

Презентация по лабораторной работе №2

Дисциплина: Имитационное моделирование

Лобанова П.И.

21 февраля 2025

Российский университет дружбы народов, Москва, Россия

Информация

- Лобанова Полина Иннокентьевна
- Учащаяся на направлении “Фундаментальная информатика и информационные технологии”
- Студентка группы НФИбд-02-22
- polla-2004@mail.ru

Цель

Исследование протокола TCP и алгоритма управления очередью RED.

Задание

1. Описание моделируемой сети:

- сеть состоит из 6 узлов;
- между всеми узлами установлено дуплексное соединение с различными пропускной способностью и задержкой 10 мс;
- узел r1 использует очередь с дисциплиной RED для накопления пакетов, максимальный размер которой составляет 25;
- TCP-источники на узлах s1 и s2 подключаются к TCP-приёмнику на узле s3;
- генераторы трафика FTP прикреплены к TCP-агентам.

2. – Измените в модели на узле s1 тип протокола TCP с Reno на NewReno, затем на Vegas.
Сравните и поясните результаты.
- Внесите изменения при отображении окон с графиками (измените цвет фона, цвет траекторий, подписи к осям, подпись траектории в легенде).

Выполнение

```

proc plotWindow {tcpSource file} {
    global ns
    set time 0.01
    set now [$ns now]
    set cwnd [$tcpSource set cwnd_]
    puts $file "$now Scwnd"
    $ns at [expr $now+$time] "plotWindow $tcpSource $file"
}

# Процедура finish:
proc finish {} {
    global tchan
    # подключения кода AWK:
    set awkCode {
        {
            if ($1 == "0" && NF>2) {
                print $2, $3 >> "temp.q";
                set end $2
            }
            else if ($1 == "a" && NF>2)
                print $2, $3 >> "temp.a";
        }
    }

    set f [open temp.queue w]
    puts $f "TitleText: red"
    puts $f "Device: Postscript"

    if { [info exists tchan_] } {
        close $tchan_
    }

    exec rm -f temp.q temp.a
    exec touch temp.a temp.q
    exec awk $awkCode all.q
    puts $f "\nqueue"
    exec cat temp.q >@ $f
    puts $f "\n\ave_queue"
    exec cat temp.a >@ $f
    close $f

    # Запуск xgraph с графиками окна TCP и очереди:
    exec xgraph -bb -tk -x time -t "TCPRenoCWND" WindowVsTimeReno &
    exec xgraph -bb -tk -x time -y queue temp.queue &
    exit 0
}

$ns run

```

Рис. 1: Заполнение файла

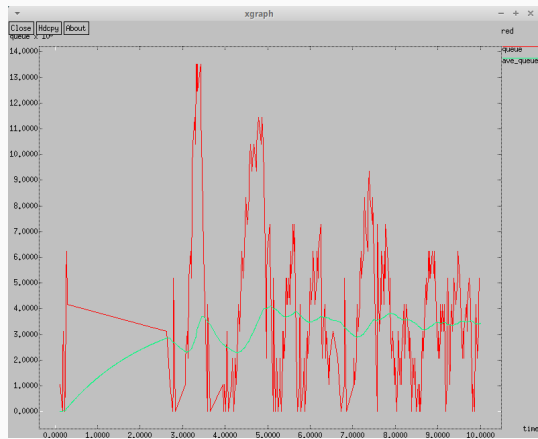


Рис. 2: График изменения длины очереди и средней длины очереди

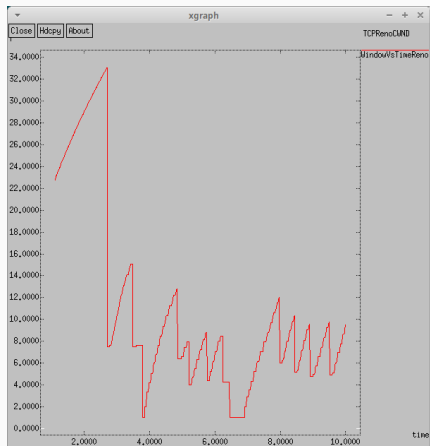


Рис. 3: График изменения TCP-окна

Изменила в модели на узле s1 тип протокола TCP с Reno на NewReno.

