

УДК 004

**В.И. Гостев,
С.Н. Скуртов,
О.В. Невдачина,
С.В. Кучер**

СРАВНИТЕЛЬНАЯ ОЦЕНКА PID-, PI- И RED-АЛГОРИТМОВ ДЛЯ AQM-СИСТЕМ ПРИ ПЕРЕМЕННЫХ ПАРАМЕТРАХ TCP/IP СЕТИ

В работе рассмотрены процессы в системах активного управления очередью Active Queue Management (AQM-системах) при использовании PID-, PI- и RED-регуляторов при переменных параметрах TCP/IP сети (случайном изменении нагрузки трафика – случайном изменении числа сессий TCP и случайном изменении времени следования туда и обратно – round trip time RTT) на основе интерактивной системы MATLAB.

Ключевые слова: интерактивная система MATLAB, перегрузка, методы управления очередью пакетов, маршрутизатор, PID-, PI- и RED-регуляторы.

У роботі досліджено процеси в системах активного керування чергою Active Queue Management (AQM-системах) при використанні PID-, PI- і RED-регуляторів при змінних параметрах TCP/IP мережі (випадковій зміні навантаження трафіку – випадковій зміні числа сесій TCP та випадковій зміні часу прямування у прямому та зворотному напрямках – round trip time RTT) на основі інтерактивної системи MATLAB.

Ключові слова: інтерактивна система MATLAB, перевантаження, методи керування чергою пакетів, маршрутизатор, PID-, PI- і RED-регулятори.

The processes in the systems of Active Queue Management (AQ-systems) are investigated by use of PID-, PI- and RED-controllers at variable parameters of a TCP/IP network on the basis of an interactive system MATLAB.

Keywords: interactive system MATLAB, overload, methods of a package queue management, router, PID- PI- and RED-regulators.

Для современных телекоммуникационных систем с коммутацией пакетов характерно явление перегрузки, для борьбы с которым используют различные методы. Среди них важное место занимают методы управления очередью пакетов в маршрутизаторах. Существует два вида управления очередью: активное и пассивное. При пассивном управлении происходит отбрасывание пакетов, которые приходят тогда, когда в соответствующей канальной очереди отсутствуют свободные места. Это метод отбрасывания хвоста (*Tail Drop*) прост в реализации, но имеет ряд существенных недостатков, с которыми успешно справляются методы активного управления очередью – *Active Queue Management (AQM)*. Системы *AQM*, чтобы избежать переполнения очереди, отбрасывают или

маркують определённую часть пакетов, которые попадают в маршрутизатор, до момента переполнения соответствующей канальной очереди. Типичным примером *AQM*-систем являются системы с *PID* (*proportional-integral-derivative*)-регулятором, *PI* (*proportional-integral*)-регулятором и *RED* (*random early detection*)-регулятором [1–6]. При использовании *PID*-, *PI*- и *RED*-алгоритмов поступающие в буфер пакеты случайно отбрасываются/маркируются с вероятностью, которая зависит от длины очереди. В работах [1–6] описаны и проанализированы линеаризованные системы *AQM* с этими алгоритмами как системы автоматического управления. Эти системы описаны передаточными функциями с постоянными параметрами, хотя реальные *AQM*-системы являются системами со случайными, стохастическими параметрами. В данной работе исследованы *AQM*-системы с *PID*-, *PI*- и *RED*-регуляторами как системы с переменными параметрами при случайном изменении нагрузки трафика (случайном изменении числа сессий *TCP*) и случайном изменении времени следования туда и обратно – *round trip time RIT* на основе интерактивной системы *MATLAB*.

