

**Батыр С.С., Хорхордин А.В.**

Донецкий национальный технический университет, г. Донецк,  
Кафедра «Автоматики и телекоммуникаций»  
e-mail: [sbatyr@gmail.com](mailto:sbatyr@gmail.com)

## **ПОСТРОЕНИЕ МОДЕЛИ СЕТИ ПЕРЕДАЧИ ДАННЫХ ДЛЯ ИССЛЕДОВАНИЯ ТЕХНОЛОГИИ AQM**

### ***Аннотация***

***Батыр С.С., Хорхордин А.В. Построение модели сети передачи данных для исследования технологии AQM***

*Участок сети Интернет, состоящий из передатчика протокола TCP и пограничного маршрутизатора, рассмотрен как объект в теории автоматического управления. Проведен обзор и анализ существующих алгоритмов технологии AQM. Выделены входные и выходные переменные маршрутизатора, построена его структурная схема. Синтезированы структурных схем и уравнений модели участка сети. Промоделированы существующие и предложен новый алгоритм для управления очередью.*

***Ключевые слова:*** перегрузка, нелинейная дискретная система, алгоритм управления очередью, RED

**Введение.** Ввиду непрекращающегося роста популярности сети Интернет и стремительного роста количества широкополосных подключений остро встает вопрос с перегрузками каналов связи провайдеров Интернет

Современные средства коммуникаций используют преимущественно каналы связи с пакетной коммутацией. Это позволяет повысить эффективность использования каналов, однако приводит к снижению надежности доставки информации. При перегрузке канала с пакетной коммутацией возможна ситуация, когда пакет на входе канала не поместится во входной буфер, а будет отброшен.

Это экономически слабо обоснованная, но широко распространенная практика среди провайдеров 3-его уровня. Широкие каналы связи позволяют использовать простое и дешевое коммуникационное оборудование, которое не поддерживает технологии управления.

**Анализ публикаций.** В работе [1] показано, что соблюдать требования QoS для современных мультимедийных приложений без внедрения технологий активного управления очередью (Active Queue Management – AQM) возможно лишь при загрузке сетевого оборудования не более чем на 30 %. Внедрение AQM позволяет соблюсти требования при загрузке каналов связи до 60-70 %.

Наиболее популярной и широко внедренным алгоритмом технологии AQM является RED (Random Early Detection)[2]. Разработанный в 1993 году, алгоритм, несмотря на свою простоту, оказался достаточно эффективным, хотя и не лишен некоторых недостатков. Для их устранения были разработаны усовершенствованные версии данного алгоритма: Weighted RED [3], Adaptive RED [4].

Опубликованные результаты теоретических исследований алгоритмов управления очередями с потерями [5] показали, что алгоритм RED весьма близок к субоптимальному

регулятору длины очереди при условии настройки его параметров под параметры обслуживаемого потока.

Для исследования эффективности работы алгоритмов широко используется математическое и имитационное моделирование. Исследование в [5] проводилось с привлечением аппарата марковских цепей. Эффективность данного метода весьма высока, однако получаемые результаты требуют дополнительной интерпретации для применения на практике.

В работах [4] и [6] использовалась потоковая модель сети, которая базируется на нелинейных дифференциальных уравнениях. Данный математический аппарат достаточно простой и доступный в использовании, особенно при использовании численных методов решения, однако дает не совсем точные результаты, особенно при исследовании быстропеременных потоков трафика.

Имитационное моделирование в пакете ns2 позволяет проверить работу алгоритмов AQM в условиях максимально соответствующим реальным, однако построение, моделирование и обработка результатов для модели на несколько тысяч соединений весьма трудоемка и требует значительных вычислительных мощностей. Учет стохастической природы событий в процессах передачи данных дополнительно увеличит сложность модели и объемы обрабатываемых результатов.