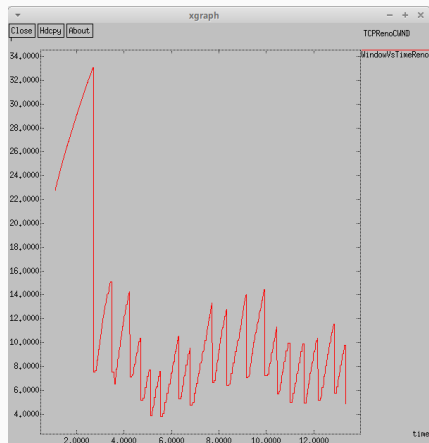


Рис. 4: График изменения TCP-окна

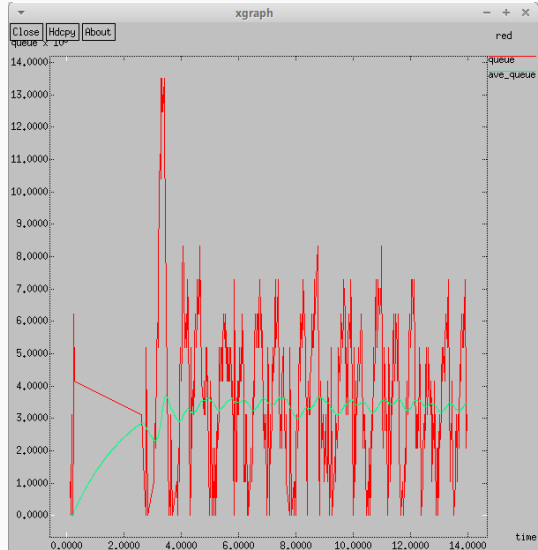


Рис. 5: График изменения длины очереди и средней длины очереди

Изменила в модели на узле s1 тип протокола TCP с NewReno на Vegas.

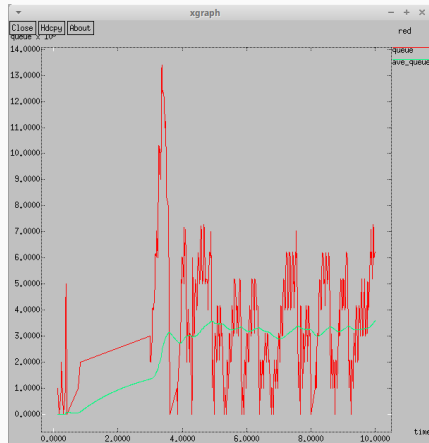


Рис. 6: График изменения длины очереди и средней длины очереди

Внесла изменения при отображении окон с графиками (изменила цвет фона, цвет траекторий, подписи к осям, подпись траектории в легенде).

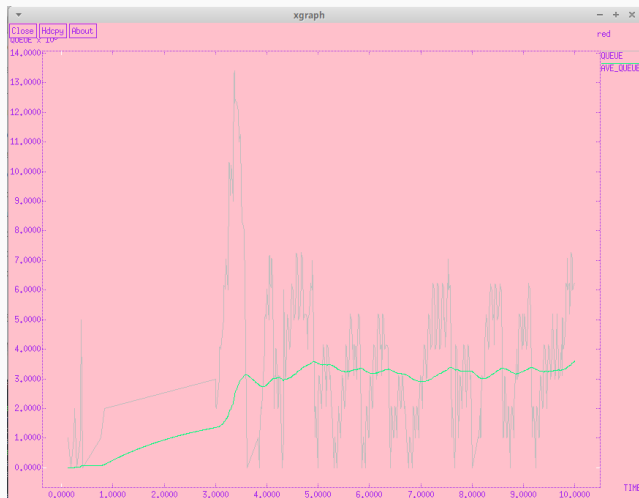


Рис. 7: График изменения длины очереди и средней длины очереди

Вывод

Я исследовала протокол TCP и алгоритм управления очередью RED.