в зависимости от конкретной топологии это может увеличить насыщенность, что приводит к очередям и дисперсии.

Наконец, и в последнем эксперименте мы рассмотрим влияние маршрутизации на обучающие возможности. В таблице 2.1 показана ошибка

обучения при использовании четырех разных политик маршрутизации в безмасштабной сети с 15 узлами и интенсивностью трафика макс. = 0.5. Для этого мы рассмотрим четыре разные политики маршрутизации: SP1 и SP2 соответствуют двум различным политикам маршрутизации, которые следуют подходу кратчайшего пути, MAN соответствует политике маршрутизации, разработанной вручную, которая может не следовать кратчайшему пути подход, но пытается загрузить баланс использования ссылок. Наконец, конфигурация «POOR» соответствует намеренно плохой конфигурации производительности, в которой несколько ссылок насыщены.

В этом случае мы наблюдаем, что в трех первых сценариях оба регрессора снова работают хорошо и что, как и в предыдущих случаях, они соответствуют дисперсии. Причина в том, что интенсивность движения низкая, и, как таковая, дисперсия также низкая. Обратите внимание, что политика маршрутизации «POOR» работает хуже, поскольку узкие места передаются по нескольким каналам, что создает очереди и увеличивает дисперсию. В этом случае нейронная сеть способна соответствовать дисперсии, в то время как полиномиальные регрессоры работают плохо.

Из этого эксперимента мы устанавливаем, что политика маршрутизации не влияет на точность оценок, за исключением случаев, когда она вызывает организацию очереди, что приводит к дисперсии и более высоким ошибкам.

Далее обсудим наши экспериментальные результаты, чтобы лучше понять использование нейронных сетей для моделирования компьютерных сетей. Учитывая высокую вычислительную стоимость экспериментов, описанных в этой главе, наши результаты ограничены сетями из 15 узлов, однако мы можем извлечь несколько важных уроков:

Нейронные сети могут точно моделировать среднюю сквозную задержку как функцию матрицы входного трафика для рассматриваемых сценариев: во всех случаях нейронные сети дают отличную производительность. Основная причина этого заключается в том, что для механизмов маршрутизации и пересылки, рассматриваемых в экспериментах, сети являются детерминированными сложными системами с памятью и могут быть изучены с незначительной ошибкой. В результате нейронные сети должны рассматриваться как актуальный инструмент в области сетевого моделирования.

Основным источником ошибок нейронных сетей является дисперсия набора данных: задержка на пакет является случайным процессом, и оценка его среднего значения с конечным количеством выборок приводит к ошибке, которая распространяется на ошибку обучения регрессоров. Эта ошибка может быть уменьшена путем использования большего количества выборок, но более длительное время измерения может привести к потере стационарного состояния измерения. Самоподобный трафик также создает важные проблемы для уменьшения ошибки измерения.

Эта ошибка пропорциональна дисперсии процесса задержки, которая, в свою очередь, зависит от количества очередей, понесенных пакетами. Хорошо



известно, что организация очереди зависит от множества факторов: интенсивности трафика, маршрутизации и т. д. Это справедливо не только для задержки, но и для любого случайного процесса, связанного с сетями.

Исходя из этого, мы заключаем, что основным источником ошибок при моделировании нейронных сетей является внутренняя ошибка оценки случайного процесса с конечными выборками. Для частного случая задержки это связано с очередью. Мы обнаружили, что ошибка подгонки незначительна, и мы не выявили никаких других источников ошибок в наших экспериментах. Кроме того, мы не обнаружили какого-либо неотъемлемого влияния основных характеристик сети (размер, топология, маршрутизация и характеристики трафика) на точность нейронных сетей, за исключением случаев, когда они приводят к дополнительным очередям.

Высокая интенсивность трафика требует более глубоких нейронных сетей: полиномиальные регрессоры привели к хорошей производительности, за исключением случаев, когда интенсивность трафика высока ( $\max > 1$ ). Мы нашли аналогичные результаты с нейронными сетями, которые используют один слой. Следовательно, сети, работающие вблизи насыщения, требуют более сложных регрессоров. Это говорит о том, что насыщенные сети имеют более сложные функции, которые требуют более глубоких нейронных сетей.

Вывод: среднюю сквозную задержку в сетях связи можно точно смоделировать с помощью нейронных сетей. Мы обнаружили, что нейронные сети работают замечательно хорошо с незначительной ошибкой, близкой к неснижаемой ошибке измерения.

Конечная цель этой работы - заложить основы для применения новых методов ML в области сетевого моделирования. Будущая работа включает в себя применение этих методов в реальных сценариях с реальным трафиком и протоколами и разработку способов представления сетевых данных (разработка функций).

### **2.1.3 Прогнозирование и обеспечение пропускной способности**

Одной из ключевых причин прогнозирования трафика является улучшение методов обеспечения динамической пропускной способности. С очень долгого времени исследователи работали над лучшим использованием пропускной способности. В начале 90-х годов были предложены оптимальные методы управления пропускной способностью для обеспечения эффективного управления полосой пропускания в виртуальном тракте (VP), используемом в методике асинхронного режима передачи (ATM). В этой статье авторы предложили стратегию, в которой пропускную способность VP можно проверять и изменять в течение определенного периода времени, чтобы контролировать потери пропускной способности, а также минимизировать затраты. Процесс прибытия потока трафика сети ATM предполагался марковским процессом. Это предположение было также сделано в более ранней статье, где управление полосой пропускания в VP в сети ATM было улучшено с использованием статистического совместного использования

пропускной способности. В [22] методы распределения полосы пропускания в VP в широкополосных сетях были предложены с использованием политики, в которой учитывается соотношение между затратами и использованием полосы пропускания путем установки порога, предполагающего, что процесс поступления в сеть является марковским процессом. В [23] на основе модели Маркова предложена схема оценки пропускной способности с учетом будущего временного интервала. Их главная цель состояла в том, чтобы разработать политику по преодолению разрыва между задержкой и использованием. В [18] была предложена адаптируемая архитектура нейронной сети для онлайн-моделирования и интерактивного моделирования. Алгоритм оценки рекурсивного веса, разработанный авторами, обеспечивает обновление весов сети для получения выходных данных сети после адаптации, близких к текущим битрейтам. Сообщалось, что их алгоритм прост в вычислениях, и было обнаружено, что он работает лучше, чем предыдущие методы.

В [14] был предложен алгоритм динамического обеспечения полосы пропускания, основанный на их схеме прогнозирования. Авторы использовали модель на основе ARCH, модель временного ряда с условно-непостоянными дисперсиями и конечной безусловной дисперсией в асимптотической области. Метод прогнозирования, Модифицированное прогнозирование вероятности-скачок, был введен путем установки пределов вероятности на основе условного распределения прогноза. Две самые последние ошибки прогноза были учтены в прогнозе исходя из предположения, что ошибки отражают переходный характер динамики движения. Схема обеспечения полосы пропускания для прогнозирования использовалась для обеспечения динамического выделения полосы пропускания на основе значения полосы пропускания, полученного из системы прогнозирования, а также на основе требований к уровню обслуживания.

В [24] комбинация вейвлет-анализа с множественным разрешением и моделей линейных временных рядов использовалась для прогнозирования обновлений каналов в магистральной IP-сети и сообщила о существовании долгосрочных трендов и сильных периодичностей в магистральной IP-трафике. Вейвлет-анализ мультиразрешения был использован, чтобы сгладить траекторные измерения для определения долгосрочного тренда. После первоначального преобразования ARIMA низкого порядка использовалась для моделирования и прогнозирования долгосрочного тренда и изменчивости трафика. было установлено, что их прогноз дает точную доходность на следующие шесть месяцев.

Прогнозирование трафика не только позволяет исследователям разрабатывать схемы динамического распределения полосы пропускания, но также используется при обнаружении аномалий в сетях, о чем сообщалось в [25] и [26]. В [25] авторы разработали  $k$ -ary sketch, разновидность структуры данных эскиза, и реализовали схемы прогнозирования временных рядов, такие

как ARIMA и Holt-Winters, для обнаружения сетевых аномалий. Обнаружение аномалии было достигнуто путем мониторинга потоков с высокими ошибками прогноза. В [26] авторы разработали устойчивую к аномалиям нестационарную схему прогнозирования дорожного движения, которая способна обрабатывать единичные и непрерывные аномалии.

Нелинейная модель временных рядов ОАРУГ для моделирования и прогнозирования движения

Прогнозирование сетевого трафика играет очень важную роль во многих областях, таких как управление перегрузкой, адаптивные приложения, управление сетью и проектирование трафика. Хорошая модель движения должна обладать способностью улавливать характерные характеристики движения, такие как зависимость на больших расстояниях (ДЗ), самоподобие и распределения с тяжелыми хвостами. В этой статье мы предлагаем модель нелинейного временного ряда, Обобщенную авторегрессивную условную гетероскедастичность (ОАРУГ), с инновационным процессом, обобщенным на класс распределений с тяжелыми хвостами. Наша модель установлена на реальных данных, и наши результаты подтверждают правильность соответствия нашей модели. Затем мы оцениваем схему прогнозирования для прогноза нашей модели. Сравнительное исследование с другими существующими моделями показывает, что наша модель имеет лучшую точность прогнозирования. Кроме того, оценка параметров является менее сложной, чем другие модели, используемые до сих пор при моделировании данных интернет-трафика.

Моделирование трафика является сложной задачей, особенно при выборе модели, которая представляет сложные характеристики интернет-трафика. Это постоянное исследование, в котором многие исследователи работают более десяти лет. Это исследование постоянно развивается по мере увеличения объема трафика с развитием технологий и увеличением количества пользователей. Из-за различных факторов, таких как стоимость и эффективная производительность, сетевые поставщики завтра будут сосредоточены на эффективном способе распределения полосы пропускания. Исследователи работают над моделями для интернет-трафика, чтобы иметь возможность прогнозировать трафик, уменьшая потери и недостаточную пропускную способность.

В нашей работе мы пытаемся ответить на ключевой вопрос: можем ли мы определить модель, которая может представлять изменчивую природу интернет-трафика, и разработать простую методологию прогнозирования на основе идентифицированной модели? При ответе на эти вопросы следует учитывать две ключевые проблемы: 1) конкретные характеристики трафика и 2) желаемое выполнение прогноза. Определение хорошей модели является основным критерием, так как без эффективной модели прогноз или прогноз могут быть неточными. Чтобы идентифицировать модель, нам нужно использовать лучший способ оценки параметров. Как только модель определена и прогноз сделан, мы можем динамически распределить

пропускную способность на основе предыдущих ошибок прогноза. Хорошая модель должна быть в состоянии позаботиться о таких характеристиках трафика, как дальняя зависимость, самоподобие и распределение тяжелых хвостов. Единственный способ объяснить функциональность интернет-трафика и повысить его производительность - это модели, полученные из анализа сетевых измерений. После того, как модель определена, прогноз делается на основе оценочной модели.

Хорошая модель трафика играет очень важную роль даже для будущего Интернета, где виртуальные сети могут быть построены поверх той же физической инфраструктуры. Некоторые преимущества виртуализации могут быть даны как:

- виртуальные сети могут быть гибкими, чтобы адаптироваться к потребностям и поддерживаться;
- несколько виртуальных сетей могут быть созданы одновременно;
- виртуальные сети могут быть обеспечены изоляцией при необходимости;
- виртуальные сети могут обеспечить экономию масштаба и объема для всей сети.

Для такой виртуальной сети ресурсы могут быть выделены отдельному потоку или объединенному потоку. Первый шаг к получению распределения ресурсов состоит в том, чтобы получить измерения прошлого профиля движения потока. Соответствующая модель может быть приспособлена к наблюдаемым измерениям и обеспечить приемлемый прогноз для модели. Для каждого объекта виртуальной сети можно использовать алгоритм равномерного распределения для вычисления распределения, а также для интервала для распределения на основе прогноза. Затем вычисленное значение распределения может быть зарезервировано в ссылках для вычисленного интервала времени. Очень важным требованием для эффективного распределения ресурсов является возможность прогнозировать профиль скорости потока в будущих временных интервалах. Цель прогноза или прогноза состоит в том, чтобы избежать задержки и, следовательно, ненужного (резервирование ресурсов больше, чем текущий трафик) или недопустимого (резервирование ресурсов ниже, чем текущий трафик), распределения ресурсов.

Потребность в трафиковом моделировании и прогнозировании возникла с целью сетевых провайдеров, которые постоянно надеются предоставить клиентам динамически обеспечиваемую полосу пропускания на основе периодически измеряемых данных. Эта работа также очень сложная и многообещающая, поскольку она будет полезна как для сетевых провайдеров, так и для пользователей в отношении распределения пропускной способности. Пользователям не нужно беспокоиться о том, чтобы платить слишком много за их пропускную способность, а провайдерам сети не нужно беспокоиться о потере пропускной способности.

Нашей первой целью является разработка модели, которая наилучшим образом характеризовала бы динамику скорости передачи данных данного набора данных. Показано, что интернет-трафик демонстрирует всплески, где дисперсия не постоянна. Эта динамическая дисперсия Интернет-трафика может быть учтена моделью, которая учитывает эту условную дисперсию. Одной из таких моделей является модель Обобщенного авторегрессионного условного гетероскедастика (ОАРУГ), которая использовалась во многих анализах финансовых данных. Нашей главной целью в настоящее время является прогнозирование трафика движения для предлагаемой модели. Мы сохранили распределение пропускной способности в качестве нашей будущей работы.