**Цель исследования.** Исследование алгоритмов технологии AQM и проектирование регулятора для стабилизации длины очереди. Для достижения поставленной цели необходимо построение модели маршрутизатора и участка сети передачи данных, учитывающей дискретный характер объекта.

**Описание объекта.** Для проведения исследования возьмем наиболее типичную на сегодняшний день ситуацию.

Некоторое множество компьютеров-хостов объединены локальной высокоскоростной сетью. Обозначим их число через  $N$ . Они подключаются к маршрутизатору, подключенному с одной стороны к локальной сети, с другой – к низкоскоростному каналу связи.

Для создания нагрузки на низкоскоростной канал все  $N$  компьютеров передают данные по протоколу FTP на удаленный сервер. Маршрут передачи пакета с данными будет выглядеть так: хост отправителя - высокоскоростной сегмент сети – маршрутизатор – низкоскоростной сегмент сети – маршрутизатор – высокоскоростной сегмент сети – сервер-получатель. Квитанция, подтверждающая получение пакета, продельывает тот же путь в обратном порядке.

Размер квитанции обычно составляет порядка 64 байт. Это TCP-пакет инкапсулированный в IP-пакет с пустым полем данных. В нем изменяется лишь значения в заголовке. Благодаря малому размеру квитанций можно считать, что перегрузка линий связи в обратном направлении маловероятна и размер очереди маршрутизатора, установленного со стороны сервера, всегда близок нулю.

Так же такое подключение позволяет избежать перегрузок маршрутизаторов на переходе из низкоскоростной сети в высокоскоростную на стороне сервера. То есть в сети с такой топологией будет лишь одна очередь. На рис. 1 она будет на «Маршрутизаторе 1» на направлении передачи из сегмента I в сегмент II.

Схема сети будет иметь вид:

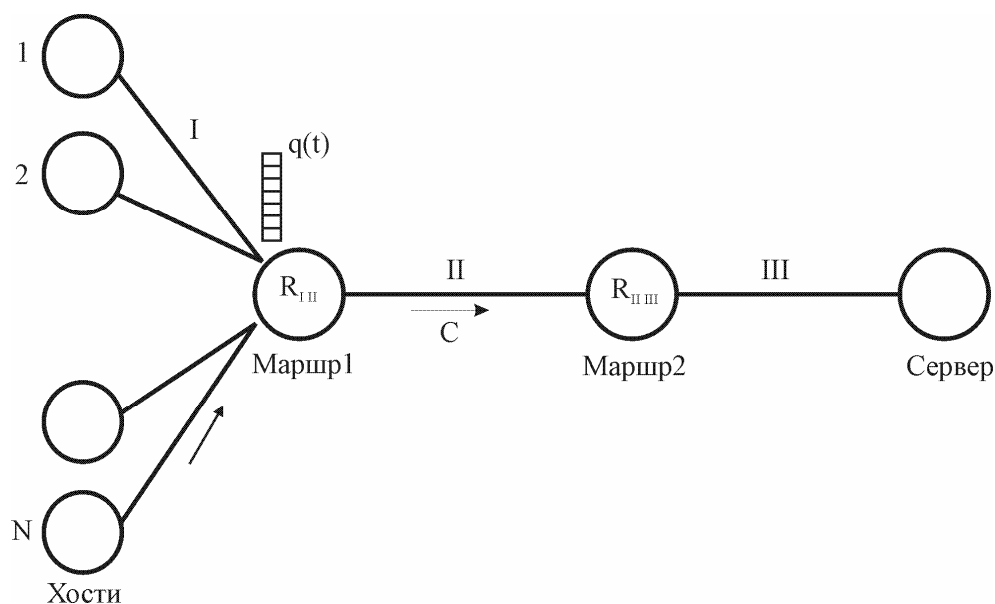


Рисунок 1 – Схема модели сети

Рассмотрим схему сети подробнее и проведем её анализ.

