Лабораторная работа №4

Эмуляция и измерение задержек в глобальных сетях

Тазаева Анастасия Анатольевна

Содержание

# 1 Цель работы

Основной целью работы является знакомство с NETEM — инструментом для тестирования производительности приложений в виртуальной сети, а также получение навыков проведения интерактивного и воспроизводимого экспериментов по измерению задержки и её дрожания (jitter) в моделируемой сети в среде Mininet.

# 2 Задание

1. Задайте простейшую топологию, состоящую из двух хостов и коммутатора с назначенной по умолчанию mininet сетью 10.0.0.0/8.
2. Проведите интерактивные эксперименты по добавлению/изменению задержки, джиттера, значения корреляции для джиттера и задержки, распределения времени задержки в эмулируемой глобальной сети.
3. Реализуйте воспроизводимый эксперимент по заданию значения задержки в эмулируемой глобальной сети. Постройте график.
4. Самостоятельно реализуйте воспроизводимые эксперименты по изменению задержки, джиттера, значения корреляции для джиттера и задержки, распределения времени задержки в эмулируемой глобальной сети. Постройте графики.

# 3 Выполнение лабораторной работы

## 3.1 Запуск лабораторной топологии

1. Запустила виртуальную среду с mininet. Из основной ОС подключилась к виртуальной машине (рис. 1).

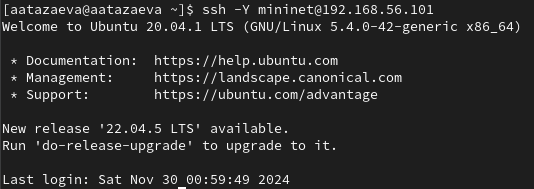


Рис. 1: Подключение к mininet

1. Исправила права запуска X-соединения (рис. 2).

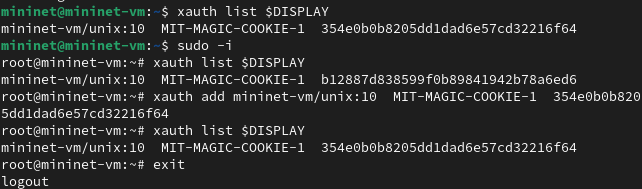


Рис. 2: Исправление прав запуска X-соединения

1. Создала простейшую тополонию, состоящую из двух хостов и коммутатора с назначенной по умолчанию mininet сетью 10.0.0.0/8 (рис. 3). Терминалы коммутатора и контроллера закрыла.

sudo mn --topo=single,2 -x

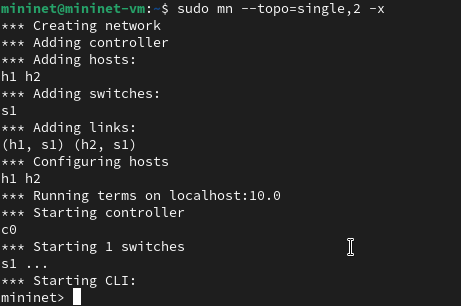


Рис. 3: Создание топологии

1. На хостах h1 и h2 ввела команду ifconfig (рис. 4 и 5), чтобы отобразить информацию, относящуюся к их сетевым интерфейсам и назначенным им IP-адресам. В дальнейшем при работе с NETEM и командой tc будут использоваться интерфейсы h1-eth0 и h2-eth0 .

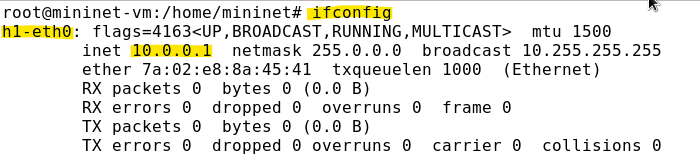


Рис. 4: ifconfig h1

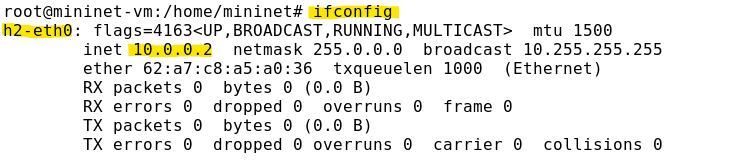


Рис. 5: ifconfig h2

1. Проверила подключение между хостами h1 и h2 с помощью команды ping с параметром -c 6 (рис. 6).