Литература по моделям интернет-трафика богата моделями, основанными на линейных временных рядах, таких как Авторегрессивное скользящее среднее (ARMA), Авторегрессивное интегрированное скользящее среднее (ARIMA), Фракционное интегрированное авторегрессивное скользящее среднее (FARIMA). Было установлено, что модели временных рядов ARIMA подходят для моделирования нестационарной передачи данных, а также для моделирования изменяющихся во времени телефонных линий. Одна из проблем, связанных с оценками параметров ARIMA, заключается в том, что вполне возможно, что оценки параметров ARIMA могут изменяться в течение более длительного промежутка времени. Таким образом, с течением времени такие недостатки могут вызывать нежелательную корреляцию в нововведениях (с точки зрения шума) модели, что, в свою очередь, влияет на генерируемые прогнозы. Такие модели, как ARMA, ARIMA и FARIMA, не могут уловить нелинейное поведение интернет-трафика, хотя ARIMA заботится о нестационарном поведении трафика. Однако эти модели имеют постоянную дисперсию и, следовательно, не могут уловить взрывную природу интернет-трафика, что является очень важной характеристикой интернет-трафика.

Позднее из-за нелинейного поведения Internet Traffic были представлены модели нелинейных временных рядов. Нелинейная модель временных рядов, такая как пороговая авторегрессия (TAR), использовалась для моделирования данных Интернет-трафика и видео VBR-трафика. В [13] авторы разработали интегрированную схему прогнозирования, а также предложили метод для одновременного запуска нескольких предикторов и прогнозирования, который показал наименьшую ошибку прогнозирования. В [14] модель на основе ARCH использовалась для прогнозирования и адаптивного предоставления полосы пропускания. Несмотря на то, что авторы позаботились о том, чтобы учесть характеристики Интернет-трафика, такие как дистрибутивы процесса инноваций с большими хвостами, они не приняли во внимание взрывную природу Интернет-трафика, который на самом деле является очень важной характеристикой. Временные ряды - не единственные модели, предлагаемые для интернет-трафика. Например, Нечеткая логика и нейронные сети использовались для моделирования движения,

прогнозирования и контроля перегрузок. В [31] авторы использовали Neural Network Ensemble (NNE) для прогнозирования трафика TCP/IP с использованием методов прогнозирования временных рядов, таких как ARIMA и Holt-Winters.

Была проведена ограниченная работа, в ходе которой исследователи пытались понять динамизм сети и то, как она контролирует динамическое предоставление полосы пропускания. Большинство из этих работ основаны на предположении, что природа трафика известна. В [20],[21],[22] процесс поступления потока трафика для динамического управления виртуальным трактом, используемый в сетях ATM, основан на марковском предположении. Оценка пропускной способности на основе модели Маркова была предложена в [23] для будущего временного интервала, чтобы учесть изменяющийся во времени трафик в виртуальных каналах ATM. Кроме того, схемы адаптивного управления полосой пропускания были тщательно изучены в [18] в отношении зависящего от времени движения Пуассона с использованием точечной стационарной методики приближения потока жидкости. Хотя была проделана большая работа на основе предполагаемой модели, ни один из этих методов не основывался на периодически измеряемых данных для оценки полосы пропускания. Потребность в периодически измеряемых данных была мотивирована результатами 4, указывающими на отказ модели Пуассона, и результатами, показывающими самоподобные характеристики и зависимость на больших расстояниях в различных временных масштабах. Кроме того, прогнозирование движения было использовано для обнаружения аномалий в сетях, как показано в [25], [26].

В [14] была проделана большая работа по прогнозированию и динамическому обеспечению ширины полосы. Авторы использовали модель ARCH, которая представляет собой модель временного ряда с условно-непостоянными дисперсиями, но с конечной безусловной дисперсией в асимптотической области. Их модель использовала распределение Стюдента- $t$  для инновационного процесса, чтобы приспособиться к явлению с тяжелыми хвостами, а степень тяжести с хвостами контролируется числом степеней свободы. Это отклонение от традиционного предположения о нормальном распределении инновационного процесса. Они внедрили метод прогнозирования Модифицированное прогнозирование с вероятностно-скачком, в котором пределы вероятности устанавливаются на основе условного распределения прогноза. В прогнозе они рассмотрели две самые последние ошибки прогноза, основываясь на предположении, что ошибки отражают переходную природу трафика движения. Для достижения динамической инициализации полосы пропускания они использовали схему инициализации полосы пропускания прогнозирования, где полоса пропускания для будущего момента времени предоставляется на основе значения полосы пропускания, полученного из системы прогнозирования, а также на основе требования к уровню обслуживания.

Хотя предыдущий подход к прогнозированию и обеспечению полосы пропускания на основе ARCH (ARIMA-ARCH) является общим и может применяться к любому временному окну измерений, все же вполне возможно, что между двумя последовательными измерениями произойдет законный всплеск трафика для которого предоставление может не быть адекватным. Чтобы позаботиться об ограничениях модели ARCH, мы предлагаем модель ОАРУГ. Модель ОАРУГ является расширением модели, основанной на ARCH, для включения частей скользящего среднего. Модель на основе ARCH принимает средневзвешенное значение последних квадратов наблюдений в качестве аппроксимации текущей условной дисперсии. Зависимость от прошлых различий делает эту модель более подходящей для данных интернет-трафика, учитывая высокую интенсивность интернет-данных.

Для идентификации модели мы использовали программное обеспечение для моделирования временных рядов ITSM. Модель лучшего кандидата может быть выбрана с использованием информационных критериев Акаике (AIC) и байесовских критериев формирования информации (BIC). Следует подчеркнуть, что такие метрики, как AIC и BIC, не только оценивают соответствие между значениями, предсказанными моделью и фактическими измерениями, но и штрафуют модели с большим числом параметров. Как только порядок модели определен, параметры оцениваются с использованием Оценщика максимального правдоподобия (MLE), мощного и эффективного Оценщика. Прогнозирование будущего - одна из фундаментальных задач анализа временных рядов. Мы использовали простой метод одноэтапного прогнозирования, чтобы получить последующие значения для определенного временного окна. В будущем мы хотели бы разработать алгоритм динамического распределения полосы пропускания на основе алгоритма прогнозирования. Мы хотели бы разработать алгоритм для динамического распределения полосы пропускания на основе алгоритма прогноза в качестве будущей работы.

Модель ОАРУГ является расширением модели ARCH. Обобщенная модель авторегрессионного условного гетероскедастика (ОАРУГ) с порядком  $p \geq 0$  и  $q \geq 0$  определяется как:

$$Z(t) = \sqrt{h(t)}e(t), \quad (3)$$

$$h(t) = \alpha_0 + \sum_{i=1}^p \alpha_i Z_{t-i}^2 + \sum_{j=1}^q \beta_j h_{t-j}^2, \quad (4)$$

где  $e(t) \sim \text{IID}(0,1)$ ;

$\alpha_0 > 1$ ,  $\alpha_1 \geq 0$  и  $\beta_j \geq 0$  являются константами.

$$\sum_{i=1}^p \alpha_i + \sum_{j=1}^q \beta_j < 1, \quad (5)$$

$e(t)$  не зависит от  $Z_{t-k}$ ,  $k \geq 1$ .

Стохастический процесс  $Z_t$ , определенный с помощью приведенных выше уравнений, называется процессом ОАРУГ ( $p, q$ ). Здесь  $h_t$  - условная дисперсия  $Z_t$ , заданная  $\{Z_s, s < t\}$ , а  $e_t$  - переменная инновации (ошибки).

Определена модель авторегрессии условного гетероскедастика (ARCH) с порядком  $p \geq 0$

$$\begin{aligned} Z(t) &= \sqrt{h(t)}e(t), \\ h(t) &= \alpha_0 + \sum_{i=1}^p \alpha_i Z_{t-i}^2, \end{aligned} \quad (6)$$

где  $e(t) \sim \text{IID}(0,1)$ ;

$\alpha_0 > 1, \alpha_1 \geq 0$  являются константами, а  $e(t)$  не зависит от  $Z_{t-k}, k \geq 1$ .

Стохастический процесс  $Z_t$ , определенный уравнениями выше, называется а ARCH ( $p$ ) процесс. Здесь  $h_t$  - дисперсия  $Z_t$ , а  $e_t$  - инновационная переменная (ошибка).

Непостоянная дисперсия данных интернет-трафика может быть учтена моделью ОАРУГ, поскольку модель ОАРУГ заботится о динамической дисперсии. Распределения остатков, как было обнаружено, значительно отклоняются от нормального, то есть они распределены с тяжелыми хвостами. Чтобы приспособиться к распределению процессов с инновациями с большим хвостом, мы используем распределение Стьюдента- $t$ , где степень управления с большими хвостами может контролироваться количество степеней свободы. Это представляется следующим образом,

$$\sqrt{\frac{v}{v-2}} e(t) \sim t_v, \quad v > 2, \quad (7)$$

где  $t_v$  - распределение Стьюдента- $t$  с  $v$  степенями свободы.

Как только выбор порядка для нашей модели сделан, мы приступаем к оценке параметров  $\alpha$  и  $\beta$ . Поскольку наш модельный порядок равен  $(1,1)$ , нам просто нужно найти  $\alpha_0, \alpha_1$  и  $\beta_1$ . Методика оценки параметров, используемая в этой статье, является оценкой максимального правдоподобия (ОМП), мощной и эффективной оценкой. Отношение правдоподобия для ОАРУГ с распределением Стьюдента- $t$  определяется как:

$$L(\alpha_0, \dots, \alpha_p, \beta_1, \dots, \beta_q, v) = \prod_{t=p+1}^n \frac{\sigma_t^{-1} \sqrt{v}}{\sqrt{v-2}} t_v \left( \hat{Z}_t \frac{\sigma_t^{-1} \sqrt{v}}{\sqrt{v-2}} \right), \quad (8)$$

где  $\hat{Z}_t$  - средние поправки наблюдений в момент времени  $t$ ;

$\sigma_t$  - стандартное отклонение ( $\sigma_t = \sqrt{h_t}$  в момент времени  $t \geq 1$ ).



Максимизация проводится с учетом коэффициентов  $\alpha_0$ ,  $\alpha_1$  и  $\beta_1$  и степеней свободы  $\nu$  от  $t$  - плотности  $t_\nu$ .

Наборы данных для исследования ОАРУГ получены из реальных данных о трафике сети из базовой сети Abilene. Сеть Abilene - это высокопроизводительная магистральная сеть Интернет-2, соединяющая исследовательские институты для ускорения разработки современных интернет-приложений. Все каналы имеют пропускную способность 10 Гбит / с, в которой используется ОС-192. Для нашего анализа мы использовали данные за 24 часа, которые агрегируются каждые 5 минут, и мы рассмотрели три ссылки: связь Хьюстон-Канзас-Сити, связь Канзас-Сити-Денвер и совокупная трасса Абилина. Поскольку наша цель состоит в том, чтобы разработать модель, которая наилучшим образом соответствует Интернет-трафику, мы используем первые 24-часовые данные для подбора нашей модели и данные за следующие 24 часа для сравнения.

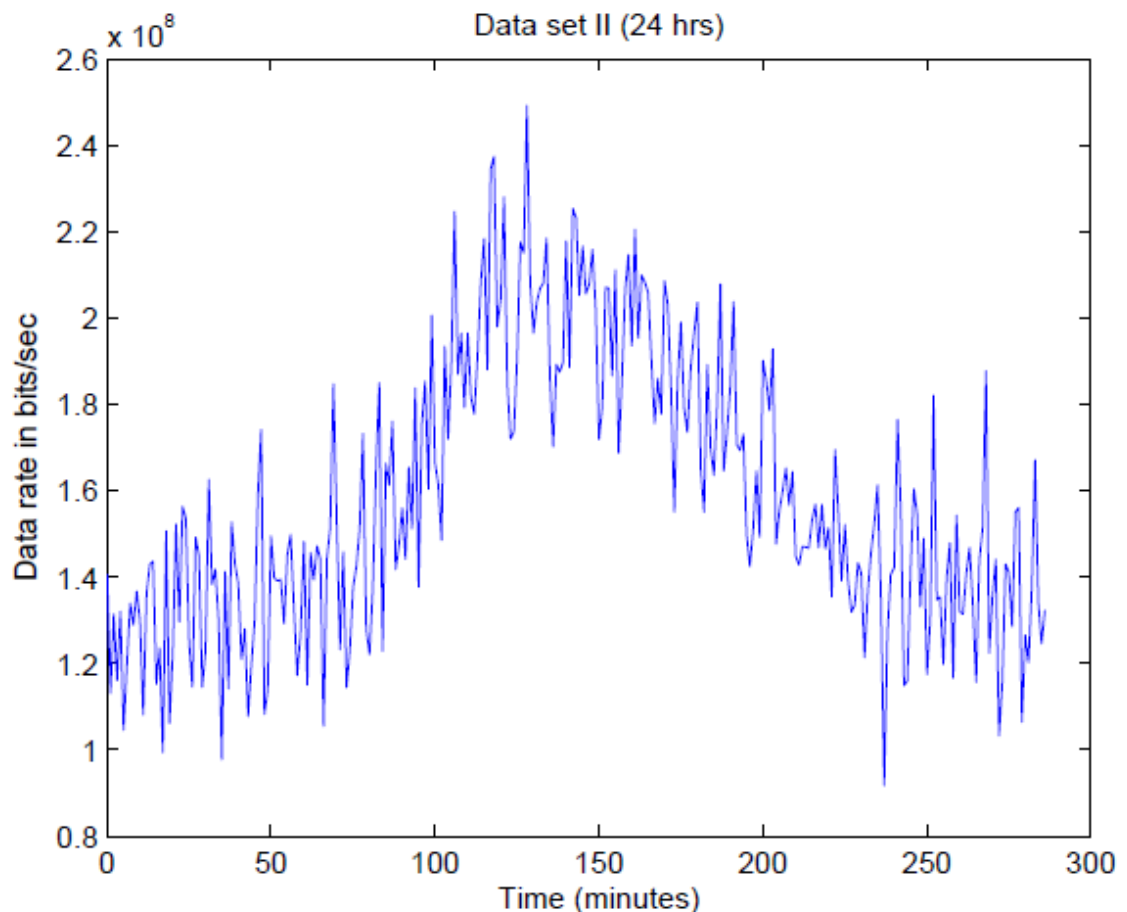


Рисунок 2.15 - График временных рядов набора данных II

наш прогноз с фактическим трафик. Временной ряд набора данных II приведен на рисунке 2.15.

