

УДК 621.372.061.3.001.63:681.3

А.В. Карпухин

Харьковский национальный университет радиоэлектроники, Харьков

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ХАОТИЧЕСКИХ ЯВЛЕНИЙ В ВЫСОКОСКОРОСТНЫХ СЕТЕВЫХ ИНФОРМАЦИОННЫХ СИСТЕМАХ С ПРОТОКОЛОМ TCP*

Вопросы хаотического поведения информационных систем с высокими скоростями передачи данных еще недостаточно изучены, несмотря на то, что наблюдаемые в них эффекты существенно влияют на характеристики таких систем (например, возникают значительные потери пакетов в TCP-соединениях).. Приведены основные результаты, полученные в этой области. Предлагаются оригинальные подходы к решению проблемы предотвращения хаотизации поведения информационных систем с высокоскоростными информационными потоками. Использован синергетический подход к моделированию высокоскоростных информационных систем. Высокоскоростные информационные системы представляются моделью ансамбля взаимодействующих нелинейных осцилляторов.

Ключевые слова: хаос, структуризация, синергетика, информационные потоки, фракталы, TCP/IP, самоподобие, математические модели.

1. Влияние самоподобия сетевого трафика на характеристики сети

Проблема самоподобия в различных областях науки и техники (гидрология, геофизика, биофизика, биология, финансовая экономика) интересовала исследователей давно (достаточно обширная библиография содержится в [1]). Появление и широкое распространение компьютерных сетей (глобальных – в 70-х годах 20 столетия, локальных – в начале 80-х годов), а также увеличение количества разнообразных сетевых услуг (WWW, Iphone и т.д.) привело к тому, что сетевой трафик стал более сложным и непредсказуемым. Особенно сильно эти свойства стали проявляться с появлением технологий высокоскоростной передачи данных. Это связано с тем, что одним из основных показателей качества (QoS) работы сетей с пакетной передачей является количество потерянных пакетов. Потеря пакетов приводит к дополнительной нагрузке на сеть и, в конечном счете, к «заторам» (congestions). При больших скоростях передачи данных потери пакетов, выражающиеся в долях процента, приводят к значительным потерям информации. Причем увеличение пропускной способности сети слабо влияет на улучшение ее работы.

Применение концепции самоподобия к телекоммуникационным системам впервые предложил Б. Мандельброт [2].

В последующих многочисленных работах, посвященных исследованию сетевого трафика, было показано, что указанные выше явления связаны со свойствами самоподобия трафика.

В последние 10-15 лет проблеме самоподобия сетевого трафика было посвящено большое число работ. Их можно условно разделить на две группы. Первая (и самая обширная) включает в себя работы, в которых авторы анализируют сетевой трафик и определяют его статистические характеристики. Источником анализируемых данных является либо натурный эксперимент (например, [3]), либо моделирование с помощью программных средств (например, ns [4], OPNET [5]).

Ко второй группе относятся работы (к сожалению, немногочисленные), в которых авторы рассматривают информационную систему как динамическую систему, в которой самоподобие является внутренним свойством самой системы [6]. Авторами использовался метод восстановления многомерных траекторий, который состоит в использовании смещенных во времени отсчетов значений трафика двух TCP-соединений ($x[i], y[i]$) в n точках. На рис. 1, 2 приведены результаты для двух наборов параметров (размер буфера, длительность задержки канала и пропускная способность канала):

$$x[i] = \frac{1}{n} \sum_{j=1}^n \text{cwnd}_x[i-j]; \quad y[i] = \frac{1}{n} \sum_{j=1}^n \text{cwnd}_y[i-j]. \quad (1)$$

Как видно из рис. 1, 2, в системе наблюдаются периодические движения, причем на рис. 1 движение простое, а на рис. 2 – более сложное. Эти траектории представляют собой простые аттракторы.