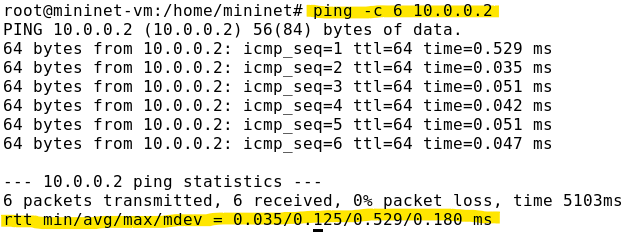


Рис. 6: Проверка подключения ping h1 -> h2

1. Минимальное(=0.035 ms), среднее(=0.125 ms), максимальное(=0.529 ms) и стандартное(=0.180 ms) отклонение времени приёма-передачи (RTT).

## 3.2 Интерактивные эксперименты

### 3.2.1 Добавление/изменение задержки в эмулируемой глобальной сети

1. На хосте h1 добавила задержку в 100 мс к выходному интерфейсу (рис. 7):

sudo tc qdisc add dev h1-eth0 root netem delay 100ms

Рис. 7: Задержка для хоста h1

Рис. 7: Задержка для хоста h1

Здесь: - sudo : выполнить команду с более высокими привилегиями; - tc : вызвать управление трафиком Linux; - qdisc : изменить дисциплину очередей сетевого планировщика; - add : создать новое правило; - dev h1-eth0 : указать интерфейс, на котором будет применяться правило; - netem : использовать эмулятор сети; - delay 100ms : задержка ввода 100 мс.

1. Проверила, что соединение от хоста h1 к хосту h2 имеет задержку 100 мс, используя команду ping с параметром -c 6 с хоста h1 . Минимальное(=100.221 ms), среднее(=100.647 ms), максимальное(=101.032 ms) и стандартное(=0.329 ms) отклонение времени приёма-передачи (RTT) (рис. 8).

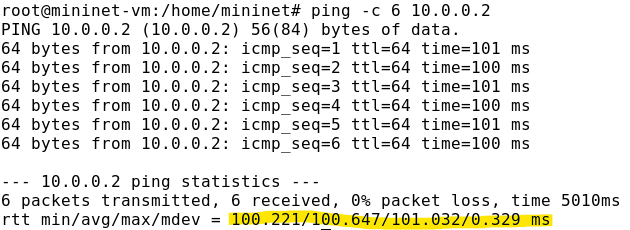


Рис. 8: Задержка для хоста h1. Проверка

1. Для эмуляции глобальной сети с двунаправленной задержкой необходимо к соответствующему интерфейсу на хосте h2 также добавить задержку в 100 миллисекунд (рис. 9):

sudo tc qdisc add dev h2-eth0 root netem delay 100ms

Рис. 9: Задержка для хоста h2

Рис. 9: Задержка для хоста h2

1. Проверила, что соединение между хостом h1 и хостом h2 имеет RTT в 200 мс (100 мс от хоста h1 к хосту h2 и 100 мс от хоста h2 к хосту h1 ), повторив команду ping с параметром -c 6 на терминале хоста h1 . Минимальное(=200.238 ms), среднее(=200.904 ms), максимальное(=201.980 ms) и стандартное(=0.632 ms) отклонение времени приёма-передачи (RTT) (рис. 10).