Сегмент I – высокоскоростная сеть, соединяющая источники с маршрутизатором. Для определенности примем её пропускную способность  $VI=100$  Мбит/с. По этому участку пакеты передаются в маршрутизатор R<sub>II</sub>, в котором они буферизируются и передаются на следующий участок.

Сегмент II – низкоскоростной участок сети. Примем его пропускную способность равной  $VI=10$  Мбит/с. Он соединяет два маршрутизатора R<sub>II</sub> и R<sub>III</sub>.

Сегмент III – снова высокоскоростная сеть, соединяющая маршрутизатор и принимающий сервер. Как и для участка I, примем её пропускную способность  $VI=100$  Мбит/с.

Так как пропускная способность участка II гораздо ниже пропускной способности участков I и III, то этот участок представляет собой так называемое «бутылочное горлышко». Через него должен протекать «тонкой струйкой» тот поток информации, который передается из сегмента I в сегмент III.

Модель типовой сети определена, теперь необходимо проанализировать их и составить математические модели.

**Основная часть.** На начальном этапе анализа воспользуемся методом «черный ящик». Это позволит определить входные и выходные переменные. После этого перейдем к «серому ящику» и определим переменные состояния и взаимодействие между ними.

Структурная схема пограничного маршрутизатора разработана в работе [7]. Он представляет собой входной и выходной буферы и блок обработки. Для схемы, представленной на рис.1, блок обработки все поступающие пакеты будет передавать со входа на выход с определенной задержкой обработки. Примем, что производительность маршрутизатора гораздо выше, чем интенсивность поступающего трафика. Тогда на входном буфере очередь будет пренебрежимо мала. И маршрутизатор как динамическое звено будет представлен лишь выходной очередью. Представим маршрутизатор как объект управления. Для этого необходимо выделить входы, выходы и возмущающие воздействия на объект.

Входы маршрутизатора как объекта управления. Можно выделить три канала воздействия на выходную очередь: 1) максимальная длина очереди  $Q_{max}$ ; 2) интенсивность исходящего потока  $C$ ; 3) маркированный вес пакета при постановке в очередь  $K_q$ .

Выходами маршрутизатора будут: 1) текущая длина очереди пакетов в выходном буфере  $q$ ; 2) сигнал об отбрасывании пакета  $lost$ , передается на каждый источник

индивидуально. Соответственно, сколько источников передачи данных работают в качестве нагрузки на маршрутизатор, столько выходов lost будет у маршрутизатора.

Возмущающими воздействиями на маршрутизатор будут: 1) интенсивность поступающего на вход маршрутизатора потока трафика от каждого источника  $\chi_i$ ; 2) поток случайных равномерно распределенных чисел  $\omega_i$ , определяющих вероятность отбрасывания пакета для каждого из источников.

$$\omega_i(k) = \text{rand}(\overline{0,1}) \quad (1)$$

Внутри маршрутизатора можно выделить два блока. Первый отвечает за моделирование работы буфера маршрутизатора, второй – за генерацию сигналов о отбрасывании пакета соответствующего источника при переполнении буфера.

Полученная «серая» модель представлена на рис.2.