Примерный график АСФ набора данных II приведен на рисунках 2.16. Пример графика АСФ показывает, что данные нестационарны. Чтобы

устранить нестационарность и тенденцию в данных, нам необходимо предварительно обработать данные. Агрегированные временные ряды преобразуются с использованием преобразования Бокса-Кокса и дифференцируются один раз. График преобразованных данных для набора данных II приведен на рисунке 2.17.

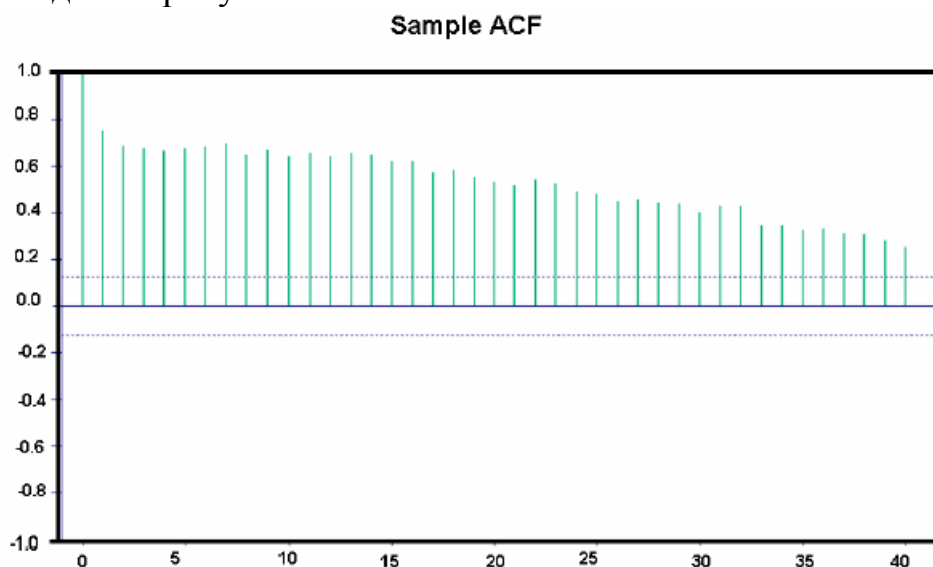


Рисунок 2.16 - Пример ACF набора данных II

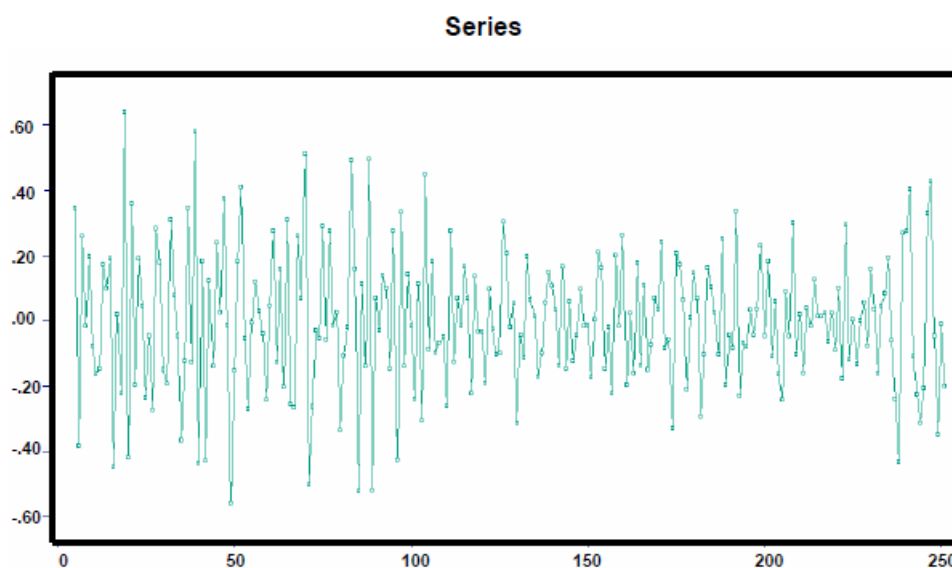


Рисунок 2.17 - Данные, преобразованные из первого разностного журнала, из набора данных II

Таблица 2.2 - Оценки параметров для наборов данных I, II и III

| Estimates  | Data set I | Data set II | Data set III |
|------------|------------|-------------|--------------|
| $\alpha_0$ | 0,0003     | 0,002       | 0,004310     |
| $\alpha_1$ | 0,2069     | 0,16982     | 0,122784     |
| $\beta_1$  | 0,1158     | 0,808       | 0,876969     |
| $\nu$      | 23         | 8           | 5            |

Для выбора заказов  $p$  и  $q$  мы использовали модифицированный информационный критерий Акаике (AICС). Исчерпывающий поиск выполняется для различных комбинаций  $p$  и  $q$  с использованием инструмента ITSM, и модель с минимальным AICС выбирается в качестве наилучшей модели-кандидата. Причина выбора AICС заключается в том, что такие метрики, как AIC и BIC, не только оценивают соответствие между значениями, прогнозируемыми моделью, и фактическими измерениями, но также штрафуют модели с большим числом параметров. Наилучший минимальный порядок  $p$  и  $q$  для нашей модели равен (1,1).

После предварительной обработки данных мы установили модель ОАРУГ (1,1). Неизвестные параметры модели  $\alpha_0$ ,  $\alpha_1$  и  $\beta_1$  и  $\nu$  оцениваются с использованием функции максимального правдоподобия. Оценки всех параметров вместе с  $\nu$ -значением для наборов данных I, II и III приведены в таблице 2.2. Можно отметить, что оценки  $\alpha_0$ ,  $\alpha_1$  и  $\beta_1$  и  $\nu$  являются статистически значимыми с нуля, что оправдывает модель ОАРУГ и ее сложность процесса инноваций. Чтобы оценить пригодность для модели ОАРУГ, мы нанесли Q-Q (t-распределенный) график остатков ОАРУГ.

Для прогнозирования мы использовали рекурсивный одностадийный прогноз для получения последующих значений. Первые 24 часа набора данных используются в качестве обучающей части для моделирования трафика движения; следующие 24 часа используются для выполнения прогноза и сравнения. В нашей методологии прогнозирования мы устанавливаем шаг прогнозирования равным единице. Основываясь на параметрах, оцененных по временным рядам с использованием модели ОАРУГ, а также на информации, полученной в последний момент времени данных временных рядов, мы приступаем к прогнозированию трафика на следующий момент времени. Мы обновляем данные о трафике каждый раз, когда нам доступен фактический трафик; и этот процесс рекурсивно выполняется в течение следующих 24 часов. Этот прогнозируемый трафик сравнивается с фактическими данными трафика, и мы сравниваем производительность на основе ошибки прогнозирования. Используемая методология прогнозирования также проста для выполнения в режиме реального времени.

Мы наблюдаем, что модель ОАРУГ очень хорошо соответствовала данным о трафике, и это подтверждается проверкой достоверности соответствия, которая дается графиком Q-Q с квадратной корреляцией  $R^2$ , близкой к 1, как показано на рисунке 2.17. Такие хорошо подобранные статистические данные являются надежной эмпирической основой для прогнозирования и обеспечения пропускной способности.

После того, как тест на пригодность сделан, мы сделали прогноз на следующие 24 часа. На рисунке 2.18 мы видим, что прогноз точно соответствует фактическому трафику. Наша методика прогноза показывает несколько случаев слабого и слабого прогноза, что является положительным

моментом. Чтобы подтвердить эффективность нашей модели, мы провели сравнительное исследование с другими моделями, такими как ARIMA (1,1,1), ARCH (1) и ARIMA (1,1,1) -ARCH (1). Мы использовали данные о трафике, чтобы соответствовать всем четырем моделям ARIMA, ARCH, ARIMA-ARCH и ОАРУГ, и было отмечено, что ОАРУГ имеет наилучшее соответствие. Фактически, если есть недооценка, это вызывает озабоченность, поскольку будет нераспределение полосы пропускания, что может привести к потере информации в результате ограничения полосы пропускания. Чтобы подтвердить эффективность и сложность нашей модели, мы провели сравнительное исследование с другими моделями, такими как ARIMA, ARCH и ARIMA-ARCH. Анализ этого сравнительного исследования приводится в следующих разделах.

В этом разделе мы представляем сравнительные результаты прогнозирования эффективности ОАРУГ с ARIMA, ARCH и ARIMA-ARCH. Мы использовали данные о трафике, чтобы соответствовать всем четырем моделям

Q-Q (t-распределение  $df = 7,203$ ) Остатки ОАРУГ  $R^2 = 0,985023$ .