При определенных значениях параметров (рис. 3) в системе наблюдаются хаотические движения. В данном случае аттрактор системы является странным аттрактором. В этой же работе было вычислено значение т.н. экспоненты Ляпунова ($\lambda \approx 1,11$), что говорит о том, что траектории двух систем расходятся по экспоненциальному закону. Это подтверждает наличие хаотического режима.

* Данная работа выполнена при поддержке ДФФД (грант Ф25.1/027 «Аналіз поведінки інформаційних систем з великою нелінійністю: синергетичний підхід»).

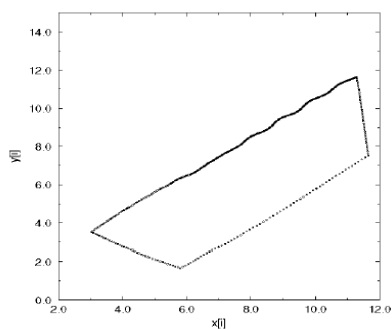
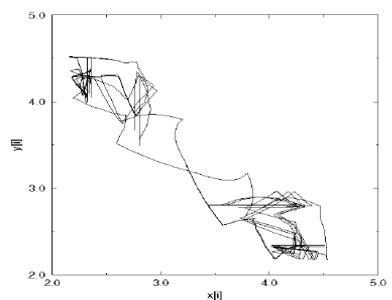


Рис. 1. Простой аттрактор



Periodic attractors of two competing TCP sources: a) $C = 0.2$ Mbps, $d = 10$ ms, $B = 20$ packets, time shifts $n = 100$. b) $C = 0.5$ Mbps, $d = 10$ ms, $B = 4$ packets, time shifts $n = 90$.

Рис. 2. Простой аттрактор

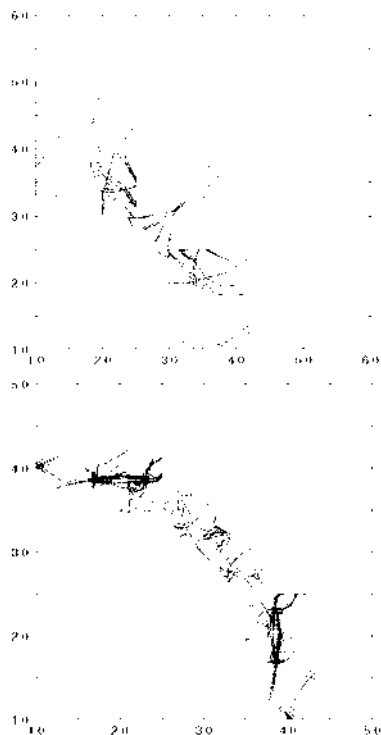


Рис. 3. Странный аттрактор

Необходимо отметить, что, как следует из работы [7], время появления TELNET-соединений и FTP-сессий статистически соответствует распределению Пуассона, в то время как появление NNTP, SMTP, FTPDATA и WWW соединений этому распределению не соответствует. Авторы установили, что время появления (arrival) TCP сессий, инициированных пользователем (таких как remote-login, file-transfer) хорошо моделируются с помощью рас-

пределения Пуассона, в то время, как моделирование появления TELNET-пакетов этим методом приводит к недооценке уровня пиковых выбросов (burstiness) TELNET-трафика. В частности, FTP-соединение для передачи данных (FTP data connection) в рамках одной сессии проявляет ярко выраженный «взрывной» (burst) характер.

В работах многих исследователей отмечается, что агрегированный сетевой трафик является самоподобным (или фрактальным) [1, 7–9]. Показано, что РТХ (распределение с тяжелыми хвостами, heavy tailed distribution) размеров файлов, появления пакетов и длительности передачи вносят основной вклад в самоподобную природу агрегированного сетевого трафика. Очевиден и тот факт, что сам TCP является первичной причиной самоподобия и такое его поведение может иметь «ужасные» последствия в компьютерных сетях при увеличении пропускной способности глобальных вычислительных сетей (WAN) до значений, выражающихся в нескольких гигабайтах в секунду. В частности, даже если трафик, генерируемый приложениями, имеет параметр Херста $H = 0,5$ (т.е. является несамоподобным) TCP модулирует этот трафик и делает его самоподобным (имеющим параметр Херста $H = 1,0$). Причем все существующие реализации протокола TCP (Reno, Vegas, Tahoe) обладают этим «нежелательным» свойством.