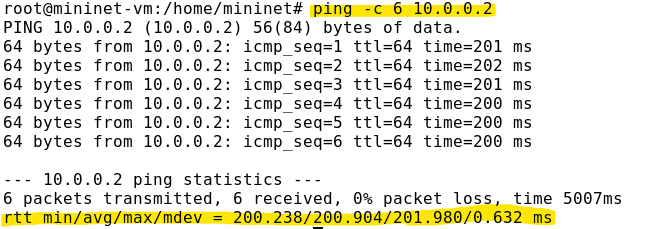


Рис. 10: Задержка для хоста h1(+задержка от хоста h2. Проверка

### 3.2.2 Изменение задержки в эмулируемой глобальной сети

1. Изменила задержку со 100 мс до 50 мс для отправителя h1 (рис. 11):

sudo tc qdisc change dev h1-eth0 root netem delay 50ms

и для получателя h2 (рис. 12):

sudo tc qdisc change dev h2-eth0 root netem delay 50ms

Рис. 11: Изменение задержки для хоста h1

Рис. 11: Изменение задержки для хоста h1

Рис. 12: Изменение задержки для хоста h2

Рис. 12: Изменение задержки для хоста h2

1. Проверила, что соединение от хоста h1 к хосту h2 имеет задержку 100 мс, используя команду ping с параметром -c 6 с терминала хоста h1 . На (рис. 13) выделила минимальное, среднее, макси-мальное и стандартное отклонение времени приёма-передачи (RTT).

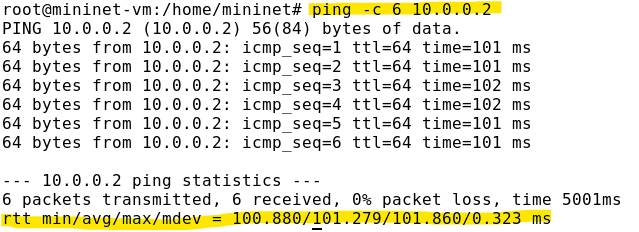


Рис. 13: Задержка для хоста h1 после изменения задержки. Проверка

##Восстановление исходных значений (удаление правил) задержки в эмулируемой глобальной сети#

1. Восстановила конфигурацию по умолчанию, удалив все правила, применённые к сетевому планировщику соответствующего интерфейса. Для отправителя h1 (рис. 14):

sudo tc qdisc del dev h1-eth0 root netem

Для получателя h2 (рис. 15):

sudo tc qdisc del dev h2-eth0 root netem

Рис. 14: Удаление задержки для хоста h1

Рис. 14: Удаление задержки для хоста h1

Рис. 15: Удаление задержки для хоста h2

Рис. 15: Удаление задержки для хоста h2

1. Проверила, что соединение между хостом h1 и хостом h2 не имеет явноустановленной задержки, используя команду ping с параметром -c 6 с терминала хоста h1. На (рис. 16) выделила минимальное, среднее, макси-мальное и стандартное отклонение времени приёма-передачи (RTT).

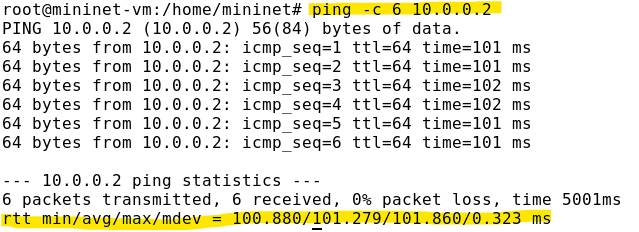


Рис. 16: Задержка для хоста h1 после удаления задержки. Проверка

### 3.2.3 Добавление значения дрожания задержки в интерфейс подключения к эмулируемой глобальной сети

1. Добавила на узле h1 задержку в 100 мс со случайным отклонением 10 мс (рис. 17):

sudo tc qdisc add dev h1-eth0 root netem delay 100ms 10ms

Рис. 17: Установка задержки для хоста h1 с случайным отклонением

Рис. 17: Установка задержки для хоста h1 с случайным отклонением

1. Проверила, что соединение от хоста h1 к хосту h2 имеет задержку 100 мс со случайным отклонением ±10 мс, используя в терминале хоста h1 команду ping с параметром -c 6 . На (рис. 18) видно минимальное, среднее, максимальное и стандартное отклонение времени приёма-передачи (RTT).