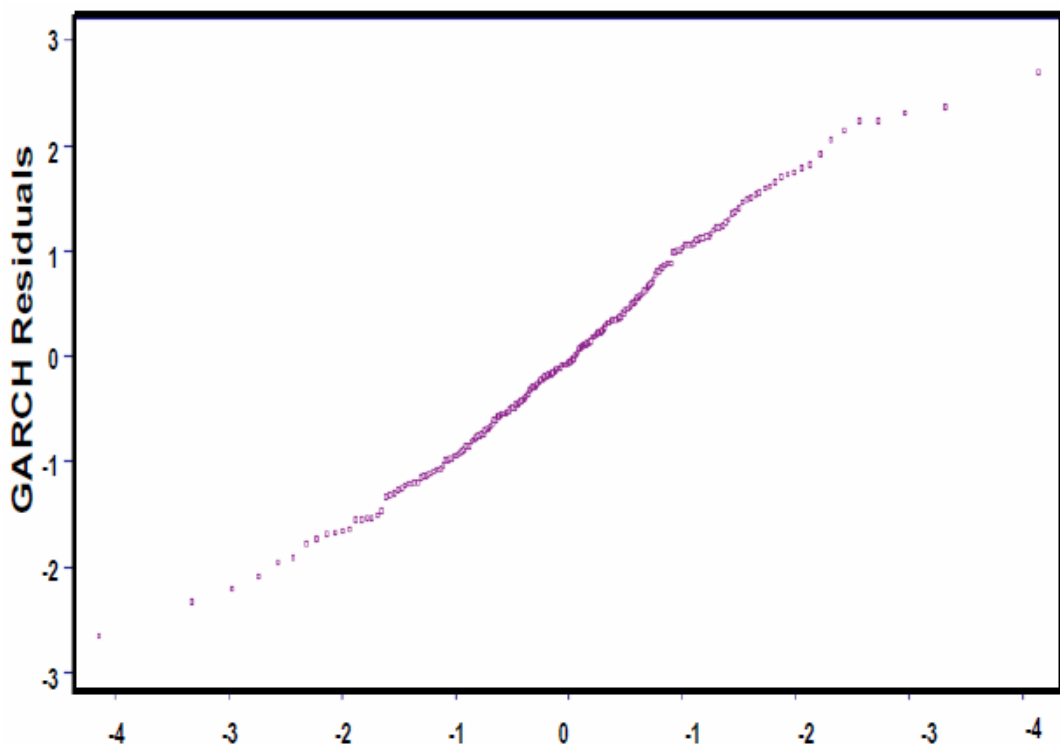
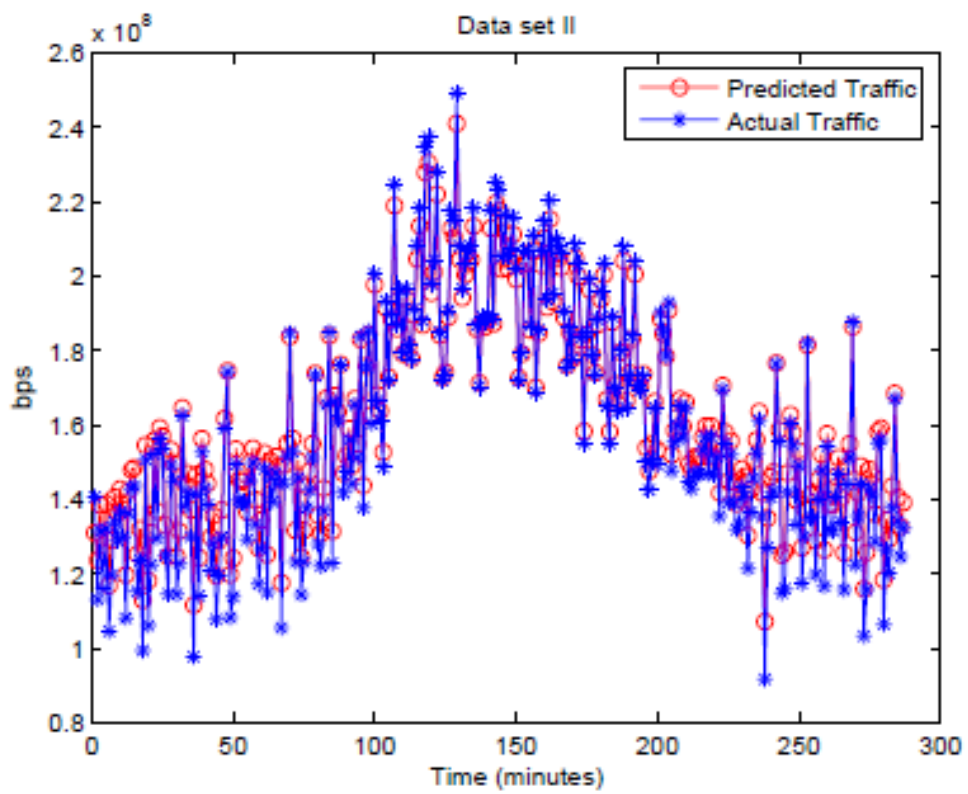
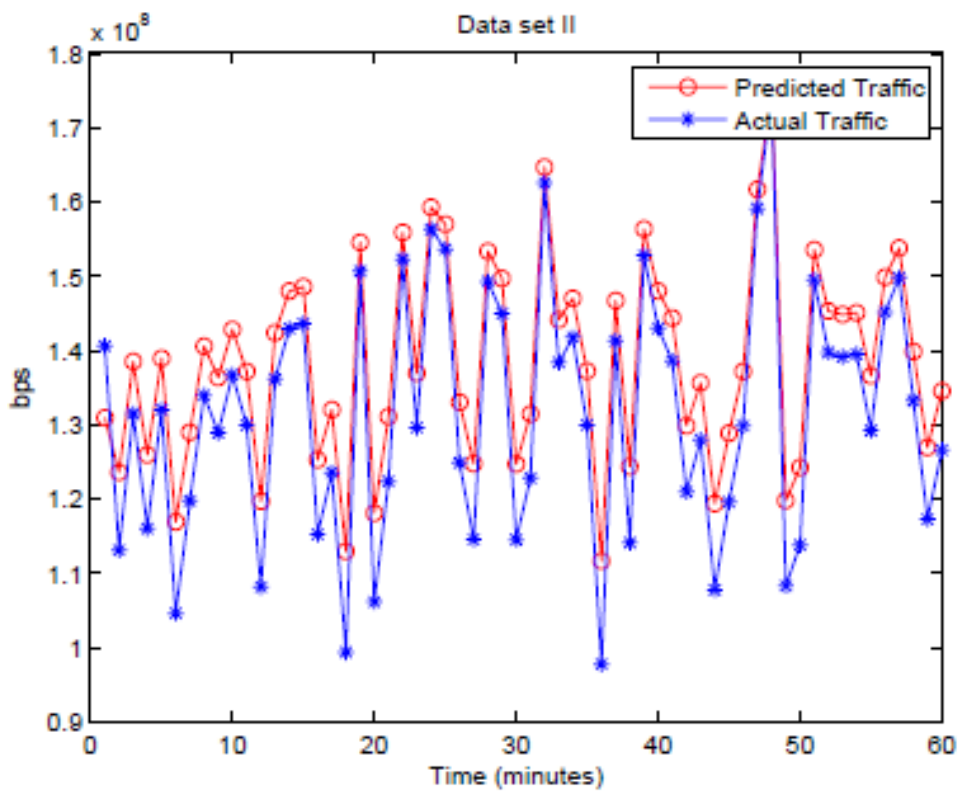


Рисунок 2.18 - Q-Q график остатков ОАРУГ для наборов данных II



(a)



(б)

Рисунок 2.19 - Прогнозируемая траектория для набора данных II (а) 24-часовой интервал (5-минутная агрегация) (б). Увеличенный вид (а) из интервала 0-60

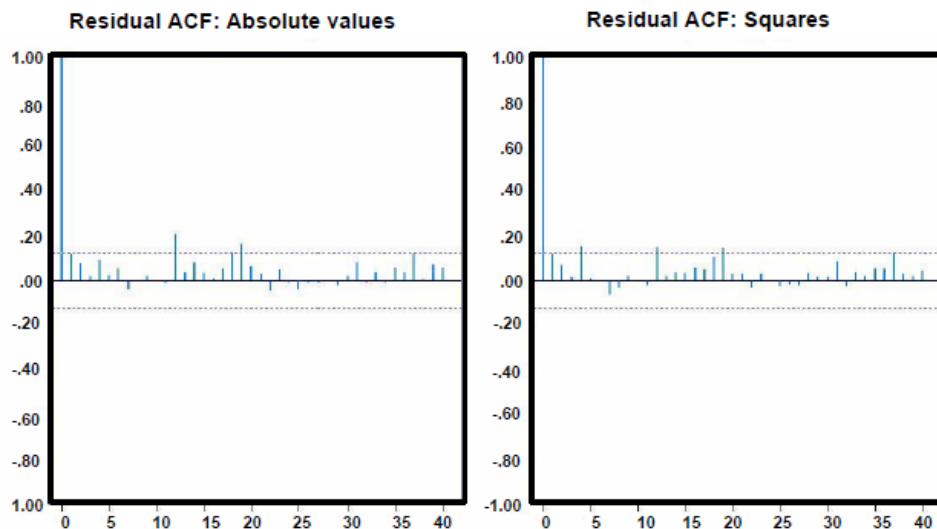


Рисунок 2.20 - График ACF абсолютного значения и квадратов модели ARIMA для набора данных I

ARIMA, ARCH, ARIMA-ARCH и OAPYГ и наблюдайте, насколько хорошо работают эти подогнанные модели. Чтобы проверить пригодность для этих моделей, мы наблюдали выборку ACF / PACF из остатков подобранной модели. Как можно видеть из рисунков 2.15, 2.16 и 2.17, остатки ARIMA, ARCH и ARIMA-ARCH для всех трех наборов данных значительно отличаются от нуля, в то время как в случае OAPYГ, как на рисунке 2.18, образец ACF имеет Остаток значительно близок к нулю. Эти результаты ясно показывают хорошие возможности производительности нашей модели. Наша модель способна точно соответствовать трафика по сравнению с другой моделью. Для дальнейшей проверки производительности нашей модели мы провели более детальное сравнительное исследование прогнозирующего поведения нашей модели и ARIMA-ARCH. На рисунке 2.19 (б) мы видим, что

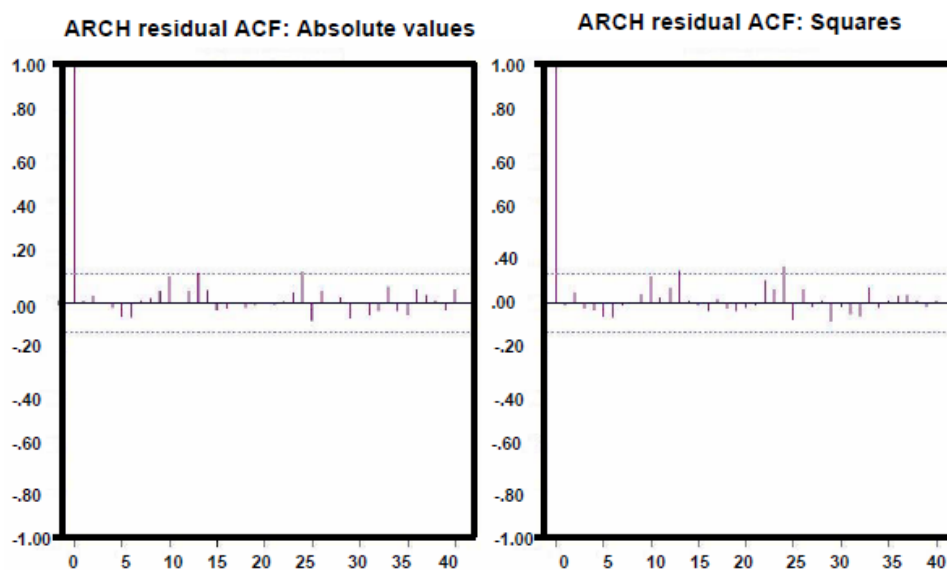


Рисунок 2.21 - График ACF абсолютного значения и квадратов невязок модели ARCH для набора данных I

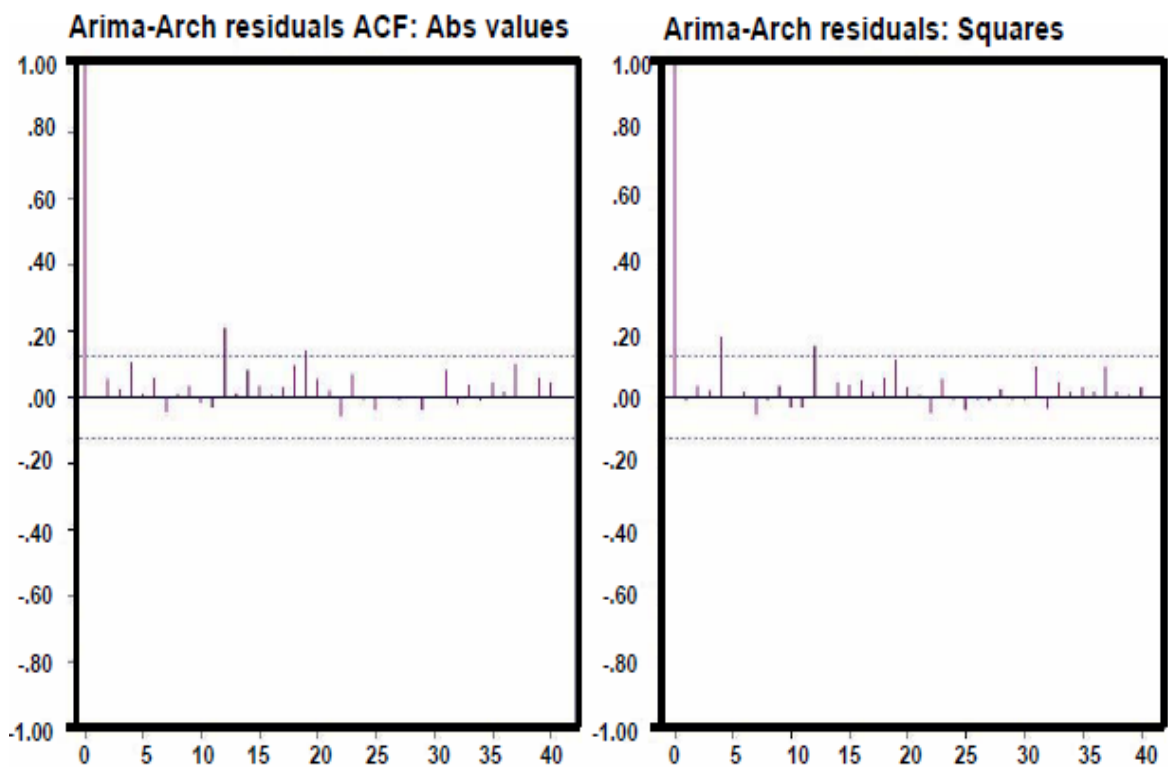


Рисунок 2.22 - График ACF абсолютного значения и квадратов модели ARIMA-ARCH для набора данных I

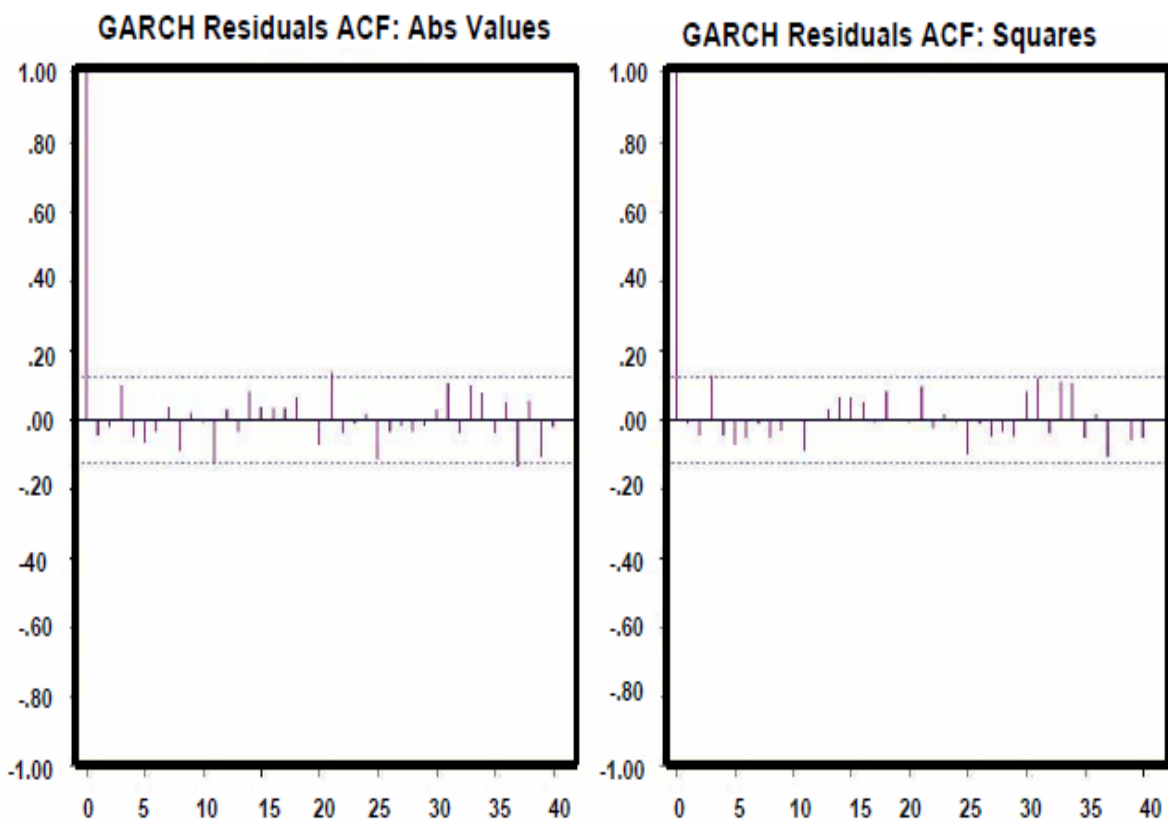


Рисунок 2.23 - График ACF абсолютного значения и квадратов модели ОАРУГ для набора данных I