Для анализа сетевого трафика используются различные методики. Основным параметром для определения степени самоподобия, как известно, является параметр Херста. Кроме того, могут быть использованы следующие подходы: вычисление с.о.в (coefficient of variation) [9], либо вычисление т.н. cwnd (congestion window) [6, 10].

Есть три главных компонента контроля заторов в быстродействующих сетях: предупреждение, предотвращение, и восстановление. Предотвращение заторов состоит в проектировании и построении сети, которая минимизирует вероятность того, что затор произойдет, и это включает в себя правильно проектируемые компоненты (с учетом ожидаемого изменения трафика на временных интервалах часов, дней, или месяцев), хорошо разработанные алгоритмы маршрутизации и, наконец, правильно выбранную сетевую политику для того, чтобы гарантировать, что скорость линии доступа пользователя не превышает максимальную скорость, указанную в абонентском пакете пользователя. Предотвращение заторов – в основном проблема предсказания и включает в себя обнаружение того факта, когда затор неизбежен, и выполнение действий, которые его предотвращают. Восстановление – это действие, предпринятое сетью после того, как ухудшение работы обнаружено для того, чтобы как можно более уменьшить влияние затора. Чтобы знать, какие действия необходимо предпринять для восстановления, важно знать среднее время существования затора и какие виды потерь пакетов (или задержки) возникают во время заторов.

Моделирование трафика TCP с использованием модели Пуассона или других моделей, которые не отражают LRD-зависимость (Long Range Dependence) в реальном трафике, приводит к результатам, которые значительно недооценивают измеряемые критерии качества работы, такие как средняя задержка пакета или максимальный размер очереди.

В работе [11] исследуются пиковые выбросы в широком диапазоне масштабов временной оси, и обсуждается их воздействие на заторы в сети. Авторы установили, что периоды заторов могут быть весьма длинными, с потерями пакетов, которые в большей степени сконцентрированы; в отличие от транспортных моделей Пуассона, линейные увеличения размера буфера не приводят к большим уменьшениям количества отброшенных пакетов; небольшое увеличение числа активных соединений может привести к большому увеличению уровня потерь пакетов. Авторы приходят к заключению, что, потому что значение уровня трафика в ON-период непредсказуемо, затруднительно эффективно масштабировать сеть, чтобы уменьшить заторы. В работе отмечается, что, в отличие от моделей Пуассона, в действительности трафик имеет свойство LRD (что вызывает фактические потери пакетов). Авторы предлагают использовать «фильтрованную» переменную, чтобы обнаружить низкочастотную компоненту частоты заторов, и, таким образом, давать предупреждение прежде, чем потери пакетов становятся существенными.

Другой многообещающий подход к моделированию сетевого трафика был предложен Erramilli, Singh, Pruthi [12]. Он основан на детерминированных хаотических картах (deterministic chaotic maps, точечных отображениях в русской терминологии). Хаос – это поведение динамической системы, в котором простые нелинейные детерминированные уравнения невысокого порядка, описывающие систему, могут привести к поведению, которое подражает вероятностным процессам. Чтобы проиллюстрировать основную идею, рассмотрим нелинейную карту $f(\cdot)$, которая описывает изменение переменной x_n в дискретные моменты времени как $x_{n+1} = f(x_n)$. Процесс генерирования трафика отдельного источника может теперь быть смоделирован, предусматривая, что источник либо производит, либо не производит пакет во время n в зависимости от того, является ли x_n выше или ниже соответственно выбранного порога.

Если $f(\cdot)$ – хаотическая карта, то получающийся в результате ее применения трафик может подражать сложным явлениям. В частности Erramilli и Singh (1992) [13] показали, что простая двухпараметрическая нелинейная хаотическая карта, называемая скачкообразной картой, может обладать многими из вышеупомянутых фрактальных свойств при измерениях реального трафика.