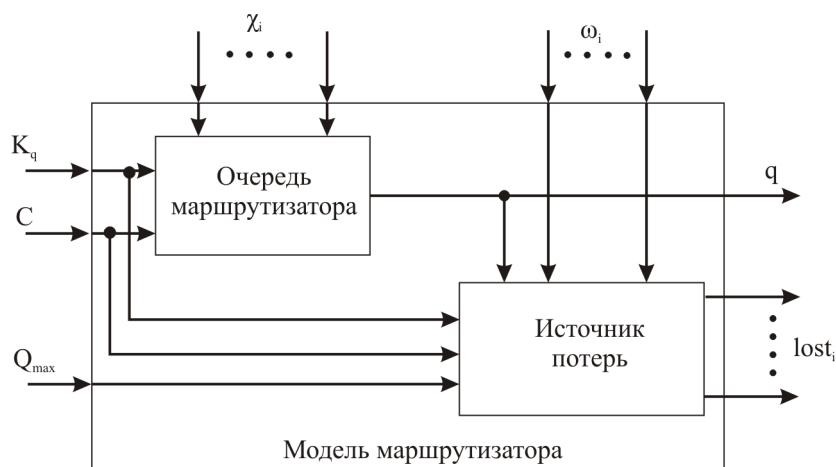


Рисунок 2 – Модель серого ящика

Функция технологии AQM – управление очередью, то есть контроль её размера и осуществление определенных действий при её перегрузке. Она заключается в маркировании пакетов определенным весом и осуществляет их помещение в очередь/отбрасывание в зависимости от веса. Наиболее распространенные на сегодня алгоритмы AQM – это Drop-Tail(DT) и RED.

С точки зрения ТАУ эта функция заключается в следующем: при прибытии пакета по линии связи маршрутизатор вычисляет вес пакета и на основе его значения принимает решение поместить его в буфер на обработку либо отбросить. Используются два основных алгоритма вычисления веса пакета: метод «отсечения хвоста» (drop tail) и метод RED. Опишем математически оба метода.

Обозначим вес пакета как  $p$ . Тогда:

- Drop tail

$$p = \begin{cases} 0, & \text{если } q < Q \\ 1, & \text{если } q \geq Q \end{cases} \quad (2)$$

- RED (с учетом пороговых значений  $T_{min}$  и  $T_{max}$ )

$$p = \begin{cases} 0, & 0 \leq q \leq T_{min} \\ \frac{q - T_{min}}{T_{max} - T_{min}} p_{max}, & T_{min} \leq q \leq T_{max} \\ 1, & T_{max} < q \end{cases} \quad (3)$$

Кроме буферизации маршрутизатор пересобирает пакет, проверяет его контрольную сумму, определяет его дальнейший маршрут. На это требуется дополнительное время, поэтому введем такой параметр как время обработки пакета маршрутизатором. Обозначим его  $T'_{router}$  и примем одинаковым для всех маршрутизаторов для всех пакетов одного размера.

Тогда задержка, вносимая маршрутизатором, во время передачи пакета будет состоять из двух частей: времени стояния в очереди и времени обработки  $T'_{router}$ . Если выходная пропускная способность маршрутизатора  $C$ , тогда общее время задержки пакета в маршрутизаторе составит:

$$T_{router} = T'_{router} + \frac{q}{C} \quad (4)$$

Построим дискретную модель маршрутизатора. В качестве времени дискретизации примем время передачи пакета минимального размера по сети с максимальной пропускной способностью. Отправка пакета размером 64 байта в сети с пропускной способностью 100 Мбит/с займет  $64 \cdot 8 / (100 \cdot 10^6) = 5,12 \cdot 10^{-6}$  [с]. Именно это время и примем за шаг дискретизации в модели:  $\Delta t = 5,12 \cdot 10^{-6}$  [с]. Пакеты большего размера будут передаваться в течение нескольких тактов.

Тогда уравнение очереди маршрутизатора примет вид.

$$q(k+1) = q(k) + K_q \sum_i \chi_i - C \quad (5)$$

Для блока «Источник потерь» выходной сигнал в импульсной форме будет создаваться нелинейным элементом на выходе.

$$z = N(x) = \begin{cases} 1 & x > 0 \\ 0 & x \leq 0 \end{cases} \quad (6)$$

Сигнал по каждому каналу будет описываться следующим выражением:

$$lost_i(k+1) = N \left( \left( 1 - K_q + K_q \left( \frac{q(k) - q_{max}}{N} \right) \cdot N(q(k) - q_{max}) \right) - \omega_i(k) \right) \quad (7)$$

