-

Complex Dynamics in Communication Networks

Data Traffic, Topology and Congestion

D.K. Arrowsmith[1], R.J. Mondragón[2], M. Woolf[1]

[1]School of Mathematical Sciences and
[2] Department of Electronic Engineering

Queen Mary, University of London

London 2004

The aim of this paper is to describe the behaviour of packet traffic flowing on realistic networks as load is varied. The use of intermittency in iterated maps to provide various relevant statistical types of binary data will be described. The dynamical modelling of packet traffic using *Erramilli* intermittency maps is introduced together with the dynamics of Transmission Control Protocols. Regular and scale-free network topologies are used for Internet packet traffic modelling and the congestion behaviour of packet lifetimes on these networks under increasing load is investigated.

1 Introduction

We consider the interaction between the topology of a network and the traffic carried along its channels. The binding elements between the topology and the traffic dynamics are the routing mechanisms. In a packet-based network, like the Internet, the transmission of information is carried out in discrete packets. The path that a packet follows when travelling the network is determined by the routing algorithm. Usually, from the topological properties of the network and statistical properties of the traffic, the routing algorithm tries to minimise the packet delivery time and maximise the throughput; this implies that packet flow affects the behaviour of the routers which in return regulate the flow. The dynamics of a packet network can also be regulated by controlling the packet production at the various packet sources and varying the server capacities. An example is *Transmission Control Protocol* or *TCP*, where the source of traffic adjusts its rate of packet transmission as a function of the round trip delay time.

Previously, the statistical properties of packet traffic, which are dependent on the control and routing algorithms, were described by a model where the traffic input was *Poisson-like* where the auto-correlation decays exponentially fast. Traffic with this decay property, of which the Poisson type is a particular example, is referred to as having *Short Range Dependence* (*SRD*). From studies carried out in the early 1990's [17] it is known that *Poisson-like* models do not capture all the statistical properties of packet traffic. Packet traffic exhibits spurts of activity over a large number of time scales. These bursts last from milliseconds to days and they look similar independently of the time scale. This phenomenon is known as *self-similar* traffic. One characteristic of this self-similar traffic is that it has *Long Range Dependence* (*LRD*), i.e. the traffic is strongly correlated at all time scales of engineering interest. This observation was a surprise as, previously, the properties of packet traffic were described as *SRD* processes. We begin in section 2 by briefly discussing some of the properties of *LRD* packet traffic.

Even though some researchers [26, 40] have suggested that the burstiness in packet traffic is connected to the behaviour of individual users within the network, the modelling of packet traffic is based on its measured characteristics more than on the underlying mechanisms responsible for the self-similarity. The bursty traffic is often described by stochastic methods based on Gaussian self-similar processes, for example fractional Brownian motion and fractional Gaussian noise [35, 31, 46, 25] or, on the superposition of *on/off* sources with *heavy tailed on* or *off* periods [48] and chaotic maps [14, 15, 39]. All these models describe successfully the burstiness of the traffic but their approach is very different. In section 3 we introduce a non-linear chaotic map as a model to generate packet traffic with varying statistical properties. We also discuss several equivalent deterministic models for packet production models which have calculational advantages. The packet production dynamics is also extended to employ the *TCP* window dynamics [13].

The packet delivery time is the time that elapses between the creation of a packet at its source s, to

the arrival at its destination d. This time is known as the *end-to-end* time, packet *lifetime* or *latency*. A packet travels through the network visiting different nodes. If one of the nodes is busy, the packet is stored in the queue at that node. Eventually, as the node serves its queue, the packet is forwarded toward its destination. Usually longer routes and/or congested queues mean longer delivery times. The routing algorithm tries to reduce the packet delivery time by selecting short, lightly utilised routes. In such a network, the traffic characteristics are not drastically changed as the packet transverses the network. The delivery time for a packet from its source to its destination is finite. As the load increases, the delivery time will typically increase accordingly. There is a critical load where the delivery time diverges, or at least increases dramatically. At this point the network is congested.

In a regular-symmetric network, it is possible to predict the traffic load where congestion occurs (a dynamical characteristic) by only considering the average of the shortest path lengths from all sources to all destinations (a topological characteristic) [21, 19, 41, 51]. In section 4 the relevant topological characteristics when studying congestion are introduced and the network dynamics is introduced in section 4.4.

In section 5 we make the connection between the traffic characteristics, the topology of the network and congestion by looking first at *Poisson* traffic and then at the differences when the input traffic is *LRD*. In section 6 we briefly introduce some mechanisms to control *LRD* traffic, by limiting the size of the queues [5], by reducing the rate of packet production [45] and by using *TCP*-like control [52].

2 Long-Range Dependence

In 1994, *LRD* was shown to be a feature of Internet packet traffic by Leland *et al.*, [29]. The *LRD* behaviour manifests itself along a communication channel as *bursty* activity in the packet rate (no. of packets/unit time) which persists on all relevant time scales. The bursty traffic makes it much more difficult to implement effective traffic congestion protocols (e.g. *TCP*).