Очевидно, что получение трафика с помощью нелинейных хаотических карт делает привлекательным такой подход к моделированию динамических

систем. Как только соответствующая хаотическая карта будет получена из ряда измерений трафика, генерирование потока пакетов для индивидуального источника становится быстрым и легким. С другой стороны, получение соответствующей нелинейной хаотической карты, основанной на ряде измерений реального трафика, в настоящее время требует значительных экспериментальных затрат; развитие более строгих статистических методов оценки для динамических систем недавно привлекло значительное внимание в литературе. Необходимо отметить, что побуждением для обоих подходов, самоподобного стохастического моделирования и детерминированного нелинейного моделирования, является желание получить относительно простое описание сложного сетевого трафика. Кроме того, оба подхода к моделированию приводят к единственному параметру, который описывает фрактальную природу трафика (параметр Hurst для самоподобных моделей, и фрактальных размерностей для моделей хаотических карт) и охватывает интуитивное понятие "взрыва" (burst), где обычные способы измерений не применимы. Однако, связь на математическом уровне между параметром Херста и соответствующей ему фрактальной размерностью не всегда ясна. Наконец, в целях анализа технических характеристик оба подхода предъявляют новые и сложные проблемы. В то время как традиционное моделирование в основном использует стохастические модели входящего трафика, изучение очередей в потоках прибытия пакетов, которые сгенерированы с помощью нелинейных хаотических карт, может вполне обеспечить новое понимание работы систем организации очередей, где процессы прибытия пакетов обнаруживают фрактальные свойства.

2. Причины самоподобия сетевого трафика

Большинство исследователей считают, что основной причиной самоподобия трафика является TCP (основной транспортный протокол Internet). Чтобы понять механизм воздействия протокола TCP на самоподобие сетевого трафика, необходимо описать в общих чертах алгоритм работы TCP [8, 9, 23].

TCP – сервис, ориентированный на соединение, который гарантирует надежную (и в правильной последовательности) доставку потока байтов, освобождая приложения от необходимости «волноваться» о без вести пропавших или повторно заказанных данных. Он включает механизм контроля потока, который гарантирует, что отправитель не переполняет буфер приемника и механизм контроля заторов, который пробует препятствовать попаданию слишком большого объема данных в сеть (что приводит к потерям пакетов). В то время как размер окна управления потоком данных является статическим, размер окна затора изменяется в течение долгого времени в соответствии с состоянием сети.

Существует несколько модификаций реализации протокола TCP (Reno, Vegas, Tahoe). Самой рас-

пространенной является, очевидно, TCP Reno [15].

Его механизм управления заторами состоит из двух фаз: (1) медленный старт (low start) и (2) предотвращение заторов (congestion avoidance). В фазе медленного старта, размер окна затора увеличивается по экспоненте (то есть, удваивается каждый раз, когда отправитель успешно передает в сеть количество байтов, равное текущему размеру окна затора), пока не возникает timeout, что означает потерю пакета. В этот момент значение *Threshold* устанавливается в размере половины окна. TCP Reno перезагружает размер окна затора (congestion-window) к одному и возвращается в фазу медленного старта, увеличивая окно затора экспоненциально до значения *Threshold*. Когда порог достигнут, TCP Reno тогда входит в фазу предотвращения затора, в которой окно затора увеличивается на один пакет каждый раз, когда отправитель успешно передает содержимое окна затора в сеть. Когда пакет потерян в течение фазы предотвращения затора, TCP Reno предпринимает те же самые действия как тогда, когда пакет потерян в течение медленного старта (low start).