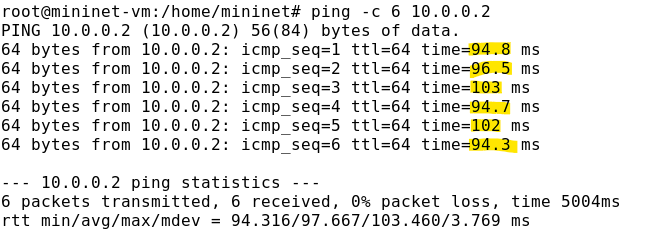


Рис. 18: Задержка для хоста h1 с случайным отклонением. Проверка

1. Восстановила конфигурацию интерфейса по умолчанию на узле h1 (рис. 19).

Рис. 19: Удаление задержки для хоста h1

Рис. 19: Удаление задержки для хоста h1

### 3.2.4 Добавление значения корреляции для джиттера и задержки в интерфейс подключения к эмулируемой глобальной сети

1. Добавила на интерфейсе хоста h1 задержку в 100 мс с вариацией ±10 мс и значением корреляции в 25% (рис. 20):

sudo tc qdisc add dev h1-eth0 root netem delay 100ms 10ms 25%

Рис. 20: Установка задержки для хоста h1 с значением корреляции

Рис. 20: Установка задержки для хоста h1 с значением корреляции

1. Убедилась, что все пакеты, покидающие устройство h1 на интерфейсе h1-eth0 , будут иметь время задержки 100 мс со случайным отклонением ±10 мс, при этом время передачи следующего пакета зависит от предыдущего значения на 25%. Используйте для этого в терминале хоста h1 команду ping с параметром -c 20 . На (рис. 21) видно минимальное, среднее, максимальное и стандартное отклонение времени приёма-передачи (RTT).

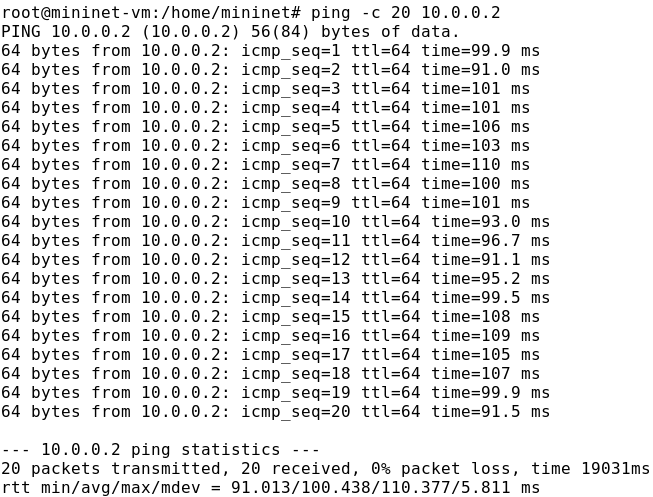


Рис. 21: Задержка для хоста h1 с значением корреляции. Проверка

1. Восстановила конфигурацию интерфейса по умолчанию на узле h1.

### 3.2.5 Распределение задержки в интерфейсе подключения к эмулируемой глобальной сети

NETEM позволяет пользователю указать распределение, которое описывает, как задержки изменяются в сети. В реальных сетях передачи данных задержки неравномерны, поэтому при моделировании может быть удобно использовать некоторое случайное распределение, например, нормальное, парето или парето-нормальное.

1. Задала нормальное распределение задержки на узле h1 в эмулируемой сети (рис. 22):

sudo tc qdisc add dev h1-eth0 root netem delay 100ms 20ms distribution normal

Рис. 22: Установка задержки для хоста h1 и задание нормального распределения

Рис. 22: Установка задержки для хоста h1 и задание нормального распределения

1. Убедилась, что все пакеты, покидающие хост h1 на интерфейсе h1-eth0 , будут иметь время задержки, которое распределено в диапазоне 100 мс ±20 мс. Используйте для этого команду ping на терминале хоста h1 с параметром -c 10 (рис. 23).