Прогнозирующее поведение ARIMA-ARCH, хотя и близко следует за фактическим трафиком, в некоторых случаях оно недооценивает фактическое трафик. Это невыгодно, когда речь идет о распределении полосы пропускания, поскольку это может привести к строгому распределению полосы пропускания. В нашей модели нет заниженного прогноза и чрезмерного прогноза, что подходит для правильного распределения полосы пропускания. Но в случае ОАРУГ почти во всех трех наборах данных, которые мы изучали, прогнозирование немного превышает фактический трафик. Наша модель не прогнозируется и не слишком прогнозируется, что просто подходит для правильного распределения полосы пропускания. Мы также провели более детальное исследование для количественного сравнения производительности с учетом точности прогноза и сложности оценки параметров. Для измерения точности мы использовали следующие показатели:

$$- \text{Ошибка прогноза ОАРУГ } E_G = \frac{\sum_{i=1}^N (X_i - X_i^G)^2}{\sum_{i=1}^N (X_i)^2}; \quad (9)$$

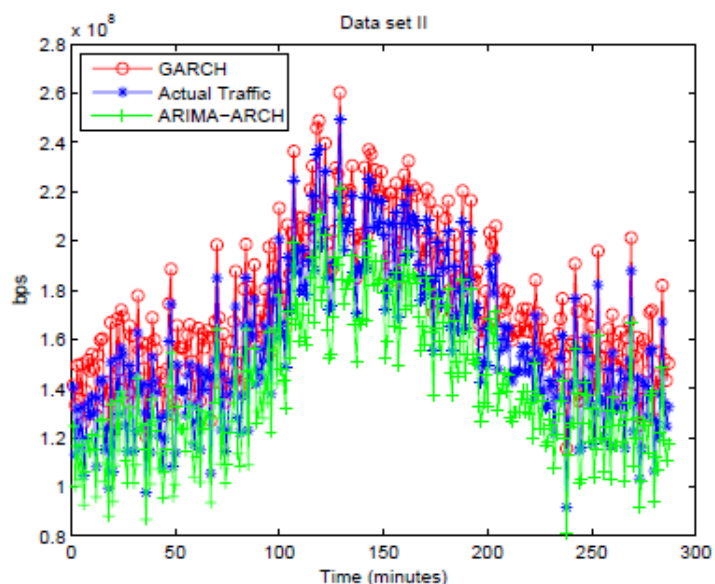
$$- \text{Ошибка прогноза ARCH } E_A = \frac{\sum_{i=1}^N (X_i - X_i^A)^2}{\sum_{i=1}^N (X_i)^2}. \quad (10)$$

где  $X_i - X_i^G$  - ошибка прогнозирования для модели ОАРУГ;  
 $X_i - X_i^A$  - ошибка прогнозирования для модели ARCH.

Чтобы измерить сложность, мы рассмотрели количество параметров для оценки в каждой модели. Далее мы оцениваем производительность и сложность нашей модели, используя вышеуказанные метрики. Для каждого набора данных мы используем первые 24-часовые данные для разработки модели временных рядов и рассматриваем следующие 24-часовые для оценки эффективности нашего подхода. Чтобы проверить производительность на более длительном интервале времени, мы применили алгоритм прогнозирования на период 7 дней. Как видно из рисунка 2.24, мы заметили, что прогнозируемый трафик близко следует за фактическим трафиком. Это очень хороший показатель того, что модель ОАРУГ способна хорошо работать даже в течение более длительных временных интервалов.

Чтобы оценить точность прогноза модели ОАРУГ, мы провели сравнительное исследование с существующей моделью ARIMA-ARCH. Как видно из таблицы 2.3, мы отметили, что прогноз ошибка модели ОАРУГ значительно меньше, чем модель ARIMA-ARCH, в которой четко указана эффективность прогноза ОАРУГ.





(a)

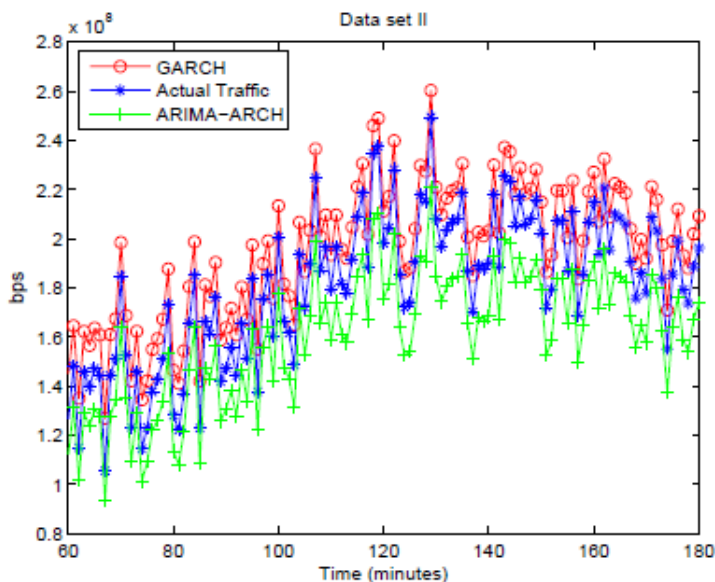


Рисунок 2.24 - Прогнозный анализ ОАРУГ и ARIMA-ARCH (a) 24-часовой интервал (5-минутная агрегация) (b). Увеличенный вид (a) из интервала 60-180

Наша модель ОАРУГ имеет уменьшенную сложность, характеризующуюся уменьшенным количеством:

Таблица 2.3 - Ошибки прогнозирования ОАРУГ и ARCH

| Error | Data set I               | Data set II              | Data set III            |
|-------|--------------------------|--------------------------|-------------------------|
| GARCH | $7,2652 \times 10^{-13}$ | $3,3393 \times 10^{-12}$ | $1,759 \times 10^{-10}$ |
| ARCH  | $1,1818 \times 10^{-4}$  | $1,8501 \times 10^{-5}$  | $2,0413 \times 10^{-4}$ |

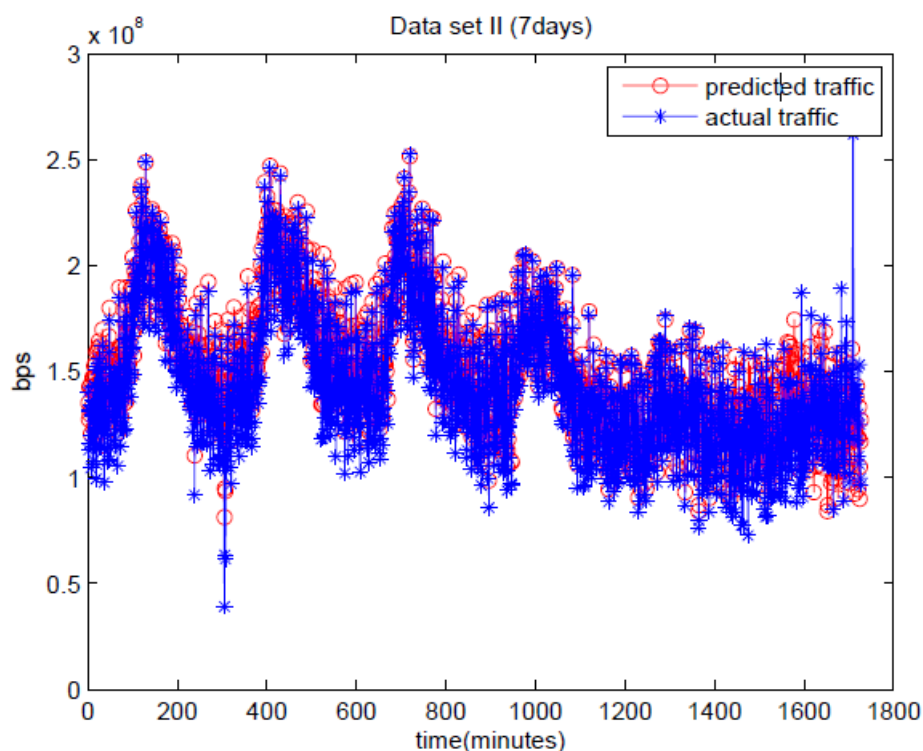


Рисунок 2.25 - Прогнозируемый трафик для набора данных II на период 7 дней

оцениваемые параметры, поскольку выбранная модель ОАРУГ имеет порядок (1,1), а выбранная модель ARIMA-ARCH имеет порядок (1,1,1).

Вывод: ОАРУГ - нелинейную модель временных рядов, которая способна фиксировать специфические характеристики данных интернет-трафика, которые не смогли вместить традиционные линейные временные ряды. Модель может уловить взрывную природу интернет-трафика с его переменной дисперсией. Статистически, если трафик имеет импульсное поведение, это означает, что он имеет дисперсию, изменяющуюся со временем. Модель ОАРУГ представляет собой нелинейную модель временного ряда, которая может эффективно улавливать условную дисперсию из-за ее зависимости от дисперсии в каждый момент времени. Модель способна прогнозировать агрегированный трафик, но дальнейшие исследования должны проводиться на менее агрегированном трафике. Поскольку мы могли успешно прогнозировать данные трафика, используя простую методологию прогнозирования, основанную на нашей модели ОАРУГ, мы также намерены разработать алгоритм динамического распределения полосы пропускания в качестве будущей работы.

### **3 Предлагаемая методология прогнозирования на основе искусственных нейронных сетей**

*Оценка доступности канала.*

Для обнаружения доступности используем формулу (11).

Предположим, (I) есть «i» количество задач и «j» количество мобильных узлов, (II) каждая задача использует полосу пропускания  $BW_T$ , а поставщик услуг имеет фиксированную полосу пропускания, и (III) фиксированная величина полосы пропускания равна  $BW_T$ , и среднее использование  $BW_A$  позволяет определить доступность канала как:

$$AV_{ch} = \frac{BW_T - \sum_{j=0}^j MB_j (\sum_{i=0}^i T_i)}{BW_{AV}}, \quad (11)$$

где  $AV_{ch}$  - канал доступен или нет;

$BW_T$  - общая пропускная способность, доступная на базе станция;

$MB_j$  - мобильные узлы, где  $j = 0$  до  $n$ , где  $n$  - максимальное количество мобильных узлов;

$T_i$  - общая пропускная способность, необходимая для задачи или услуги для одного мобильного узла, где  $i = 0$  до  $n$ ;

$BW_{AV}$  - это средняя пропускная способность для мобильного узла.

### 3.1 Решение для высокой мобильности сети

На первом этапе перемещение мобильного устройства из одной сети в другую сеть называется передачей обслуживания мобильного узла или устройства между базовыми станциями (мобильность вдоль сот). Во время передачи обслуживания QoS ухудшается до более низкого значения. В настоящее время здесь предлагается выбрать подходящую базовую станцию или подходящую ячейку для следующей зоны обслуживания. В этой нейронной сети есть шесть входов ( $X_1$ - $X_6$ ) или шесть значений в обученном наборе. Каждое значение обученного набора связано с доступностью канала базовой станции сот вокруг текущей ячейки. Детали метода определения доступности канала были предложены в статье [27]. Здесь доступность канала базовой станции соседней соты представляет собой входы или обученный набор для нейронной сети. Выход нейронной сети организован, так как наибольшее число каналов доступности является основным (предположим,  $X_1$ ), а другие являются дополнительным значением ( $X_2$ - $X_6$ ). Основное значение (доступность канала  $X_1$ , имеющая базовую станцию) выбирается в качестве следующего поставщика услуг для мобильного устройства. Скрытые слои или веса зависят от оборудования или услуг, доступных в этой ячейке. Доступность объекта связана с потоковым видео, интернетом и видеозвонками.