Решение задачи. На рис. 1а приведена блок-схема системы активного управления очередью *Active Queue Management (AQM-системы)* с обратной связью и *AQM* законами управления (*AQM*-регуляторами – *PID*-, *PI*- и *RED*-регуляторами), подробное описание которой рассмотрено в работах [1–6]. Динамика объекта описывается передаточной функцией, которая представляет собой отношение по Лапласу переменной “длина очереди” к переменной “вероятность отбрасывания/маркировки пакета” и определена в работах [1–6] для установившегося режима как

$$G(s) = \frac{\delta q(s)}{\delta p(s)} = P(s)e^{-sR_o} = \frac{C^2 e^{-sR_o}}{(s + \frac{2N}{R_o^2 C})(s + \frac{1}{R_o})} \quad (1),$$

где C – емкость связи (пакеты/сек), $R_o = q/C + T_p$ – время следования туда и обратно – *round trip time RIT* (в сек), T_p – задержка распространения (в сек), N – коэффициент нагрузки (число *TCP* сессий).

Передаточную функцию звена запаздывания обычно аппроксимируют при помощи функции *Паде*. Для приближения *Паде* второго порядка можно записать:

$$e^{-sR_o} \approx \frac{s^2 - \frac{6}{R_o}s + \frac{12}{R_o^2}}{s^2 + \frac{6}{R_o}s + \frac{12}{R_o^2}} \quad (2).$$

С учетом (2) передаточную функцию объекта управления (1) можно записать в виде:

$$G(s) = P(s)e^{-sR_o} = \frac{C^2 (s^2 - \frac{6}{R_o}s + \frac{12}{R_o^2})}{(s + \frac{2N}{R_o^2 C})(s + \frac{1}{R_o})(s^2 + \frac{6}{R_o}s + \frac{12}{R_o^2})} \quad (3).$$