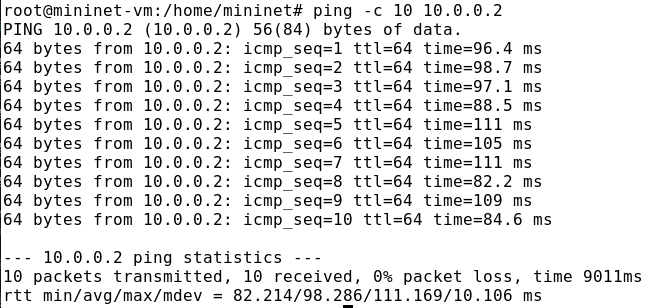


Рис. 23: Задержка для хоста h1 с нормальным распределением. Проверка

1. Восстановила конфигурацию интерфейса по умолчанию на узле h1 и завершила работу mininet в интерактивном режиме.

## 3.3 Воспроизведение экспериментов

### 3.3.1 Предварительная подготовка

1. Обновила репозитории программного обеспечения на виртуальной машине (рис. 24):

sudo apt-get update

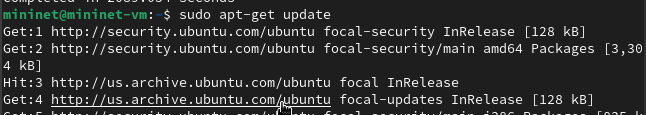


Рис. 24: Обновление репозиториев

1. Установила пакет geeqie — понадобится для просмотра файлов png (рис. 25):

sudo apt install geeqie

Рис. 25: Установка пакета geeqie

Рис. 25: Установка пакета geeqie

1. Для каждого воспроизводимого эксперимента expname создала свой каталог, в котором будут размещаться файлы эксперимента (рис. 26):

mkdir -p ~/work/lab\_netem\_i/expname

Рис. 26: Создание каталога для размещения файлов эксперимента

Рис. 26: Создание каталога для размещения файлов эксперимента

### 3.3.2 Добавление задержки для интерфейса, подключающегося к эмулируемой глобальной сети

С помощью API Mininet воспроизвела эксперимент по добавлению задержки для интерфейса хоста, подключающегося к эмулируемой глобальной сети.

1. В виртуальной среде mininet в своём рабочем каталоге с проектами создала каталог simple-delay и перешла в него (рис. 27):

mkdir -p ~/work/lab\_netem\_i/simple-delay  
cd ~/work/lab\_netem\_i/simple-delay

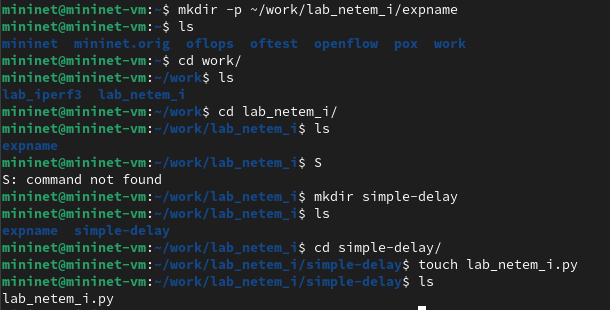


Рис. 27: Создание каталога imple-delay

1. Создала скрипт для эксперимента lab\_netem\_i.py (рис. 28 и 29).