TCP Reno в настоящее время также осуществляет быструю повторную передачу, а также реализует механизм быстрого восстановления для обеих фаз предотвращения затора и медленного старта. Вместо того чтобы реализовывать ситуацию с timeout при ожидании подтверждения (ACK) потерянного пакета, если отправитель получает три двойных ACKs (указание, что некоторый пакет был потерян, но более поздние пакеты были получены), отправитель немедленно повторно передает потерянный пакет (fast retransmit). Поскольку более поздние пакеты были получены, затор в сети, как предполагают, является менее серьезным, чем если бы все пакеты были потеряны, и отправитель только уменьшает наполовину его окно затора (congestion window) и повторно возвращается в фазу предотвращения затора (fast recovery), не проходя фазу медленного старта (low start) снова.

TCP Vegas [16] вводит новый механизм управления заторами, который пробует их предотвратить, а не реагировать на возникшие заторы. Когда окно затора увеличивается в размере, ожидаемая скорость передачи (ER) также увеличивается. Но если фактическая скорость передачи (AR) остается примерно такой же, это будет означать, что доступная полоса пропускания недостаточна, чтобы обеспечить скорость передачи ER, и таким образом, любое увеличение размера окна затора приведет к переполнению буфера маршрутизатора в узком месте (bottleneck gateway). TCP Vegas пытается обнаружить это явление и избежать затора на маршрутизаторе в узком месте, регулируя размер окна затора, и, следовательно, ER, приспосабливаясь к доступной полосе пропускания. Для того чтобы регулировать окно затора соответствующим образом, TCP Vegas определяет две пороговые величины, α и β , для фазы предотвращения затора, и третью пороговую величину γ , для перехода между фазами предотвра-

щения затора и медленного старта. Концептуально, $\alpha = 1$ подразумевает, что TCP Vegas пробует держать, по крайней мере, один пакет из каждого потока входящегося в очереди на маршрутизаторе, в то время как $\beta = 3$ держит самое большее три пакета. Если $\text{Diff} = \text{ER} - \text{AR}$, то, когда $\text{Diff} < \alpha$, Vegas увеличивает окно затора линейно в течение следующего RTT (Round Trip Time); когда $\text{Diff} > \beta$, Vegas уменьшает окно линейно в течение следующего RTT; иначе, окно скопления остается неизменным. Параметр γ может рассматриваться как "начальное" значение, когда TCP Vegas входит в его фазу предотвращения затора. Чтобы еще более улучшить характеристики TCP, Флойд и др. предложили использование маршрутизаторов RED (Random Early Detection) [17], чтобы обнаружить начинающийся затор. Маршрутизаторы RED поддерживают взвешенное среднее длины очереди. Пока средняя длина очереди остается ниже минимального порога (min_{th}), все пакеты находятся в очереди. Когда средняя длина очереди превышает (min_{th}), пакеты пропускаются (теряются) с вероятностью P . И когда средняя длина очереди превышает максимальный порог (max_{th}), все прибывающие пакеты теряются.

В последние годы были предложены несколько новых модификаций протокола TCP – Binary Increase Control TCP (BIC-TCP), CUBIC TCP, Westwood TCP (TCPW), Parallel TCP Reno (P-TCP), Scalable TCP (S-TCP), Fast TCP, HighSpeed TCP (HS-TCP), HighSpeed TCP Low Priority (HSTCP-LP), Hamilton TCP (H-TCP), Yet Another Highspeed TCP (YeAH-TCP), Africa TCP, Compound TCP и т.д...

Почти все они базируются на известных старых версиях TCP и отличаются различными способами предотвращения заторов (точнее, различными способами определения факта наличия потери пакетов, что означает возникновение затора). В различных модификациях используются различные формулы для расчета cwnd (congestion window).