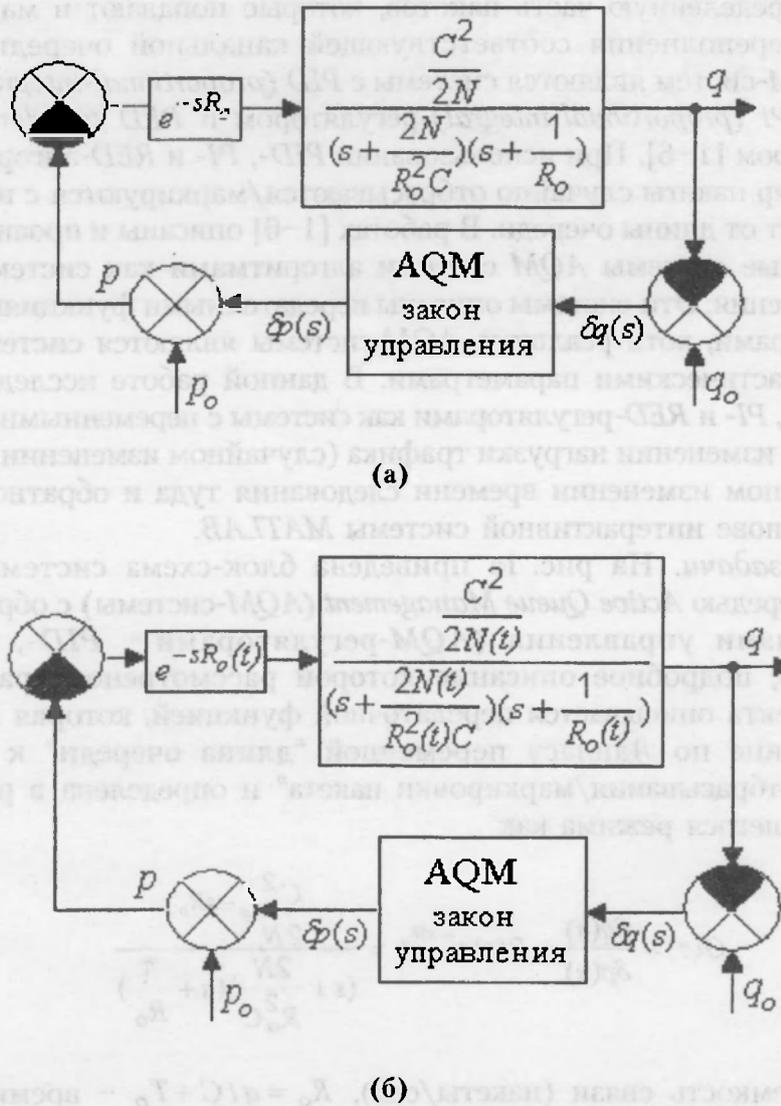


Рис. 1. Общая схема AQM-системы, скорректированной AQM-регулятором

На рис. 1а q – предполагаемая длина очереди (в пакетах), p – вероятность маркировки/отбрасывания пакетов, $\delta q = q_0 - q$, $p = p_0 + \delta p$, q_0 , p_0 – значения длины очереди и вероятности маркировки/отбрасывания пакетов в рабочей точке.

Линейная модель с передаточной функцией, описанной уравнением (3), и представленная на рис. 1а, отличается от реальной модели сети следующим:

1. Модель рассматривает только *TCP*-потоки и игнорирует другие виды потоков. Фактически, Интернет – смесь различных потоков. Некоторые источники используют механизмы управления перегрузками как *TCP*, в то время как некоторые видеоприложения принимают постоянную скорость передачи битов (*CBR*), которая безразлична к перегрузкам. Далее, различные версии выполнения *TCP*, такие как *TCP Reno*, *TCP New-Reno*, *TCP Vegas* и т.д., будут сосуществовать в Интернете. Фактически, модель точно описывает механизм управления перегрузками *TCP Reno* в установившемся режиме работы при постоянных параметрах емкости связи C , времени следования туда и обратно R_0 и коэффициенте нагрузки N .

2. Модель описує аддитивне збільшення і мультиплікативне зменшення (*AIMD*) поведінки *TCP*, в той час як ігнорує стани "повільний старт" (*slow start*) і перерив. Хоча модель точна при більшості умов, тому що запобігання перевантаженням – первинне операційне стан *TCP*, спеціально для довгоживучих передач, таких як *FTP*-потіки, але в разі короткоживучих потоків таких, як *Telnet* або *Web*, характерні стани повільного старту і перерив.

3. Основне відміння моделі від реальної моделі мережі заключається в тому, що кількість активних з'єдинень *TCP* (*N*) і час проходження туди і назад (*RTT*) в моделі на рис. 1а прийняті постійними. Однак ці параметри є в високій ступені змінними в змінюючій мережі.

Нижче розглянуті *AQM*-системи з *PID*-, *PI*- і *RED*-регуляторами як системи з змінними параметрами при випадковому зміні навантаження трафіку (випадковому зміні числа сесій *TCP* і випадковому зміні часу проходження туди і назад – *round trip time RTT*) на основі інтерактивної системи *MATLAB*.

В інтерактивній системі *MATLAB* можна представити модель об'єкта управління з'єдиненням зв'язок з змінюючимися випадковим чином параметрами і і структурну схему *AQM*-системи, скоректованої *AQM*-регулятором, зобразити в вигляді рис. 16. Конкретні моделі *AQM*-систем наведені на рис. 2 і 4.

Усилительное звено $C^2/2/N(t)$ моделюється блоками усилителем *Gain1* і делителем *Product1*, на верхній (на рис. 2 і 4) вхід якого поступає сигнал $N(t)$. Аперіодическе звено $[s + 2N(t)/R_0^2(t)/C]$ моделюється інтегратором *Integrator*, охопченим отрицательной обратной связью, которая включает умножитель *Product*, на нижній (на рис. 2 і 4) вхід якого поступає сигнал $N(t)$, делитель *Product4*, на верхній (на рис. 2 і 4) вхід якого поступає сигнал $R_0^2(t)$ і усилитель *Gain2* с коэффициентом $2/C$. Аперіодическе звено $[s + 1/R_0(t)]$ моделюється інтегратором *Integrator1*, охопченим отрицательной обратной связью, которая включает делитель *Product2*, на верхній (на рис. 2 і 4) вхід якого поступає сигнал $R_0(t)$ і усилитель *Gain4* с коэффициентом, равным единице. Звено чистого запаздывания при аппроксимации функцией Паде второго порядка (см. формулу (4)) моделюється при помощи двух интеграторов, двух усилителей с коэффициентами 6 и 12 и двух делителей на верхние (на рис. 2 і 4) входы которых поступают сигналы $R_0(t)$ и $R_0^2(t)$.