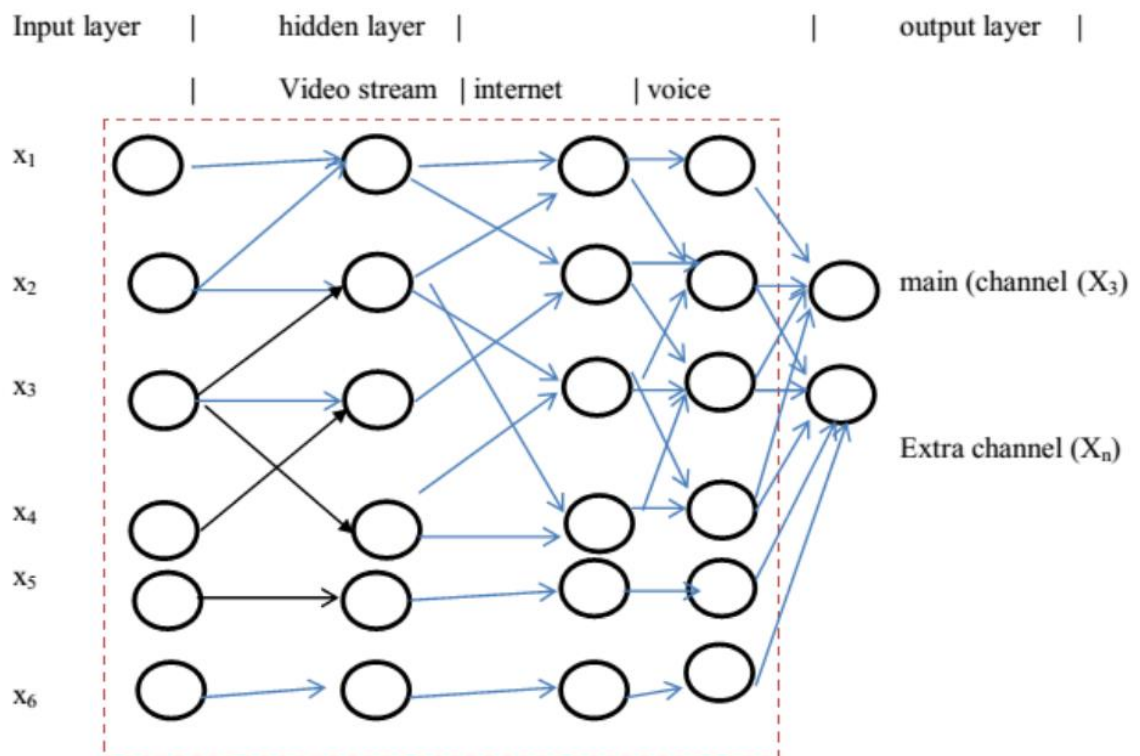


Рисунок 3.1 - Процедура нейронной сети на базовой станции, фаза -1

На рисунке 3.1 показано, что в нейронной сети есть шесть входов. Входные данные в обученных наборах связаны с доступностью канала шести соседних сотовых провайдеров. Доступность канала упоминается как  $X_1$ ,  $X_2$ ,  $X_3$ ,  $X_4$ ,  $X_5$  и  $X_6$ . Для обеспечения системы взвешивания рассматриваются три скрытых слоя. Слои - это видео поток, интернет, голос как  $W_{1i}$ ,  $W_{2i}$ ,  $W_{3i}$  соответственно. Таблица принятия решений, присутствующая на базовой станции для выбора следующей соты для предоставления услуги мобильному узлу, приведена ниже.

Таблица 3.1 - Анализ результатов для условий высокой мобильности

| Inputs | Weighatage-1 | Weighatage-1 | Weighatage-1 | Output | Selected |
|--------|--------------|--------------|--------------|--------|----------|
| $X_1$  | .....        | .....        | .....        | .....  |          |
| $X_2$  | .....        | .....        | .....        | .....  |          |
| $X_3$  | .....        | .....        | .....        | .....  | $X_3$    |
| $X_4$  | ....         | .....        | .....        | .....  |          |
| $X_5$  | ....         | .....        | .....        | .....  |          |
| $X_6$  | .....        | .....        | .....        | .....  |          |

Таблица 3.1 имеет шесть атрибутов. Атрибуты представляют собой входные данные, которые являются доступностью канала соседних ячеек, атрибуты со второго по четвертый представляют собой весовые коэффициенты, применяемые в нейронных сетях, а пятый показывает

выходные данные из соответствующей нейронной сети, а шестой - выбранный узел.

Алгоритм для фазы 1:

- шаг 1: Разработать нейронную сеть, как предложено в [27];
- шаг 2: Предоставить шесть входных данных в качестве обученного набора, который состоит из доступных каналов шести соседних ячеек;
- шаг 3: Назначить вес на звене нейронной сети как выбор пользователя, который зависит от времени;
- шаг 4: Получить канал, доступный как выход нейронной сети. Выберите ячейку, имеющую пропускную способность канала для следующей связи.

Рассмотрим пример. Вокруг зоны одного поставщика услуг есть шесть ячеек. Вокруг соты шесть соседних сот. Предположим, что поставщик услуг предоставляет услуги голосового вызова с весом 10, интернет с весом 7 и видеопоток с весом 5 на мобильный узел. Во время первого этапа доступность каналов шести станций составляет 5, 6, 8, 9, 3 и 7; их соответствующие ячейки А, В, С, D, Е и F.

Здесь этот метод рассматривает простые не перекрывающиеся нейроны для гладкого вычисления.

Функции на скрытом слое определяются как:

$$A_j(x,w) = \sum_{i=0}^n x_i \cdot W_{ij}, \quad (12)$$

где  $A_j(w,x)$  - выходное значение;

$X_i$  - входной обученный набор;

$W_{ij}$  - веса.

Согласно приведенной формуле (12) соответствующие значения будут рассчитываться следующим образом:

$$A-5 \times 10 + 5 \times 7 + 5 \times 5 = 50 + 35 + 25 = 110;$$

$$B-6 \times 10 + 6 \times 7 + 6 \times 5 = 60 + 42 + 30 = 132;$$

$$C-8 \times 10 + 8 \times 7 + 8 \times 5 = 80 + 56 + 40 = 176;$$

$$D-9 \times 10 + 9 \times 7 + 9 \times 5 = 90 + 63 + 45 = 198;$$

$$E-3 \times 10 + 3 \times 7 + 3 \times 5 = 30 + 21 + 15 = 66;$$

$$F-7 \times 10 + 7 \times 7 + 7 \times 5 = 70 + 49 + 35 = 154.$$

Если рассмотрено вышеупомянутое условие, наиболее значимым выходным сигналом будет ячейка D. Таким образом, следующим поставщиком услуг будет D. Этот метод может использовать кросс-нейронные сети в качестве мобильного устройства и получать обслуживание.

Этот тип процедуры адаптирован и на втором этапе.

### 3.2 Этап планирования задач

На втором этапе все задачи ( $T_1$ - $T_n$ ) применяются в качестве обученного набора для нейронной сети на базовой станции сети. Задачи ( $T$ ) - это услуги, предоставляемые базовой станцией конкретному мобильному устройству. Этот метод использует серию нейронных сетей и три скрытых слоя в нейронной сети для качества обслуживания. Скрытые уровни - это дрожание ( $W_{11}$ ), пропускная способность ( $W_{22}$ ) и потеря пакетов ( $W_{33}$ ). Он использует итерационный процесс нейронной сети для поиска задач в порядке убывания (от более высокого приоритета к более низкому приоритету, который выводится в качестве основной задачи из выходных данных первой нейронной сети в производную основную задачу или из выходных данных последней нейронной сети). Результат предыдущей нейронной сети или выходные данные остаются в стороне, а другие задачи задаются как обученный набор для второй нейронной сети.

Скрытые слои или веса остаются такими же ( $W_{11}$ ,  $W_{22}$  и  $W_{33}$ ), что и первая нейронная сеть. Результат или вывод второй нейронной сети рассматривается как вторая основная задача. Эта процедура продолжается до последней задачи ( $n$ -ой), являющейся главной задачей нейронной сети ( $n$ -ой). При этом порядок выполнения задания от одного до  $n$ -го задерживается.

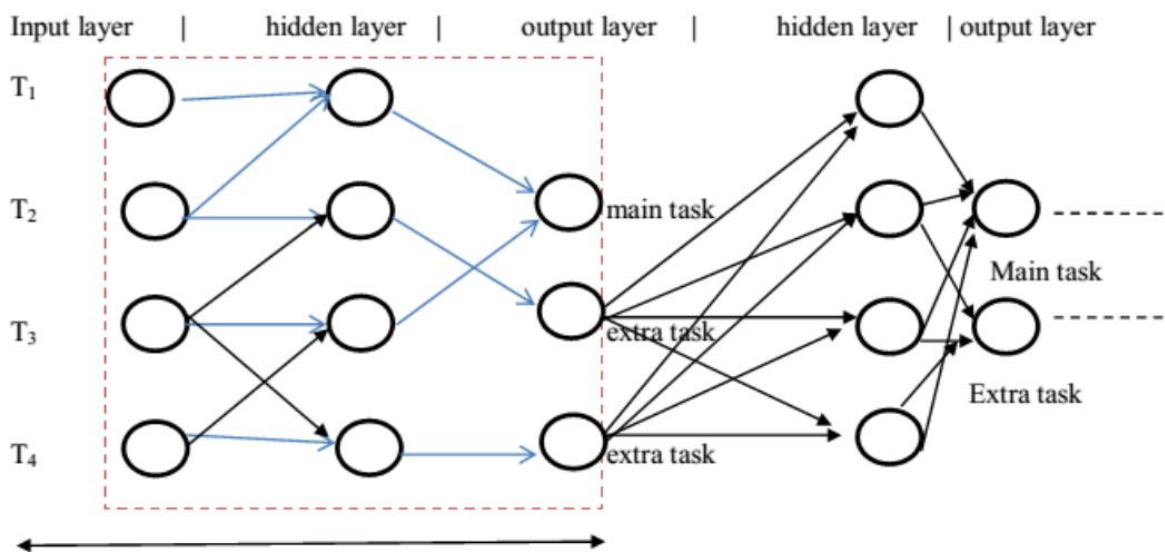


Рисунок 3.2 - Архитектура нейронных сетей на втором этапе

Диаграмма итеративной нейронной сети показывает, что существует четыре задачи. Задачи определены как  $T_1$ ,  $T_2$ ,  $T_3$  и  $T_4$ . Все задачи рассматриваются как обученный набор для нейронной сети.

Весы присутствуют на нейронах нейронной сети. Детали о скрытом слое (состоит из нейронов и функций) описаны на следующей диаграмме.

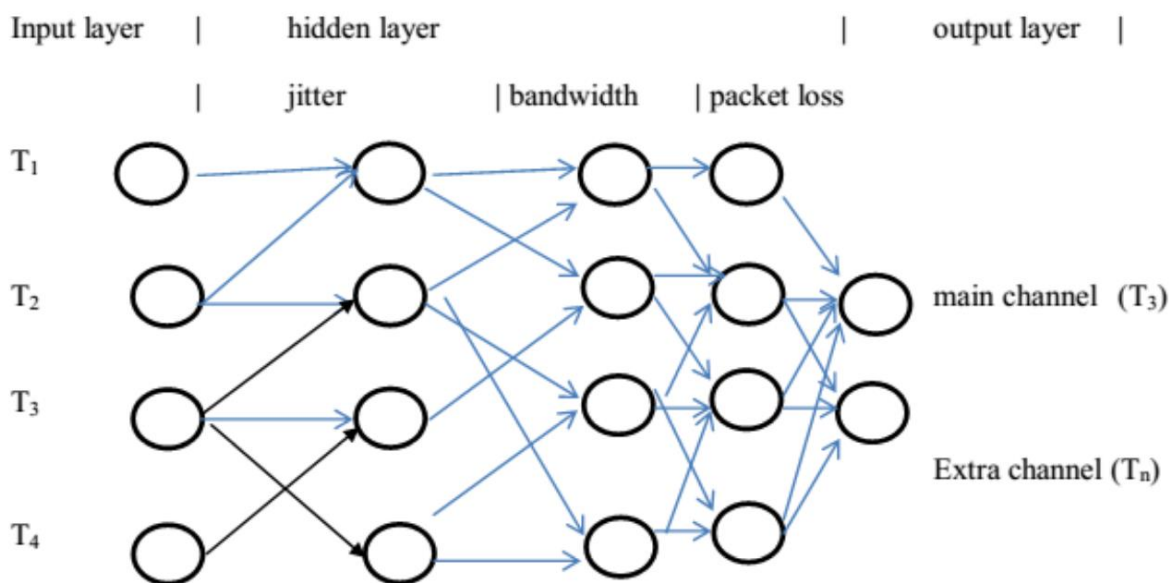


Рисунок 3.3 - Скрытый слой присутствует для планирования задач

На приведенном выше рисунке показано, что поставщик услуг предоставляет четыре задачи или услуги, которые задаются как обученный набор или ввод в нейронную сеть.

Весовые коэффициенты приведены по таким параметрам, как джиттер, пропускная способность и потеря пакетов. Эти параметры зависят от задач, поскольку задачи, присутствующие в сети, имеют свои собственные требования к этим параметрам. Расчет функции приведен в уравнении (12).

Таблица 3.2 - Результат нейронной сети в приоритетных задачах

| Serial number | Neural network number | Task identity |
|---------------|-----------------------|---------------|
| 1             | 1 <sup>st</sup>       | $T_1$         |
| 2             | 2 <sup>nd</sup>       | $T_2$         |
| 3             | 3 <sup>rd</sup>       | $T_3$         |
| ...           | ....                  | .....         |
| ...           | ....                  | .....         |
| n             | n <sup>th</sup>       | $T_n$         |

Таблица имеет три атрибута. Первый атрибут - это индекс таблицы или серийный номер. Второй атрибут - это количество примененных нейронных сетей, а третий - выбранный выход из соответствующей нейронной сети. Здесь выходные данные фазы представляют собой задачи  $T_1$ - $T_n$  в порядке сортировки.