Уравнения, описывающие работу передатчиков ТСР, описаны в работе [10]. Приведем их здесь в дискретной форме:

$$K_{wi}(k+1) = K_{wi}(k) + u_{RTT_i}(k) - \frac{1}{2} K_{wi}(k) \cdot u_{lost_i}(k) \quad (8)$$

$$T_i(k+1) = T_i(k) + d_i(k) - T_i(k) \cdot RTT_i(k) \quad (9)$$

$$\chi_i(k) = N(K_{pwm_i} \cdot d_i(k) \cdot K_{wi}(k) - T_i(k)), \quad (10)$$

где  $K_{wi}$  – переменная состояния, ширина окна передачи каждого источника,

$T_i(k)$  – переменная состояния, время работы передатчика ТСР,

$RTT_i(k)$  – вход обратной связи, сигнал подтверждения доставки данных,

$u_{lost_i}(k)$  – вход обратной связи, сигнал о отбрасывании пакета маршрутизатором,

$d_i(k)$  – вход, разрешение работы передатчика,

$K_{P_{WM}i}$  – масштабный коэффициент, определяет размер передаваемых пакетов,  
 $\chi_i(k)$  – выход, интенсивность передачи источника.

Для оценки времени оборота кадра и генерации сигнала подтверждения составим уравнения определения времени доставки и генерации сигнала на основе использованного ранее нелинейного элемента.

Время работы передатчика  $T_{TRi}(k)$  можно оценить:

$$T_{TRi}(k+1) = T_{TRi}(k) + d_i(k) \quad (11)$$

Момент доставки будет определяться исходя из момента получения подтверждения доставки предыдущего пакета плюс время доставки отправляемого. Время доставки отправляемого определим как сумму минимального времени доставки  $RTT_{\min i}$  (определяется характеристиками сетевого оборудования и каналами связи) и временем ожидания в очереди (см. ()).

$$RTT_i(k+1) = RTT_i(k) + RTT_i(k) \cdot \left( RTT_{\min i} + \frac{q(k)}{C} \right) \quad (12)$$

Сигнал подтверждения доставки генерируется, когда время работы передатчика становится больше времени работы передатчика:

$$RTT_i(k+1) = N(T_{TRi}(k) - RTT_i(k)) \quad (13)$$

Реализовав описанные уравнения в пакете MATLAB Simulink, получим следующую модель с передатчиком, маршрутизатором и регулятором RED.

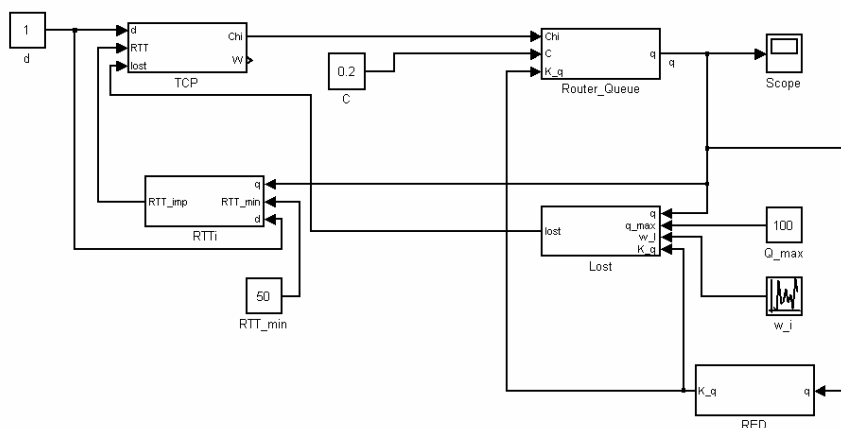


Рисунок 3 – Схема модели в MATLAB Simulink

Через пограничные маршрутизаторы проходят потоки от нескольких десятков, сотен, а то и тысяч соединений. Моделировать несколько десятков соединений в MATLAB Simulink не удобно, поэтому разработана программная модель на языке программирования MATLAB. Размер модели в ней ограничен вычислительной мощностью используемой ЭВМ.