$$e^{-sR_0(t)} \approx \frac{s^2 - \frac{6}{R_0(t)}s + \frac{12}{R_0^2(t)}}{s^2 + \frac{6}{R_0(t)}s + \frac{12}{R_0^2(t)}} \quad (4)$$

Предположим, что время следования туда и обратно $R_o(t)$ изменяется случайным образом в пределах от 220 мсек до 300 мсек, а нагрузка трафика $N(t)$ изменяется также случайным образом в пределах от 40 до 80. Такие “возмущающие воздействия” на системы, приведенные на рис. 2 и 4, могут генерировать в интерактивной системе *MATLAB* блоки *Uniform Random Number*. Отметим, что в реальных сетях и время следования туда и обратно, и нагрузка трафика может изменяться случайным образом в различных пределах, но для сравнения работы рассматриваемых регуляторов выбраны одинаковые “возмущающие воздействия”, показанные на рис. 6(а) и (б).

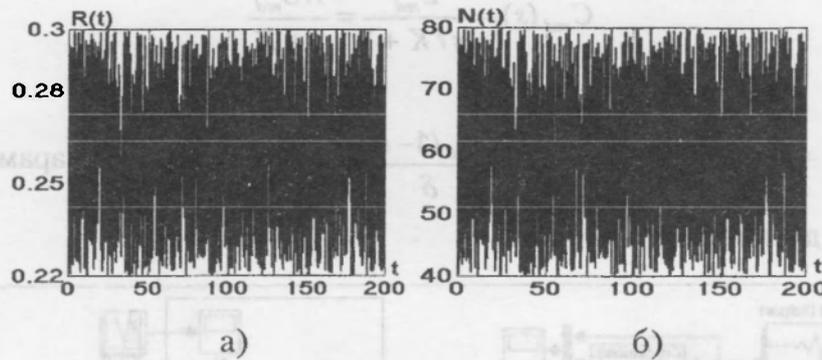


Рис.6. Время следования туда и обратно (а)
и нагрузка трафика (б)

Емкость связи выберем равной $=3750$ пакетов/сек, желаемую длину очереди примем равной $q_0=200$ пакетов.

Вначале исследуем процессы в *AQM*-системе, скорректированной *PID*-регулятором. Регулятор настраиваем так, чтобы текущая длина очереди $q(t)$ была как можно ближе к желаемой длине очереди. При этом получены следующие значения коэффициентов регулятора: $K=6,2 \cdot 10^{-5}$; $K_i=3,12 \cdot 10^{-5}$; $K_d=5 \cdot 10^{-7}$. Шаг дискретизации (шаг квантования) h_0 выбран равным 0,01 сек. Процессы в *AQM* системе, скорректированной *PID*-регулятором, приведены на рис. 7, где представлены выходная переменная *PID*-регулятора $p(t)$ – вероятность отбрасывания/маркировки пакетов (а) и выход системы $q(t)$ – текущая длина очереди (б)