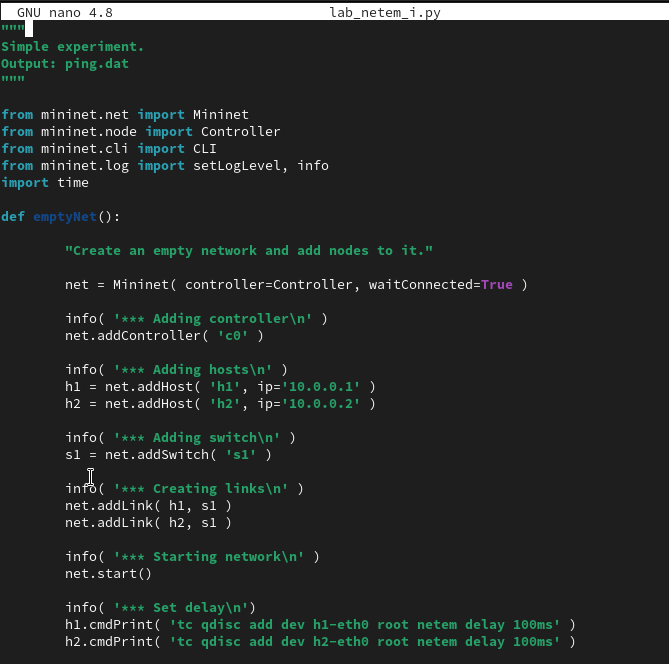


Рис. 28: lab\_netem\_i.py

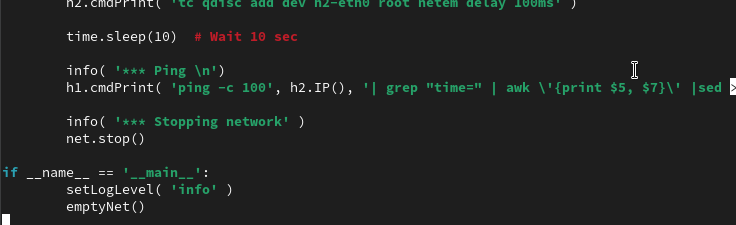


Рис. 29: lab\_netem\_i.py. Продолжение

* В каких строках скрипта задается значение задержки для интерфейса хоста?  
  h1.cmdPrint( 'tc qdisc add dev h1-eth0 root netem delay 100ms' ) h2.cmdPrint( 'tc qdisc add dev h2-eth0 root netem delay 100ms' )
* Каким образом формируется файл с результатами эксперимента для последующего построения графиков?

h1.cmdPrint( 'ping -c 100', h2.IP(), '| grep "time=" | awk \'{print $5, $7}\' | sed -e \'s/time=//g\' -e \'s/icmp\_seq=//g\' > ping.dat' )

1. Создала скрипт для визуализации ping\_plot результатов эксперимента (рис. 30).

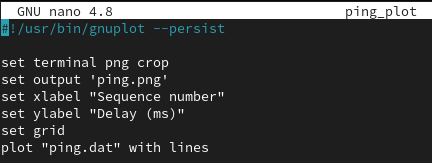


Рис. 30: ping\_plot

1. Задала права доступа к файлу скрипта (рис. 31).

Рис. 31: Изменение прав для файла скрипта

Рис. 31: Изменение прав для файла скрипта

1. Создала Makefile для управления процессом проведения эксперимента (рис. 32).

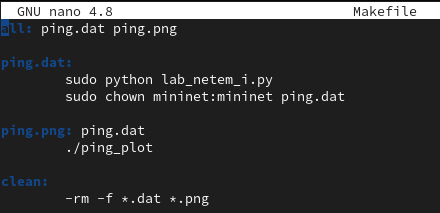


Рис. 32: Makefile

1. Выполнила эксперимент (рис. 33).

make

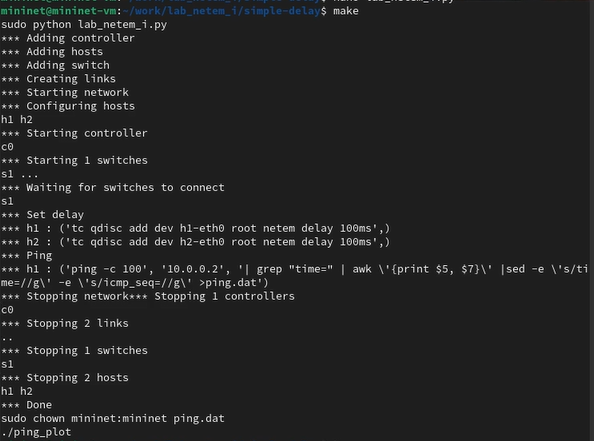


Рис. 33: Make

1. Получился следущий график (рис. 34).