Для сравнительной оценки эффективности алгоритма RED произведен синтез регулятора для решения задачи стабилизации текущей длины очереди на заданном уровне.

Синтеза регулятора проведен по модели объекта управления – очередь маршрутизатора. Она представляет собой дискретный интегратор. В качестве регулятора будем использовать дискретное ПД-звено. Параметры звена выберем согласно рекомендациям из [11]. Задача регулятора – стабилизировать текущую длину очереди на уровне 0,7 от  $Q_{max}$ .

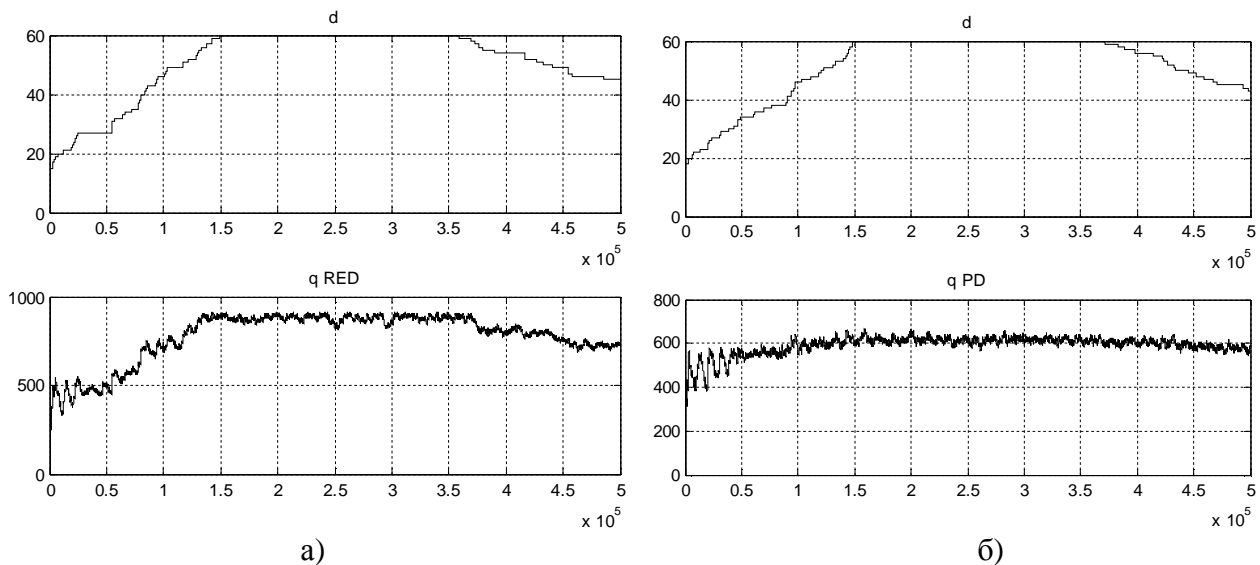


Рисунок 4 – Моделирование работы регуляторов AQM (Очередь – 900 ячеек)

При плавно возрастающем количестве источников получим следующие результаты моделирования (см . рис. ). Алгоритм RED плавно увеличил текущую длину очереди до максимально возможной и удерживал согласно уровню поступающей нагрузки. ПД-регулятор вне зависимости от уровня поступающей нагрузки удерживал текущую длину \*череди в диапазоне от 590 до 650 ячеек, что соответствует заданию:

$$q_{зад}=0,7 * Q_{max}=0,7* 900=630. \quad (14)$$

Таким образом, оставался запас свободного буфера для резкого увеличения количества потоков трафика.

Промоделируем режим работы сети при резком возрастании количества источников. Примем, что в заданный момент количество работающих источников увеличится вдвое. И оценим как отработают такую ситуацию оба регулятора. Максимальную длину буфера установим на уровне  $Q_{max}=2700$  ячеек. Настройки регуляторов оставим прежними.