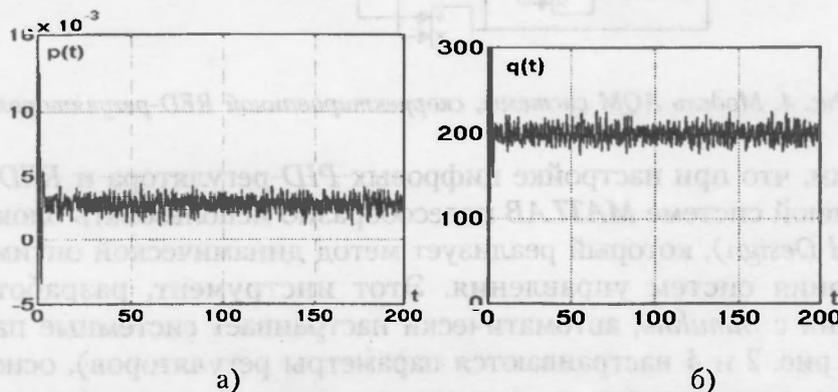


Рис. 7. Вероятность отбрасывания/маркировки пакетов и текущая длина очереди с *PID*-регулятором

Исследование системы, приведенной на рис. 2, которая скорректирована *PID*-регулятором, показывает, что выход регулятора и выход системы являются случайными, шумоподобными переменными и регулятор поддерживает текущую длину очереди около ее желаемого значения. Кроме того, поскольку коэффициент K_d на два порядка меньше чем другие коэффициенты регулятора, то можно использовать *PI*-регулятор. Процессы в *AQM*-системе, скорректированной *PI*-регулятором с параметрами $K=6,2 \cdot 10^{-5}$ и $K_i=3,12 \cdot 10^{-5}$, приведены на рис. 8, где представлены выходная переменная *PI*-регулятора $p(t)$ – вероятность отбрасывания/маркировки пакетов (а) и выход системы $q(t)$ – текущая длина очереди (б).

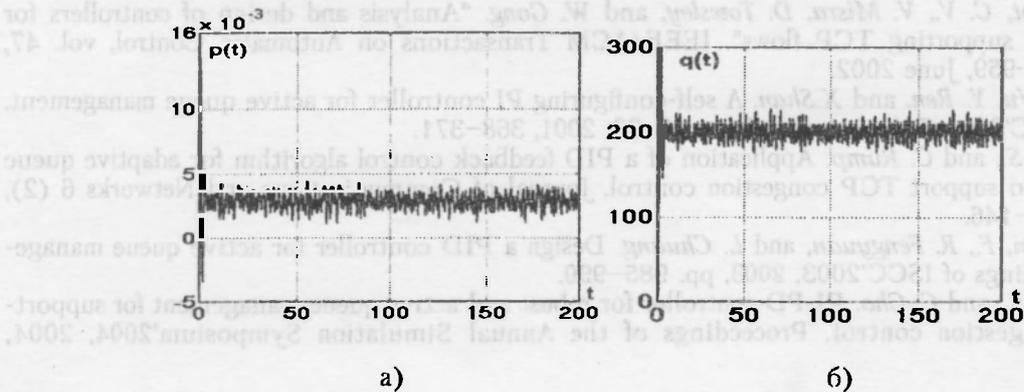


Рис. 8. Вероятность отбрасывания/маркировки пакетов и текущая длина очереди с *PI*-регулятором

Исследуем процессы в *AQM*-системе, скорректированной *RED*-регулятором. Оптимальные параметры *RED*-регулятора получены в работах [1, 2]: $\alpha_1=1,84 \cdot 10^{-4}$; $K=0,005$. Поэтому в системе на рис. 4 коэффициент $\alpha_1=0,005 \cdot 1,84 \cdot 10^{-4}$. Процессы в *AQM*-системе, скорректированной *RED*-регулятором, приведены на рис. 9, где представлены выходная переменная *RED*-регулятора $p(t)$ – вероятность отбрасывания/маркировки пакетов (а) и выход системы $q(t)$ – текущая длина очереди (б).

