**Министерство науки и высшего образования РФ**

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»**

**(ВлГУ)**

СМК 08/03-18

Срок хранения 2 года

ВЛГУ.10.05.04.04.05.00 ПЗ

**ОТЧЕТ О НИР**

«Эксперимент»

Специальность 10.05.04 – «Информационно-аналитические системы безопасности»

СПИСОК ИСПОЛНИТЕЛЕЙ

|  |  |  |
| --- | --- | --- |
| Руководитель НИР |  | к.т.н. доцент кафедры ИЗИ Ю.М. Монахов |
|  |
| Исполнитель |  | студент группы ИСБ-120 Л.А. Мшвениерадзе |
|  |  |  |  |

Владимир 2023 г

## СОДЕРЖАНИЕ

[СОДЕРЖАНИЕ 2](#_Toc141557205)

[ВВЕДЕНИЕ 3](#_Toc141557206)

[1 СЕТЕВОЕ ИСЧИСЛЕНИЕ 4](#_Toc141557207)

[2 ОБЩИЙ ПОДХОД К ЭКСПЕРИМЕНТУ 5](#_Toc141557208)

[3 ХОД ЭКСПЕРИМЕНТА 6](#_Toc141557209)

[ЗАКЛЮЧЕНИЕ 9](#_Toc141557210)

[СПИСОК ЛИТЕРАТУРЫ 10](#_Toc141557211)

[ПРИЛОЖЕНИЕ А 11](#_Toc141557212)

## ВВЕДЕНИЕ

Отчет 11 страниц.

Объект исследования:

*Теория по сетевому исчислению.*

Предмет исследования:

*Проведение эксперимента, направленного на проверку применимости и эффективности теории.*

Цель исследования:

Проверка теории сетевого исчисления на практике.

В процессе прохождения практики требовалось решить следующие задачи:

1. Составление плана проведения эксперимента.
2. Подготовка оборудования.
3. Проведение эксперимента.

Проект имеет высокую актуальность с точки зрения инновационности и научного интереса, а также обладает практической значимостью и потенциалом для оптимизации сетевых ресурсов и принятия обоснованных решений.

## СЕТЕВОЕ ИСЧИСЛЕНИЕ

Сетевое исчисление — это набор последних разработок, которые обеспечивают глубокое понимание проблем, связанных с потоками, встречающимися в сетях. В основе сетевого исчисления лежит математическая теория диодов, в частности, диода Min-Plus (также называемая алгеброй Min-Plus). С помощью сетевого исчисления мы можем понять некоторые фундаментальные свойства сетей интегрированных услуг, управление потоком окон, составление расписания и измерение буфера или определение размеров задержки.

С помощью сетевого исчисления мы можем понять некоторые фундаментальные свойства сетей интегрированных услуг, управление потоком окон, составление расписания и измерение буфера или определение размеров задержки.

Кривая обслуживания (или Service Curve) в сетевом исчислении представляет собой математическую функцию, которая определяет гарантированный уровень обслуживания для определенного типа трафика или сервиса в сети. Она используется для описания ограничений на скорость обработки данных или пакетов в сетевом устройстве, таком как маршрутизатор или коммутатор.

Кривая обслуживания определяет верхние границы (верхние ограничения) для скорости обработки трафика в зависимости от времени. Она может быть представлена в различных форматах, таких как линейные функции, экспоненциальные функции и т.д. Каждый тип трафика или сервиса может иметь свою уникальную кривую обслуживания в зависимости от его требований к задержке и пропускной способности.

Использование кривых обслуживания позволяет сетевым инженерам и администраторам гарантировать определенный уровень обслуживания для различных видов трафика, управлять пропускной способностью и задержками, а также предотвращать перегрузки и потери данных в сети.

Одним из распространенных применений кривых обслуживания является предоставление гарантированных и максимальных скоростей передачи данных для различных классов обслуживания, что позволяет обеспечивать качество обслуживания в сети, адаптированное к требованиям различных типов трафика. Это важно в сетях, где сосуществуют разнородные приложения с разными потребностями в пропускной способности и задержках.

## ОБЩИЙ ПОДХОД К ЭКСПЕРИМЕНТУ

В ходе исследования был проведен эксперимент, целью которого было наглядно подтвердить теорию о кривой обслуживания.