Результаты моделирования приведены на рис. . При резком росте нагрузки алгоритм RED допускает рост очереди до максимального уровня  $Q_{max}$  и постепенно снижает до уровня, определяемого настройками регулятора. ПД-регулятор отработал резкий рост нагрузки и не допустил перегрузки очереди маршрутизатора и стабилизировал ее длину в диапазоне 1750-1950, что соответствует заданию:

$$q_{зад}=0,7 * Q_{max}=0,7* 2700=1890. \quad (15)$$

Таким образом, получено что ПД-регулятор справляется с задачей стабилизации очереди не хуже, чем алгоритм RED.

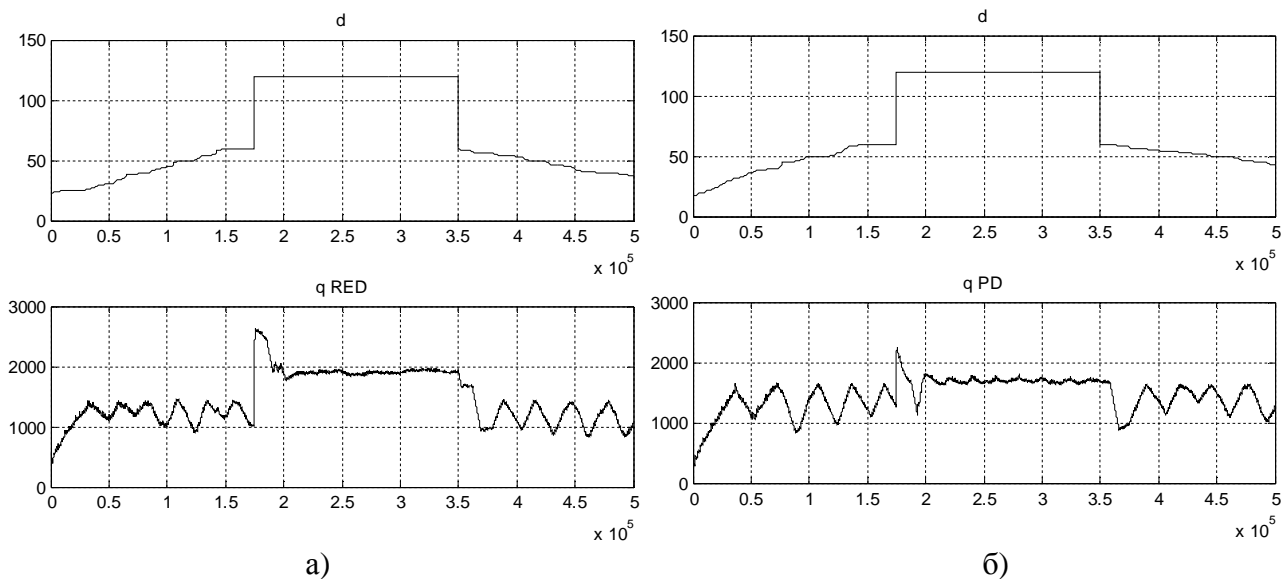


Рисунок 5 – Моделирование работы регуляторов AQM с под действием возмущений (Очередь – 2700 ячеек)

**Выводы и дальнейшие исследования.** Проведенные исследования показали, что разработанная нелинейная дискретная модель вполне адекватна и позволяет моделировать участки сети со значительным количеством источников передачи. Синтезированный ПД-регулятор справился с поставленной задачей стабилизации текущей длины очереди на заданном уровне при достаточном уровне входящей нагрузки.

Параметры качества обслуживания сети включают не только задержку доставки и её вариацию, определяемую длиной очереди маршрутизатора, но и пропускную способность соединения, вероятность потери пакета и др. ПД-регулятор стабилизировал длину очереди, но необходимо проверить, как его работа повлияла на остальные параметры качества обслуживания. Для этого необходимо использовать более общий критерий качества работы нелинейной САУ, нежели в приведенном исследовании.