В частности, ученые из Университета Северной Каролины разработали сетевой протокол, который, по их замыслу, призван сменить устаревший TCP. Создававшийся в 80-х годах прошлого столетия стандарт TCP был рассчитан на совершенно другие линии связи и намного меньшие объемы информации, нежели те, которыми приходится оперировать рядовым пользователям Internet и научным организациям сегодня. В TCP возможны длительные временные задержки, кроме того, данный протокол не обеспечивает необходимую масштабируемость и надежность. В новом стандарте, названном учеными BIC-TCP (сокращенно от Binary Increase Congestion Transmission Control Protocol), устранены все недостатки, присущие традиционному TCP. В частности, протокол BIC-TCP обеспечивает примерно в 6000 раз более высокую скорость передачи информации по широкополосным каналам и в 150 раз большую пропускную способность, нежели коммутируемые линии связи, использующие TCP. Ключевой особенностью BIC-

TCP является применение двоичного поиска для быстрого выявления оптимального пути передачи данных с минимальными потерями. Плюс к этому, в ВИС-TCP существенно сокращены объемы передаваемой служебной информации за счет применения пакетов большего объема. В результате, по словам исследователей, те задачи, на выполнение которых в TCP уходит несколько часов, в ВИС-TCP могут быть обработаны буквально за минуты, а то и секунды.

3. Новые математические подходы

Алгоритм работы взаимодействующих TCP-соединений может быть описан, как ансамбль нелинейных математических маятников, в котором каждое TCP-соединение выступает в роли такого маятника [18,1 9]. В этом случае n TCP-соединений можно описать следующей системой дифференциальных уравнений:

$$\begin{aligned} x_1'' + \omega_1^2 \sin x_1 &= f_1(x_2, x_3, \dots, x_n); \\ x_2'' + \omega_2^2 \sin x_2 &= f_2(x_1, x_3, \dots, x_n); \\ &\dots \\ x_n'' + \omega_n^2 \sin x_n &= f_n(x_1, x_2, x_3, \dots), \end{aligned} \tag{2}$$

где x_i – количество пакетов в секунду в i-ом TCP-соединении; f_n – функция, определяющая взаимное влияние TCP-соединений друг на друга; ω_i^2 – “собственная частота” TCP-соединения, которая зависит, в первую очередь, от размера буфера, пропускной способности и задержки канала т.н. “узкого горлышка” (bottleneck). В первом приближении можно считать, что все ω_i равны, т.к. все TCP-соединения совместно используют линию связи и буфер маршрутизатора. Отличие состоит в фазе колебаний маятников, описывающих отдельные TCP-соединения (рис. 4). Хотя в действительности эти частоты различаются, т.к. отдельные TCP-соединения имеют различные значения RTT.

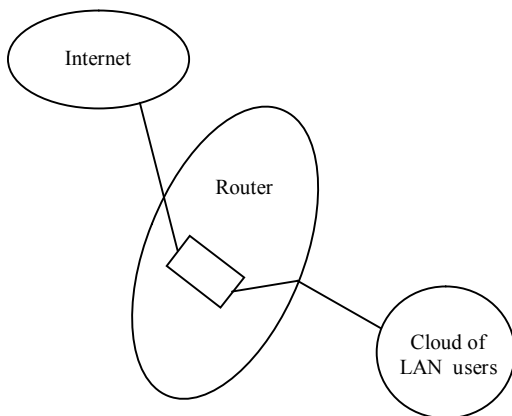


Рис. 4. Топология тестовой сети

Функции f_i можно описать в виде $\alpha \prod_1^n x_k$, где в каждом произведении для соответствующей функции отсутствует член с номером i, а α опре-

деляет степень влияния различных TCP-соединений друг на друга. В первом приближении можно считать эту величину одинаковой для всех TCP-соединений, но в действительности эта величина также, очевидно, зависит от RTT различных TCP-соединений.

Усредненный трафик в каждый момент времени в данной конфигурации сети можно вычислить как

$$\hat{X} = \frac{1}{n} \sum_{i=1}^n x_i(t), \tag{3}$$

где $x_i(t)$ определяются в результате решения системы уравнений (2).

Фазовый портрет для любых двух TCP-соединений может быть построен в результате решения системы (2), т.е. фазовые траектории определяются, как кривые в параметрической форме $x_k(t), x_l(t)$. В этом случае будет учтено влияние остальных TCP-соединений на два соединения, выбранных для построения фазового портрета.