Эксперимент — это метод научного исследования, основанный на контролируемом и систематическом изучении явлений или процессов для проверки гипотезы, получения новых знаний или подтверждения теоретических предположений. Цель эксперимента состоит в том, чтобы установить причинно-следственные связи и выявить закономерности.

Для проведения эксперимента необходимо следовать определенной методологии и учесть ряд ключевых принципов:

* Формулирование гипотезы: Определение четкой гипотезы, которую мы хотим проверить в ходе эксперимента. Гипотеза должна быть конкретной и содержать прогнозируемый результат.
* Создание экспериментального плана: разработать детальный план проведения эксперимента, включая последовательность действий, определение контрольных и экспериментальных групп (если применимо), и критерии измерения результатов.
* Проведение эксперимента: Выполнение плана эксперимента, следуя строго заранее определенным шагам и протоколу.
* Анализ данных и интерпретация результатов: Оценка полученных данных. Интерпретация результатов.
* Вывод: Выводы на основе результатов эксперимента и описание их в докладе.

## ХОД ЭКСПЕРИМЕНТА

Для проведения эксперимента был подготовлен план.

* Определение цели эксперимента: наглядно получить график кривой обслуживания.
* Распределение задач в команде: Григорий Сухецкий – разработка ПО; Павел Мухин – разработка ПО; Леван Мшвениерадзе – техническое обеспечение и подготовка оборудования.
* Оснащенность оборудованием: Источник ПК-1 (системные характеристики: Процессор - Intel Core i5-3450; Видеокарта - Nvidia 640; Сетевая карта - Realtek Semiconductor пропускная способность - 1 Гб/С. Оперативная память -12 Гб); Приёмник ПК-2 (системные характеристики: Процессор - Intel Core i5-5300U; Видеокарта - Intel HD Graphics 5500; Сетевая карта - Intel Corporation пропускная способность - 1 Гб/С. Оперативная память - 8 Гб); Узел роутер (asus RT-g32); два кабеля RG-45.
* Составление топологии сети (рис.1).
* Написание программного обеспечения (см. в приложение A). Также использовалось дополнительное программное обеспечение: PyCharm Community; Ubuntu OS 22.10; Visual Studio Code; Python 3.9.
* Оценка полученных данных. Интерпретация результатов.

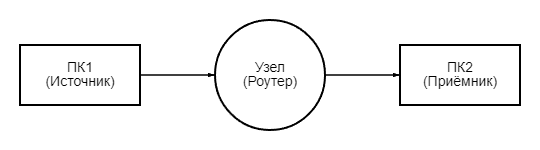


Рисунок 1.

В ходе эксперимента было написано программное обеспечение реализующая функции клиент-серверного приложения (см. в приложение A).

Функции сервера выполняет ПК1:

* Создается сокет для отправки данных клиенту.
* Инициализируется остаточный трафик R(t) и списки для хранения времени и остаточного трафика.
* Инициализируется виртуальный остаточный трафик R\*(t) и списки для хранения времени и виртуального остаточного трафика.
* Запускается бесконечный цикл для отправки данных клиенту и визуализации графиков.
* В цикле генерируется случайный объем данных и отправляется клиенту вместе с временем и размером данных.

Функции клиента выполняет ПК2:

* Создается сокет для приема данных от сервера и привязывается к адресу сервера.
* Инициализируется остаточный трафик R(t) и списки для хранения времени и остаточного трафика.
* Инициализируется виртуальный остаточный трафик R\*(t) и списки для хранения времени и виртуального остаточного трафика.
* Инициализируются Кривая прибытия и Кривая обслуживания (параметрические кривые).
* Запускается бесконечный цикл для приема данных от сервера и обновления графиков.
* В цикле получаются данные остаточного трафика R(t) от сервера.
* Обновляются данные остаточного трафика R(t) и времени прихода пакета.
* Обновляются данные виртуального остаточного трафика R\*(t).
* Вычисляются Кривая прибытия и Кривая обслуживания для каждого времени t.
* Построение всех графиков на одной фигуре:

Результатом работы программы будет: график, на котором будут отображены следующие кривые (рис. 2):

Остаточный трафик R(t) - ступенчатый график, который растет по мере получения данных от сервера.

Виртуальный остаточный трафик R\*(t) - ступенчатый график, который растет по мере времени с момента старта клиента.

Кривая прибытия - график, который представляет собой кривую.

Кривая обслуживания - график, который представляет собой кривую.

График функции R ⊗ B (max(R(t), B(t))) - это график, который представляет собой максимум из кривых R(t) и B(t) для каждого момента времени t.