## Литература

1. Michael Welzl «Network Congestion Control: Managing Internet Traffic» / Michael Welzl – Wiley, Chichester –2005 – ISBN-13 978-0-470-02528-4– 282 с.
2. Floyd, Sally "Random Early Detection (RED) gateways for Congestion Avoidance" /Floyd, Sally; Jacobson, Van – IEEE/ACM Transactions on Networking 1 (4): 397–413. doi:10.1109/90.251892 – August 1993
3. D. Clark «An Approach to Service Allocation in the Internet» / D. Clark, J. Wroclawski – Internet Engineering Task Force – INTERNET-DRAFT, MIT LCS July, 1997 – Адрес доступа: <http://tools.ietf.org/html/draft-clark-diff-svc-alloc-00>, Время доступа: декабрь 2011 г.
4. Floyd, Sally «Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management» / Floyd, Sally; Gummadi, Ramakrishna; Shenker, Scott. – AT&T Center for Internet Research at ICSI – 2001-08-01 – Адрес статьи: <http://icir.org/floyd/papers/adaptiveRed.pdf> – Время доступа: 02.12.2011



5. Миллер А. Б., “Предотвращение перегрузок в сетях передачи данных с помощью методов стохастического управления” / А. Б. Миллер – Автомат. и телемех., 2010, № 9,– с. 70–82
6. Srikant, Rayadurgam «The Mathematics of Internet Congestion Control» / Srikant, Rayadurgam.– Boston, MA, USA – 2004: – Birkhäuser. ISBN 978-0-8176-3227-4
7. Батыр С.С. Построение модели блока маршрутизации для двухпортового маршрутизатора. / С.С. Батыр// Наукові праці Донецького національного технічного університету. Сер. обчислювальна техніка та автоматизація. – Донецьк, 2008.– Вип. 14 (129).– С. 7 – 12
8. RFC 2309- Recommendations on Queue Management and Congestion Avoidance in the Internet – Адрес статьи: <http://tools.ietf.org/html/rfc2309> – Время доступа: 02.12.2011
9. RFC 793 - Transmission Control Protocol – Адрес статьи: <http://www.faqs.org/rfcs/rfc793.html>
10. Батыр С.С. «Обоснование билинейности алгоритмов управления перегрузкой протокола TCP» / С.С. Батыр, А.В Хорхордин // Наукові праці Донецького національного технічного університету. Сер. обчислювальна техніка та автоматизація. – Донецьк, 2011.– Вип. 21 (183).– С. 88 – 93
- 11 . Олсон Г. «Цифровые системы автоматизации и управления» /Г. Олсон, Д. Пиани – СПб., Невский диалект – 2001 – 557 с.

#### **Анотація**

**Батыр С.С., Хорхордин О.В. Побудова моделі мережі передачі даних для дослідження технології AQM.**

*Ділянка мережі Інтернет, що складається з передавача протоколу TCP і прикордонного маршрутизатора, розглянуто як об'єкт в теорії автоматичного управління. Проведено огляд і аналіз існуючих алгоритмів технології AQM. Виділено вхідні і вихідні змінні маршрутизатора, побудована його структурна схема. Синтезовано структурних схем та рівнянь моделі ділянки мережі. Промодельовано існуючі та запропоновано новий алгоритм для управління чергою.*

**Ключові слова:** *перевантаження, нелінійна дискретна система, алгоритм управління чергою, RED*

**Abstract Batyr S.S., Khorkhordin A.V. Building a model of data transmission network for the study of technology AQM**

*Site on the Internet, consisting of a transmitter of TCP and borderrouter is considered as an object of automatic control theory. A review and analysis of existing algorithms Technology AQM. Isolated input and output variables router built its block diagram. Synthesized by the block diagrams and equations of the model area network. Simulated current and proposed a new algorithm for queue management*

**Keywords:** *congestion, non-linear discrete system, queue management algorithm, RED*