Алгоритм для второго этапа:

– шаг 1: Установить все задачи (каждая задача имеет идентификатор задачи), предоставляемые сетью, в качестве обученного набора на нейронной сети;

- шаг 2: Сначала все задачи предоставляются в виде обученного набора для нейронной сети;
- шаг 3: Выход или наивысший выход нейронной сети, рассматриваемый как первое задание (наивысший приоритет) для мобильного устройства;
- шаг 4: Затем весь набор задач, кроме первого, применяется в качестве обученного набора для следующей нейронной сети. Нейронная сеть такая же, как первая нейронная сеть;
- шаг 5: Вторая основная задача рассматривается как вторая по важности. Шаг повторяется. Эта процедура продолжается до тех пор, пока не осталось последнее задание.

### 3.3 Этап оценки пропускной способности

На третьем этапе подходящая методика оценки полосы пропускания применяется к производному решению на первом и втором этапах. Формула для применения входных задач была обнаружена и способна рассчитать точный объем полосы пропускания, необходимый после планирования задач с учетом высокого коэффициента мобильности. Для достижения этого существует два предопределенных значения  $BW_{AV}$  и  $D_{T0}$ , определяемых пользователем перед применением последней фазы или текущей фазы.

Для определения пропускной способности,

$$BW_R = C \cdot D_{Tr} \cdot BW_{AV} \cdot \frac{1}{D_{T0}}, \quad (13)$$

где  $c$  – константа;

$D_{Tr}$  - входной обученный набор (задание, полученное на этапе 2);

$BW_{AV}$  - ширина полосы выбранного канала;

$D_{T0}$  - максимальное количество входов;

$BW_R$  - полоса пропускания, назначенная для конкретной задачи конкретного мобильного узла.

Алгоритм для третьей фазы

- шаг 1: Получить правильную задачу или набор данных из второго этапа;
- шаг 2: Получить полосу пропускания канала, доступного от базовой станции;
- шаг 3: Получить максимальное количество входов передачи.
- шаг 4: Получить ширину полосы канала, как показано в формуле (3) [2].

Интегрированный алгоритм.

- шаг 1: Поставщик услуг определяет, находится ли мобильное устройство в сценарии передачи обслуживания или нет. Если он не находится в сценарии передачи обслуживания, он непосредственно переходит к этапу 3.



Если мобильный узел находится в состоянии передачи обслуживания, он переходит к шагу 2;

– шаг 2: Примените доступность канала шести соседних ячеек, как указано в первой фазе. Затем перейдите на второй этап;

– шаг 3: На этом этапе задачи или услуги предоставляются базовой станцией мобильному узлу и обрабатывают их в итерационной логике нейронной сети, как указано на втором этапе;

– шаг 4: Получить полосу пропускания, указанную на третьем этапе;

– шаг 5: Пользуйтесь мобильным устройством, используя выбранные задачи. Если передача обслуживания присутствует, выбранный поставщик услуг свяжется с мобильным устройством.

### 3.4 Имитационная модель и параметры

В этом разделе мы моделируем предлагаемое нами предсказание сетевого трафика на основе нейронной сети (NTP-NN) в мобильной сети. Используемым инструментом моделирования является NS-2 [29], который представляет собой инструмент моделирования общего назначения, который обеспечивает моделирование дискретных событий пользовательских сетей. При моделировании мобильные узлы перемещаются в прямоугольной области 800 x 800 м в течение 30 секунд. Начальные местоположения и движения узлов получены с помощью модели произвольной точки пути (RWP) NS2. Все узлы имеют одинаковую дальность передачи 250 метров. Топология моделирования для эксперимента показана на рисунке 3.4.

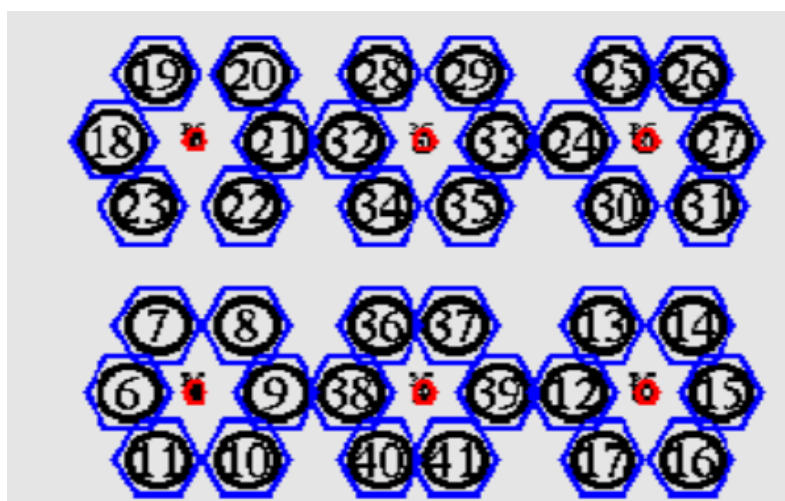


Рисунок 3.4 - Топология моделирования перед передачей обслуживания

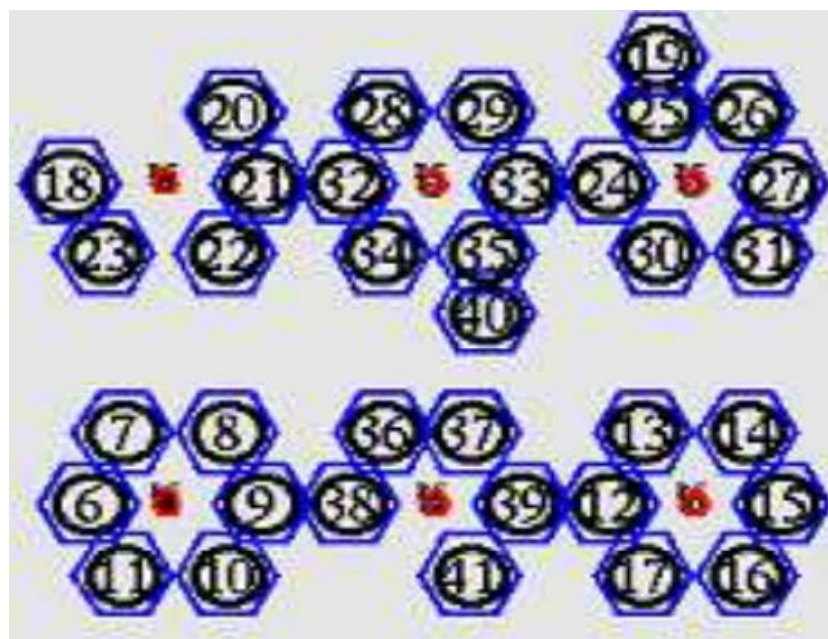


Рисунок: 3.5 - Топология моделирования после передачи обслуживания

Параметры моделирования обобщены в таблице 3.3.

Таблица 3.3 - Параметры моделирования

|                              |              |
|------------------------------|--------------|
| Количество узлов             | 36           |
| Количество сот               | 6            |
| Пользователи на соту         | 6            |
| MAC layer                    | Mac/Hsdpa    |
| Антенна                      | Omni Antenna |
| Размер пакета                | 500bytes     |
| Площадь симуляции            | 800x800 m    |
| Мощность передатчика         | 0.66 w       |
| Чувствительность передатчика | 0.395 w      |
| Протокол маршрутизации       | NOAH         |
| Модель трафика               | CBR          |
| Начальная энергия            | 4.1 J        |
| Время симуляции              | 30 Sec       |
| Скорость перемещения         | 25 m/s       |

## Результаты моделирования

### А. На основании скорости

В нашем первом эксперименте мы меняем отправку данных оценивается как 100, 200... 400 Кб.

На рисунке 3.6 показана максимальная измеренная ширина полосы. Из рисунка, наш предложенный NTP-NN достигает большой пропускной способности. На рисунке 3.7 показана средняя измеренная задержка. Из рисунка видно, что наша схема NTP-NN достигает меньшей сквозной

задержки. Из рисунка 3.8 мы можем наблюдать, что наш предложенный NTP-NN имеет низкое среднее энергопотребление.

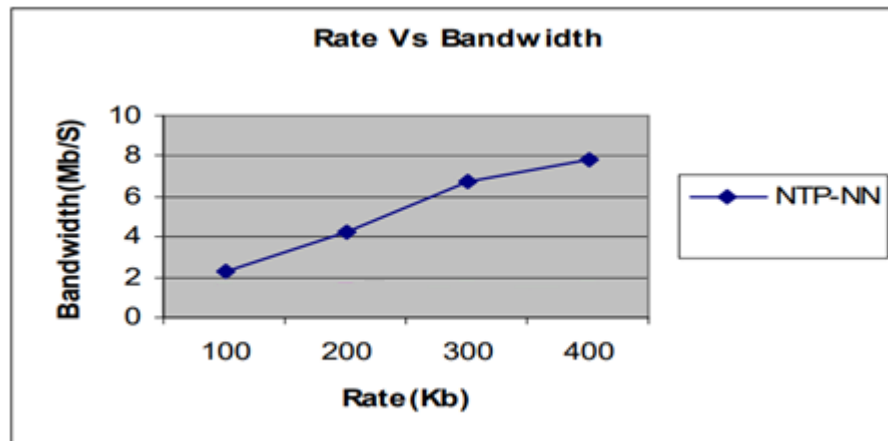


Рисунок 3.6 - Зависимость скорости от полученной пропускной способности

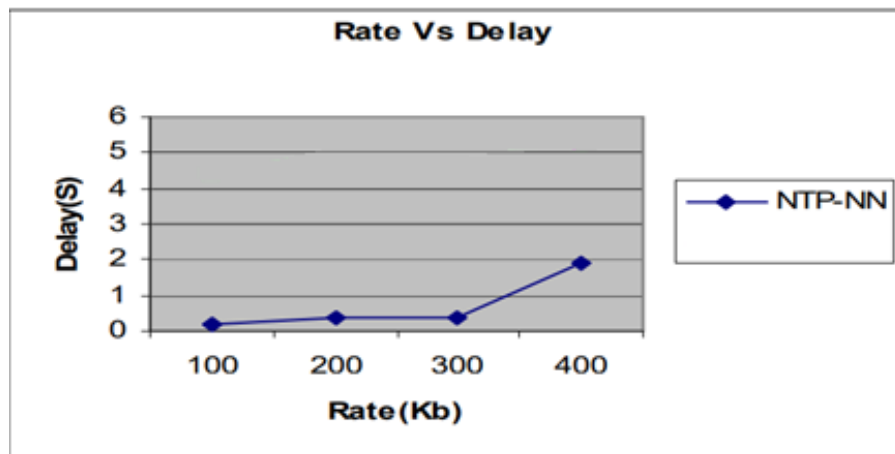


Рисунок 3.7 – Зависимость скорости от задержки

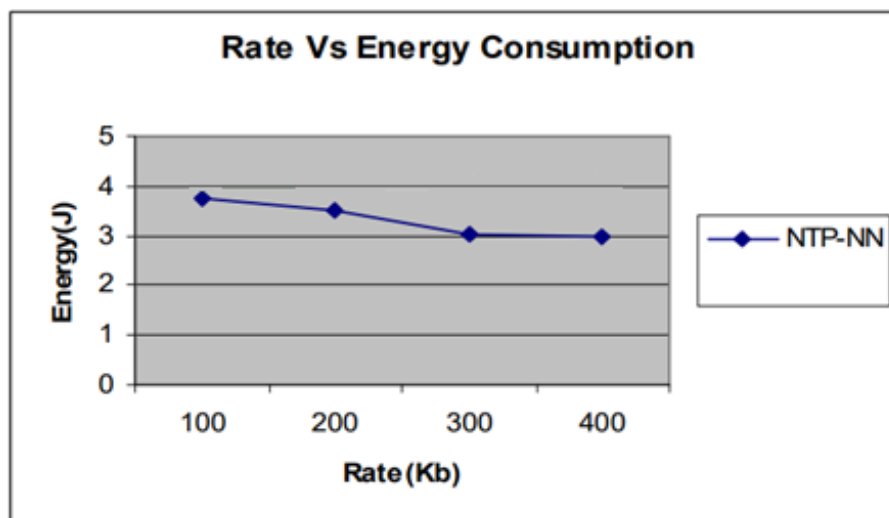


Рисунок 3.8 - Зависимость скорости от энергии

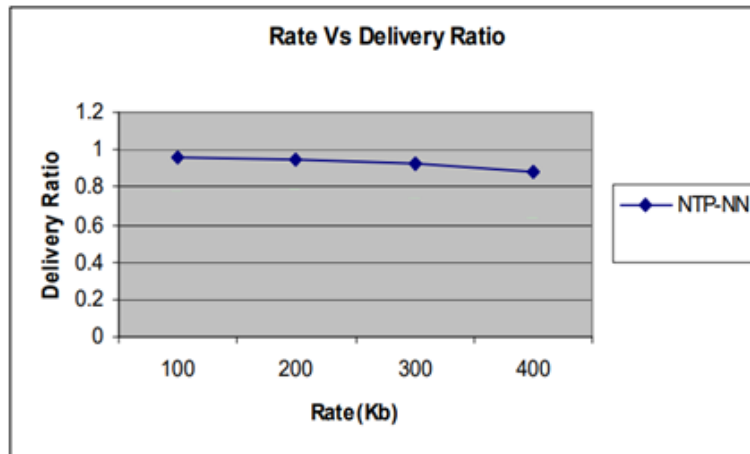


Рисунок 3.9 - Зависимость скорости от коэффициента доставки

В нашем следующем эксперименте, основанном на пользователях, мы измеряем пропускную способность каждого пользователя в обеих схемах.