Собственная частота TCP-соединения, очевидно, может быть определена как

$$\omega^2 = CB / t_c,$$

где C – скорость передачи (Мб/с); B – размер буфера (в пакетах); t_c – задержка канала (в мс).

В глобальном масштабе всей сети Internet решить проблему заторов и потерь пакетов, очевидно, не представляется возможным в связи с тем, что перестроить всю сеть нельзя в силу технических и экономических причин.

Однако, в ограниченных по размерам сетях (даже довольно больших) возможно дать рекомендации по проектированию (и дальнейшей эксплуатации) таких сетей, которые позволят свести к минимуму отрицательные явления хаотизации.

Выводы

Вопросы хаотического поведения информационных систем с высокими скоростями передачи данных еще недостаточно изучены, несмотря на то, что наблюдаемые в них эффекты существенно влияют на характеристики таких систем (например, имеют место значительные потери пакетов в TCP-соединениях). Однако, до сих пор не существуют математические модели, позволяющие не только провести анализ поведения информационных систем с большой нелинейностью, но также дать рекомендации по проектированию таких систем, обладающих минимумом отрицательных свойств в плане производительности.

Единственно правильный путь, на наш взгляд, состоит в том, чтобы описывать информационные системы с высокими скоростями передачи данных, использующие протоколы TCP/IP, как нелинейные динамические системы. В этом случае есть возможность применить весь арсенал классических методов

анализа поведения таких систем, разработанный Пуанкаре, Ляпуновым, Биркгофом и т.д. Речь идет о качественном анализе поведения динамических систем в фазовом пространстве, что дает возможность определить все возможные режимы движения (работы) системы, а также определить значения параметров, при которых в системе наблюдаются нежелательные хаотические явления (потери пакетов, снижение производительности). Эти вопросы являются актуальными в первую очередь для ISP (провайдеров Internet), которые, к сожалению, до сих пор не обращают на них должного внимания.