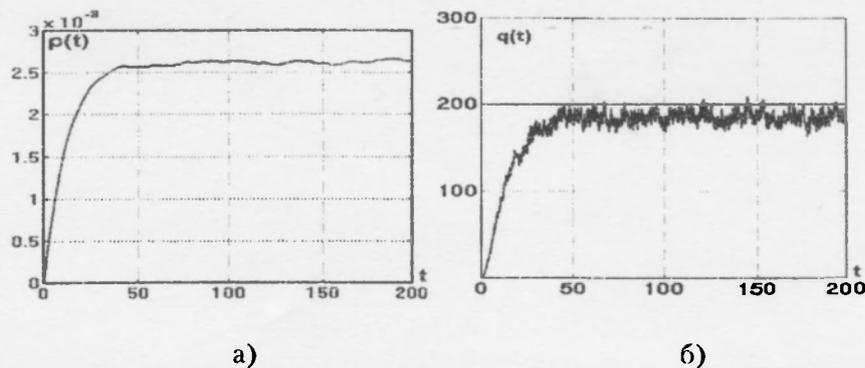


Рис. 9. Вероятность отбрасывания/маркировки пакетов и текущая длина очереди с *RED*-регулятором

Сравнивая процессы в исследуемых системах можно заключить, что *AQM*-система, скорректированная *RED*-регулятором, обладает худшими характе-

ристиками по точності и быстродействию, чем системы, скорректированные PID - и PI -регуляторами. Текущая длина очереди повторяет заданную длину очереди (желаемое значение очереди) с большой ошибкой и переходной процесс системы, скорректированной RED -регулятором, занимает почти 50 сек. Поэтому маршрутизаторы с RED -регуляторами целесообразно заменить маршрутизаторами с PI -регуляторами.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. *C.V. Hollot, Vishal Misra, Don Towsley and Wei-Bo Gong*, "On Designing Improved Controllers for Routers Supporting TCP Flows", in Proceedings of IEEE INFOCOM'2001, April 2001.
2. *Hollot, C. V., V. Misra, D. Towsley, and W. Gong*. "Analysis and design of controllers for AQM routers supporting TCP flows". IEEE/ACM Transactions on Automatic Control, vol. 47, no. 6, pp 945–959, June 2002.
3. *W. Wu, Y. Ren, and X. Shan*. A self-configuring PI controller for active queue management. Proc. of APCC'2001, Tokyo, Japan, Sept. 16–20, 2001, 368–371.
4. *Ryu, S., and C. Rump*. Application of a PID feedback control algorithm for adaptive queue management to support TCP congestion control. Journal of Communications and Networks 6 (2), 2004, pp. 133–146.
5. *Yanfei, F., R. Fengyuan, and L. Chuang*. Design a PID controller for active queue management. Proceedings of ISCC'2003, 2003, pp. 985–990.
6. *Ryu, S., and C. Cho*. PI-PD-controller for robust and active queue management for supporting TCP congestion control. Proceedings of the Annual Simulation Symposium'2004, 2004, pp. 132–139.

Отримано 5.05.2011

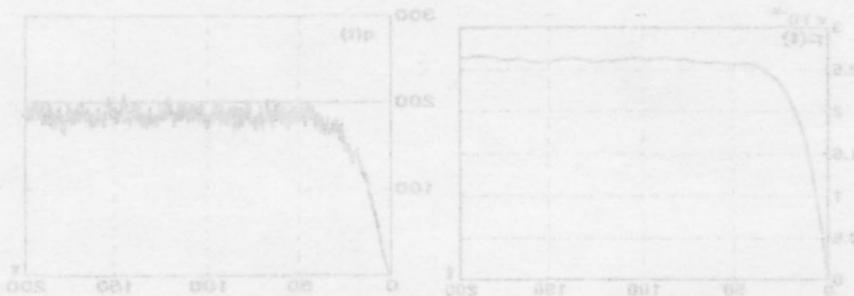


Рис. 9. Впорівнянні швидкодієвості/маршрутизатори і поточна довжина черги з RED -регулятором.