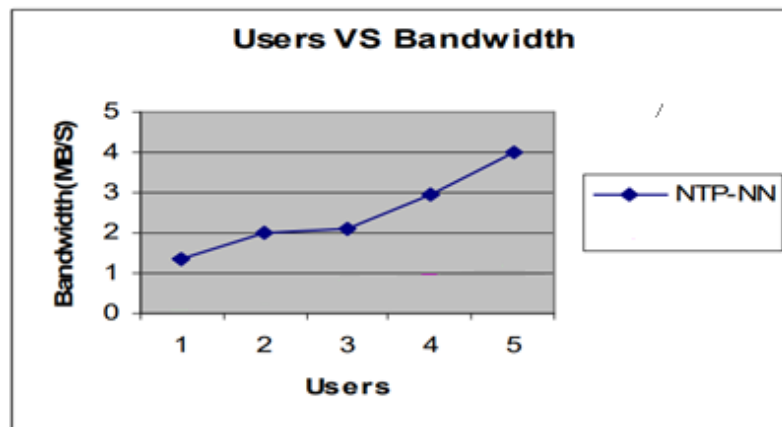


Рисунок 3.10 - Зависимость пользователей от полученной пропускной способности

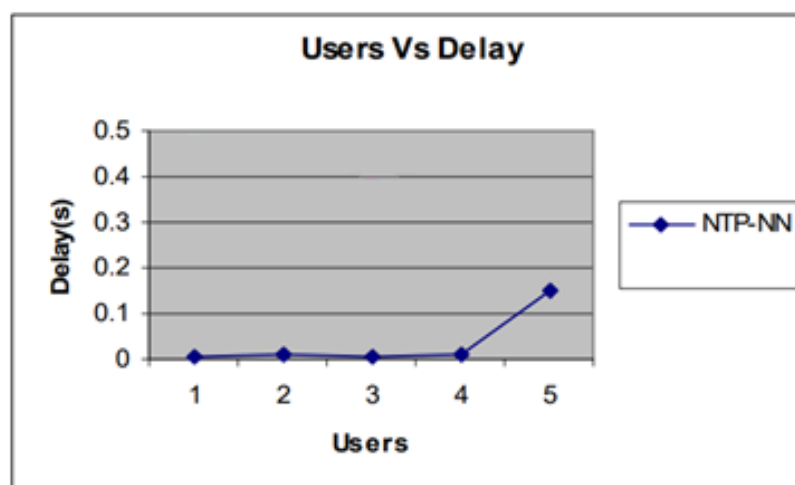


Рисунок 3.11 - Зависимость пользователей от задержки

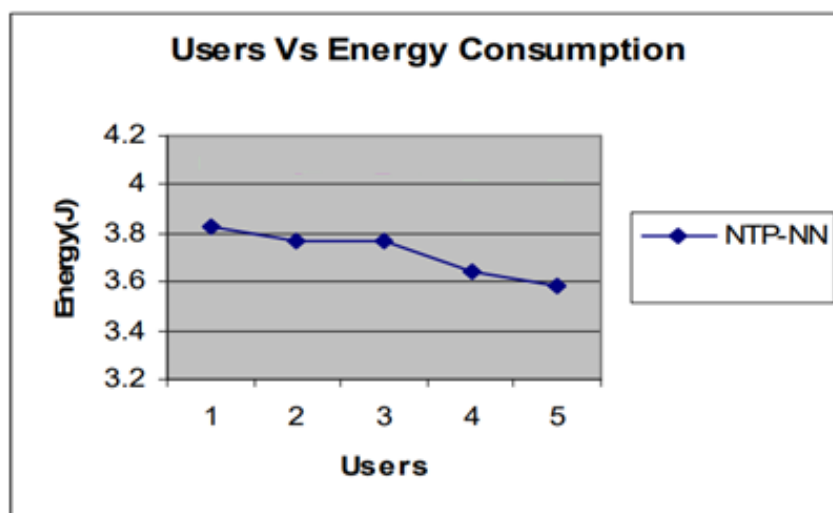


Рисунок 3.12 - Зависимость пользователей от потребления энергии

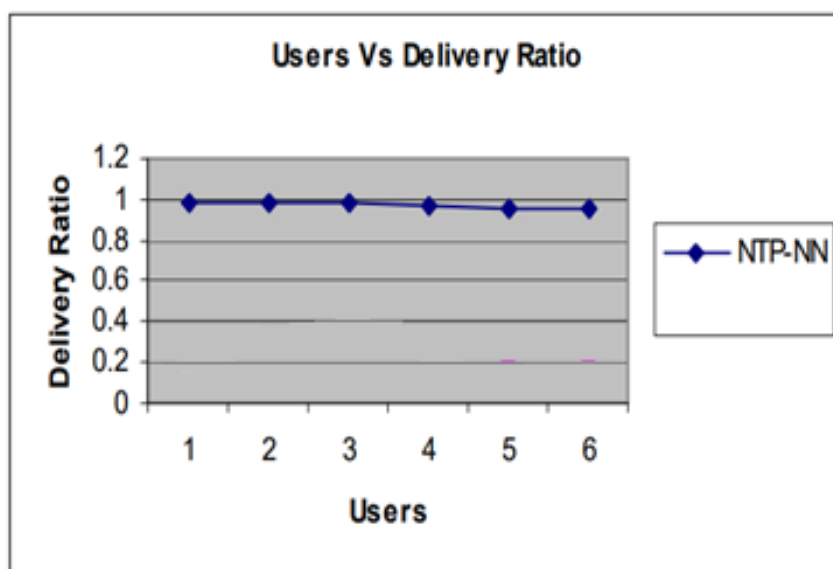


Рисунок 3.13 - Зависимость пользователей от коэффициента доставки

На рисунке 3.10 показана максимальная измеренная пропускная способность. На рисунке 3.11 показана средняя измеренная задержка. Из рисунка видно, что наша схема NTP-NN достигает меньше сквозной задержки.

## **Заключение**

В этой работе описали решение на основе нейронной сети для оценки сети и пропускной способности.

Исследование проводится в три этапа. Первый этап был использован для достижения высокой мобильности мобильного устройства, и было предоставлено решение для быстрой передачи обслуживания в сети. Вторым этапом включена нейронная сеть для определения приоритета работы. На третьем этапе была обеспечена лучшая пропускная способность для всех существующих услуг.

Эффективный расчет пропускной способности сделал больше ресурсов доступными в сети. Метод итеративной нейронной сети запланировал задания в соответствии с приоритетом. Поскольку решение является своевременным, работа эффективна с точки зрения текущего сценария. Основной проблемой качества обслуживания является ухудшение во время передачи обслуживания, чем количество больше в мобильных сетях, поскольку они имеют высокую мобильность.

Будущая работа должна быть сосредоточена на экономической эффективности с точки зрения пропускной способности и времени. Должна быть разработана какая-то распределенная система, чтобы не было зависимости от базовой станции.

## Список литературы

1. W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, On the Self-Similar Nature of Ethernet Traffic, in Proceedings of ACM SIGCOMM' 93, pages 183–193, 1993.
2. T. Karagiannis, M. Molle, and M. Faloutsos, A Nonstationary Poisson View of Internet Traffic, in IEEE INFOCOM 2004, volume 3, pages 1558–1569, Hong Kong, 2004.
3. A. Feldmann, A.C.Gilbert, W. Willinger, and T. Kurtz, The changing nature of network traffic: scaling phenomena, in ACM SIGCOMM Computer Communication Review, volume 28, pages 5–39, 1998.
4. V. Paxson and S. Floyd, Wide-Area traffic: The Failure of Poisson Modeling, in IEEE/ACM Transactions on Networking, volume 3, pages 226–244, 1995.
5. W. E. L. M. S. Taqqu, W. Willinger, and D. V. Wilson, On the Self-Similar Nature of Ethernet Traffic(Extended Version), in IEEE/ACM Transactions on Networking, volume 2, pages 1–15, 1994.
6. T. Kariagianis, M. Molle, and M. Faloutsos, Long Range Dependence - Ten Years of Internet Traffic Modeling, in IEEE Internet Computing, 2004.
7. M. Crovella and A. Bestavros, Self Similarity in World Wide Web Traffic: Evidence and Possible Causes, in IEEE/ACM Transactions on Networking, volume 5, pages 835–846, 1997.
8. M. Zukerman, T. Neame, and R. Addie, Internet Traffic Modeling and Future Tech- nology Implications, in IEEE INFOCOM 2003, volume 1, pages 587–596, 2003.
9. A. Sang and S. LiX, A predictability analysis of network traffic, in INFOCOM 2000, volume 1, pages 342 – 351, 2000.
10. S. Basu, A. Mukherjee, and S. Klivansky, Time Series Models for Internet Traffic, in IEEE INFOCOM96, California, USA, 1996.
11. M. Corradi, R. G. Garroppo, s. Giordano, and M. Pagano, Analysis of f-ARIMA processes in the modeling of broadband traffic, in ICC'01, volume 3, pages 964–968, 2001.
12. Y. Shu, M. Yu, J. Liu, O. Yang, and H. Feng, Wireless Traffic Modeling and Prediction using Seasonal ARIMA models, in IEICE Transactions on Communications, volume E88-B, pages 3992– 3999, 2005.
13. C. You and K. Chandra, Time Series Models for Internet Data Traffic, in 24th Conf. on Local Computer Networks, LCN'99, 1999.
14. B. Krithikaivasan, Y. Zeng, K. Deka, and D. Medhi, ARCH-based Traffic Forecasting and Dynamic Bandwidth Provisioning for Periodically Measured Nonstationary Traffic, in IEEE/ACM Transactions on Networking, volume 10, pages 870–883, 2006.
15. B. Zhou, D. He, and Z. Shun, Traffic Modeling and Prediction using ARIMA/GARCH Model, in Symposium on Modeling and Simulation tool for

emerging telecommunications networks: needs, trends, challenges and solutions, Meunchen, Germany, 2005.

16. C. Barakat, P. Thiran, G.Iannaccone, C.Diot, and P. OwezarskiV, A flow-based model for internet backbone traffic, in Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurement, pages 35–47, 2002.

17. B. Chen, S. Peng, and K. Wang, Traffic Modeling, Prediction, and Congestion Control for High-Speed Networks: A Fuzzy AR Approach., in IEEE/ACM Transactions on Fuzzy Systems, volume 8, 2000.

18. A. D. Doulamis, N. D. Doulamis, and S. D. Kollias, An Adaptable Neural Network Model for Recursive Nonlinear Traffic Prediction and Modeling of MPEG Video Sources, in IEEE Transactions on Neural Networks, volume 14, pages 150–166, 2003.

19. X. Tian, H. Wu, and C. Ji, A unified framework for understanding network traffic using independent wavelet models, in INFOCOM 2002, volume 1, pages 446–454, 2002.

20. C. Bruni, P. Andrea, U. Mocci, and C. Scoglio, Optimal Capacity Management of Virtual Paths in ATM Networks, in GLOBECOM 94, pages 207–211, 1994.

21. S. Ohta and K. Sato, Dynamic Bandwidth Control of the Virtual Path in an Asynchronous Transfer Mode Network, in IEEE/ACM Transactions on Communications, volume 40, pages 1239–1247, 1992.

22. A. Orda, G. Pacifici, and D. Pendarakis, An Adaptive virtual Path Allocation Policy for Broadband Networks, in IEEE INFOCOM 96, pages 329–336, 1996.

23. Y. Afek, M. Cohen, E. Haalman, and Y. Mansour, Dynamic bandwidth allocation policies, in IEEE INFOCOM96, pages 880–887, 1996.

24. K. Papagiannaki, N. Taft, Z. Zhang, and C. Diot, Long-Term Forecasting of Internet Backbone Traffic: Observations and Initial Models, in IEEE INFOCOM 03, pages 1178–1788, 2003.

25. B. Krishnamurthy, S. Sen, Y. Zhang, and Y. Chen, Sketch-based Change detection: Methods, Evaluation, and Applications, in Internet Measurement Conference(IMC 03), Miami, USA, 2003.

26. J. Jiang and S. Papavassiliou, Detecting Network Attacks in the Internet via Statistical Network Traffic Normality Prediction., in Journal of Network and Systems Management, volume 12, pages 51–72, 2004.

27. Mohan R. Akella, Rajan Batta, Eric M. Delmelle, Peter A. Rogerson, Alan Blatt, and Glenn Wilson, “Base Station Location and Channel Allocation in a Cellular Network with Emergency Coverage Requirements”, 2003.

28. Mubashar Mushtaq and Toufik Ahmed,” Hybrid Overlay Networks Management for Real-Time Multimedia Streaming over P2P Networks”, IFIP International Federation for Information Processing 2007.

29. Network Simulator, <http://www.isi.edu/nsnam/ns>



30. Melinda Barabas, Georgeta Boanea, Andrei B. Rus, Virgil Dobrota, Jordi Domingo-Pascual,” Evaluation of Network Traffic Prediction Based on Neural Networks with Multi-task Learning and Multiresolution Decomposition, 2011.