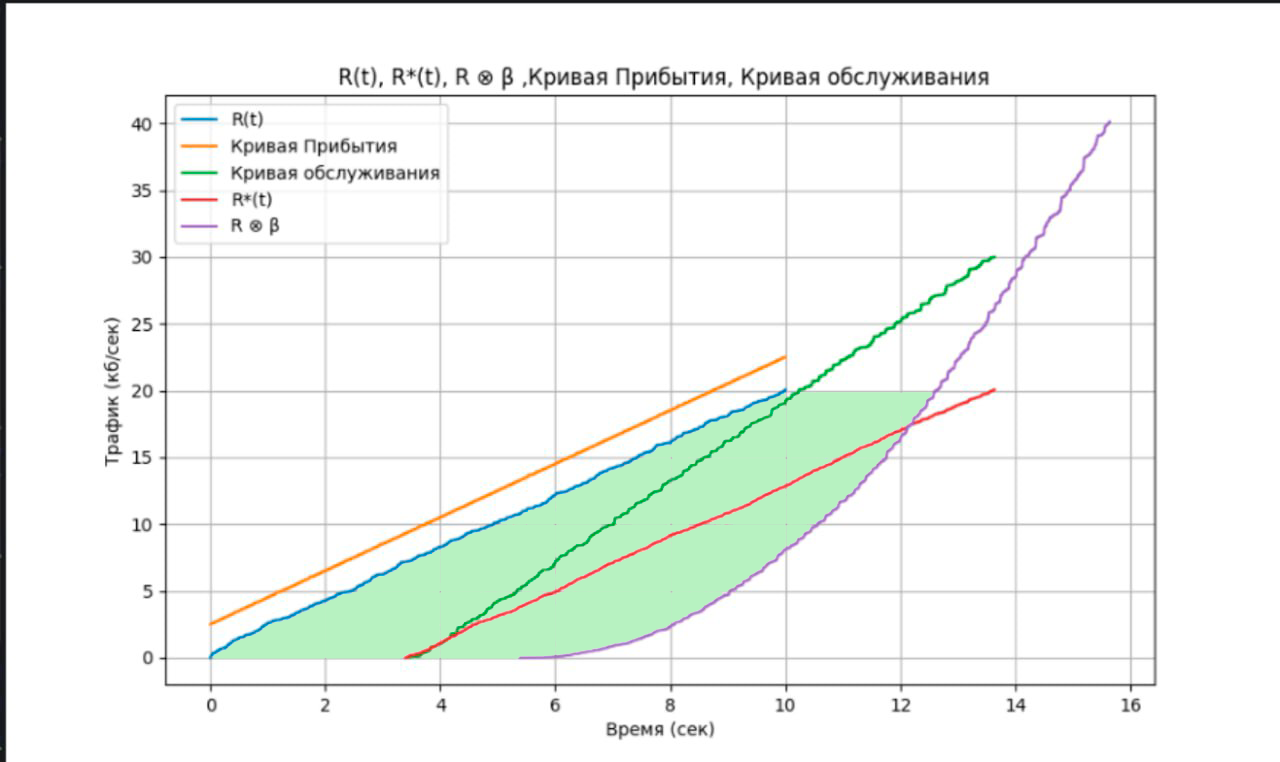


Рисунок 2.

## ЗАКЛЮЧЕНИЕ

В данном исследовании были собраны экспериментальные данные, которые подтверждают теоретические предположения и концепции Network Calculus. Более того, было разработано специальное программное обеспечение, которое позволяет получать необходимую информацию в режиме реального времени. Это программное обеспечение обеспечивает возможность наблюдения и анализа процессов передачи данных в сети, что помогает подтвердить существующие теоретические модели и применить их на практике для оптимизации работы сетевых ресурсов и управления полосой пропускания. Таким образом, исследование предоставляет ценную информацию и инструменты для улучшения производительности и эффективности сетей.

## СПИСОК ЛИТЕРАТУРЫ

1. Boudec J.-Y.L., Thiran P. A Theory of Deterministic Queuing Systems for the Internet, P. 265.

## ПРИЛОЖЕНИЕ А

Исходный код Client.py : <https://github.com/Farenes/network-calculus-study-guide/blob/experiment/client.py>

Исходный код Server.py : <https://github.com/Farenes/network-calculus-study-guide/blob/experiment/server.py>