Список литературы

1. Willinger W. *A bibliographical guide to self-similar traffic and performance modeling for modern high-speed networks, Stochastic Networks: Theory and Applications (Oxford) (F.P. Kelly, S. Zachary, and I. Ziedins, eds.) / W. Willinger, M. S. Taqqu, A. Erramilli // Royal Statistical Society Lecture Notes Series. – Oxford University Press, 1996. – Vol. 4. – P. 339-366.*
2. Mandelbrot B.B. *Self-similar error clusters in communications systems and the concept of conditional systems and the concept of conditional stationarity / B.B. Mandelbrot // IEEE Transactions on Communications Technology. – 1965. – COM-13. – P. 71-90..*
3. *On the self-similarity of ethernet traffic / W.E. Leland, M.S. Taqqu, W. Willinger, D.V. Wilson // IEEE/ACM Transactions of Networking. – 1994. – 2 (1). – P. 1-15.*
4. S. Floy. *Simulator tests. Available in [ftp://ftp.ee.lbl.gov/papers/simtests.ps.Z](http://ftp.ee.lbl.gov/papers/simtests.ps.Z) ns is available at <http://www-nrg.ee.lbl.gov/1995>.*
5. *A comparison of active queue management algorithms using the OPNETModeler / C. Zhu, O.W.W. Yang, J. Aweya, M. Oullete, D.Y. Montuno // IEEE Communication Magazine. – 2002. – 40 (6). – P. 158-167.*
6. Veres A. *The chaotic nature of TCP congestion control / A. Veres, V. Boda // In Proc. IEEE INFOCOM. – 2000. – P. 161-165.*
7. Paxson V. *Wide-Area Traffic: The Failure of Poisson Modeling / V. Paxson, S. Floyd // IEEE/ACM Transactions on Networking. – June 1995. – Vol. 3, No.3. – P. 226-244.*
8. Feng W. *The failure of TCP in High-Performance Computational Grids / W. Feng, P. Tinnakornsriruphap // In Proceedings of International Conference on Parallel Processing (ICPP'00). – 2000. – P. 155.*
9. Feng W. *The Adverse Impact of the TCP Congestion-Control Mechanism in Distributed Systems / W. Feng, P. Tinnakornsriruphap // In Proceedings of International Conference on Parallel Processing (ICPP'00). – 2000. – P. 121.*
10. Veres A. *On the propagation of long-range dependence in the internet / A. Veres, Zs. Kenesi, S. Molnar, G. Vattay // In Proceedings of SIGCOMM2000. – 2000. – P. 37.*
11. H. Fowler and Leland, *Local Area Network Traffic Characteristics, with Implications for Broadband Network Congestion Management / H. Fowler and Leland // IEEE JSAC. – September, 1991. – 9 (7). – P. 1139-1149.*
12. Erramilli A. *Modeling Packet Traffic with Chaotic Maps / A. Erramilli, R.P. Singh, P. Pruthi // Proc. 7th ITC Specialist Seminar, Morristown. – 1993. – NJ. – P. 8.1.1-8.1.3.*
13. Erramilli A. *An Application of Deterministic Chaotic Maps to Characterize Packet Traffic / A. Erramilli, R.P. Singh. – Preprint, 1992.*
14. Nagle. J. *RFC896-Congestion control in IP/TCP internetworks / J. Nagle. – 1984.*
15. Jacobson V. *Congestion Avoidance and Control / V. Jacobson // In Proceedings of the SIGCOMM'88 Symposium. – 1988. – P. 314-332.*
16. Brakmo L. *TCP Vegas: End to End Congestion Avoidance on a Global Internet / L. Brakmo, L. Peterson // IEEE Journal of Selected Areas in Communications. – 1995. – 13. – P. 1465-1480.*
17. Floyd S. *Random Early Detection Gateways for Congestion Avoidance / S. Floyd, Jacobson // IEEE/ACM Transactions on Networking. – August 1993. – 1(4). – P. 397-413.*
18. Карпукhin А.В. *Анализ поведения информационных систем с большой нелинейностью / А.В. Карпукhin, Ю.П. Мачехин // Материалы междунар. науч. конф. ISDMCI'2000, 18-23 мая 2008 г. Евпатория. – Херсон, 2008. – Т. 3. – 192 с.*
19. *Математическое моделирование сильнонелинейных информационных систем / А.В. Карпукhin, Л.О. Кириченко, Ю.П. Мачехин, Т.А. Радивилова // Материалы 1-й междунар. научн. конф. «Электронная компонентная база. Состояние и перспективы развития, 30 сентября-3 октября 2008 г. Судак. – Х.: ХНУРЭ, 2008. – 346 с.*

Поступила в редколлегию 10.06.2009

Рецензент: д-р техн. наук, проф. В.В. Поповский, Харьковский национальный университет радиоэлектроники, Харьков.

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ХАОТИЧНИХ ЯВИЩ У ШВИДКІСНИХ МЕРЕЖЕВИХ ІНФОРМАЦІЙНИХ СИСТЕМАХ З ПРОТОКОЛОМ TCP

О.В. Карпукhin

Розглянуто питання виникнення хаотичних явищ у швидкісних інформаційних системах, працюючих по протоколу TCP. Вказано причини виникнення цих явищ та їх вплив на характеристики інформаційних систем з швидкісними інформаційними потоками. Запропоновано нові підходи до моделювання таких систем.

Ключові слова: хаос, структуризація, синергетика, інформаційні потоки, фрактали, TCP/IP, самоподоба, математичні моделі.

MATHEMATICAL MODELLING OF CHAOTIC PHENOMENA IN HIGH-SPEED NETWORK INFORMATION SYSTEMS WITH TCP PROTOCOL

O.V. Karpukhin

Questions of occurrence of the chaotic phenomena in the high-speed information systems working on TCP protocol are considered. The reason of appearance of these phenomena and their influence on the performance of information systems with high-speed information streams are pointed. New approaches to simulation of such systems are offered.

Keywords: chaos, structurization, synergetic, information streams, fractals, TC/IP, self-similarity, mathematical models.