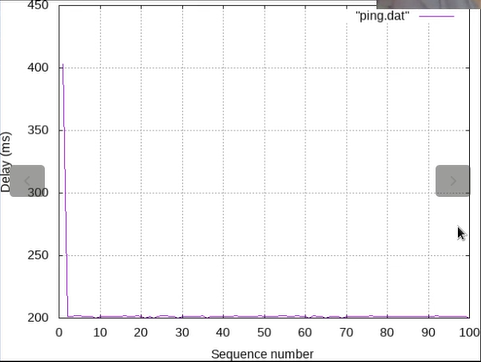


Рис. 34: График с большой разницей первого значения с последующими

1. Из файла ping.dat удалила первую строку и заново построила график (рис. 35):

make ping.png

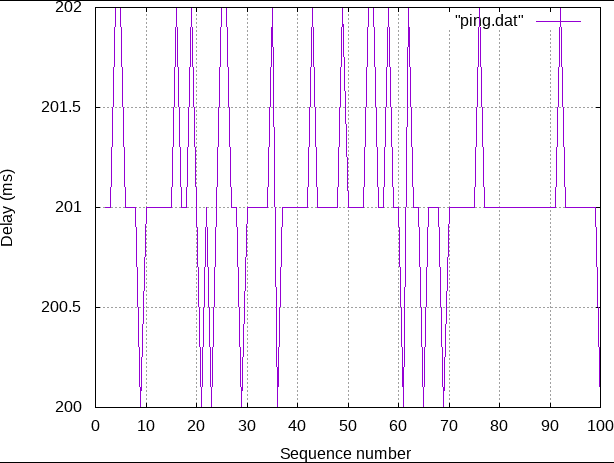


Рис. 35: График

1. Разработала скрипт для вычисления на основе данных файла ping.dat минимального, среднего, максимального и стандартного отклонения времени (рис. 36 и 38) и изменила файл Makefile (рис. 37).

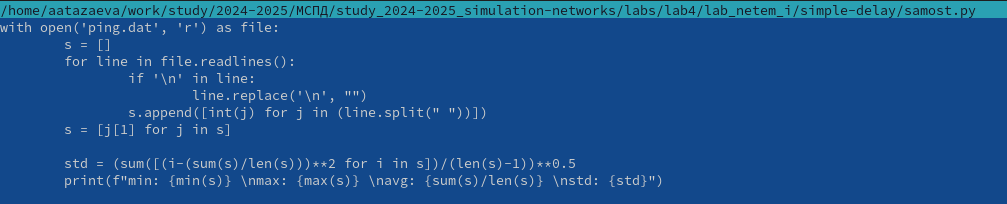


Рис. 36: samost.py

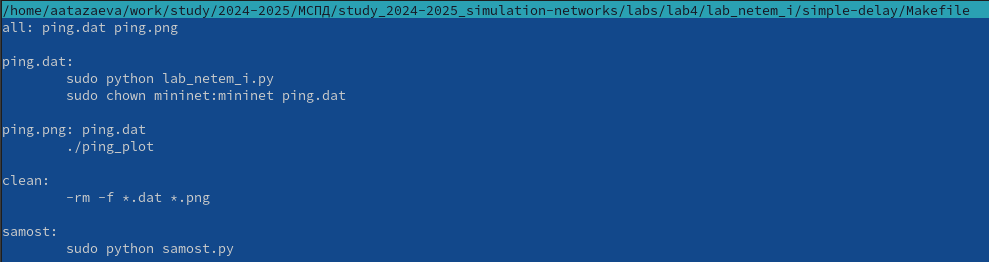


Рис. 37: Makefile. измененный

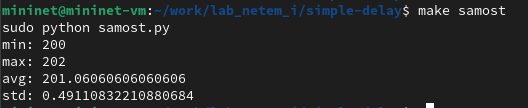


Рис. 38: samost.py выполнение

1. Очистила каталог от результатов проведения экспериментов:

make clean

# 4 Выводы

Познакомилась с NETEM — инструментом для тестирования производительности приложений в виртуальной сети, а также получила навыков проведения интерактивного и воспроизводимого экспериментов по измерению задержки и её дрожания (jitter) в моделируемой сети в среде Mininet.