The statistical nature of *LRD* traffic is formally defined in [8]. A key requirement is that the autocorrelation of packet traces, $\gamma(k)$, where the lag is k, satisfies a power law decay of the form $\gamma(k) \sim Ck^{-\beta}$, where $\beta \in (0,1)$ and C is constant. Equivalently, $\gamma(k) \sim Ck^{-2+2H}$, where $H = 1 - \beta/2 \in (\frac{1}{2}, 1)$ is the *Hurst* parameter. By comparison, *Poisson* traffic has an exponential rate of decay $\gamma(k) \sim C'\alpha^{-k}$ with $\alpha > 1$ and C' a constant. The Hurst parameter distinguishes between *LRD* traffic for $H \approx 1$ and the *onset* of *SRD* traffic for $H \approx 1/2$, when the autocorrelation decay changes to exponential.

The essential contrast between *SRD*, Poisson-like traffic, arising typically from traditional voice traffic, and the bursty nature of Internet *LRD* traffic is seen in Fig. 1. The effect of scaling is shown for (a) long-range dependent and (b) short-range dependent traffic for a time series of a random variable $X_n, n = 0, 1, 2, \ldots$ The data is averaged in batch sizes of $N = 1$, 10, 100 and $N = 1000$.

The standard deviation in the *Poisson* traffic varies as the square root of the batch size, or magnification, and we see a 'smoothing' of the traffic as N increases in Fig. 1(b). Thus the mean is an increasingly effective indicator of the instantaneous load, i.e expected packet rate, in the traffic. By comparison, for *LRD* traffic, we see that the variation around the mean remains relatively high for large N in Fig. 1(a). Even when averaged over longer time intervals by several orders of magnitude, we can still obtain packet rates which are close to 0 and 1.

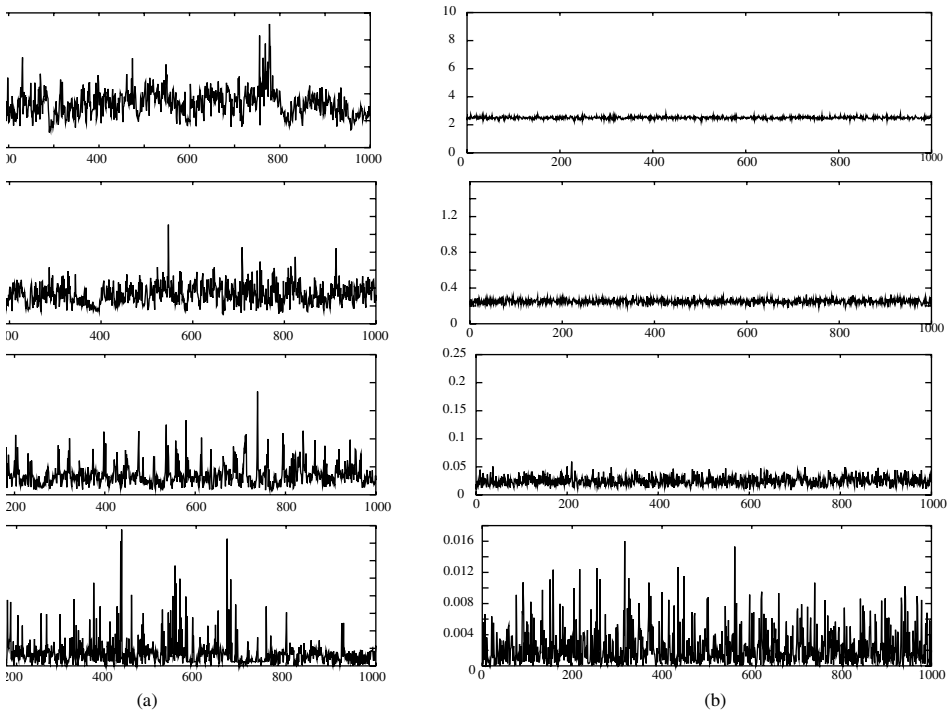

Figure 1: The batch averages of packets/unit time for (a) a real *LRD* traffic trace (Bellcore data from http://ita.ee.lbl.gov/html/contrib/BC.html), and (b) a Poisson based trace for the same load. Each for sizes $N = 1, 10, 100$ and $N = 1000$. A relatively large variance is retained in case (a).

One of the consequences of *LRD* traffic is that it increases queue lengths and latency dramatically. The length of a queue fed with *LRD* traffic sources decays as a power law, compared with an exponential decay if it is fed with *Poisson* traffic sources. The effects of *LRD* cannot be 'removed' by a control mechanism and *LRD* needs to be allowed for, both in computer models of network behaviour and in the routing algorithms used to control data flow through networks.

3 Packet production models

Previous simulations of *SRD* packet traffic generation at each host have used *SRD* Poisson (or Markovian) distributions. In this case a packet is created at a host only if a random number on the unit interval, $\mathbb{I} = \{x | x \in [0, 1]\}$, is below a discriminator value λ. Hence, for a uniform random distribution the average rate at which packets are produced at a host is λ.

An alternative to this is to use chaotic maps to model the *LRD* nature of real packet traffic. We used the family of maps $f = f_{(m_1.m_2.d)} : \mathbb{I} \to \mathbb{I}$ defined in the unit interval $\mathbb{I}$ by $x_{n+1} = f(x_n)$ where

$$x_{n+1} = \begin{cases} x_n + (1-d)\left(\dfrac{x_n}{d}\right)^{m_1}, & x_n \in [0,\, d], \\ \\ x_n + d\left(\dfrac{1-x_n}{1-d}\right)^{m_2}, & x_n \in (d,\, 1], \end{cases} \tag{1}$$

described in previous papers (see Erramilli *et al*, [14]), and related maps in [47]. Here $d \in (0, 1)$ and the

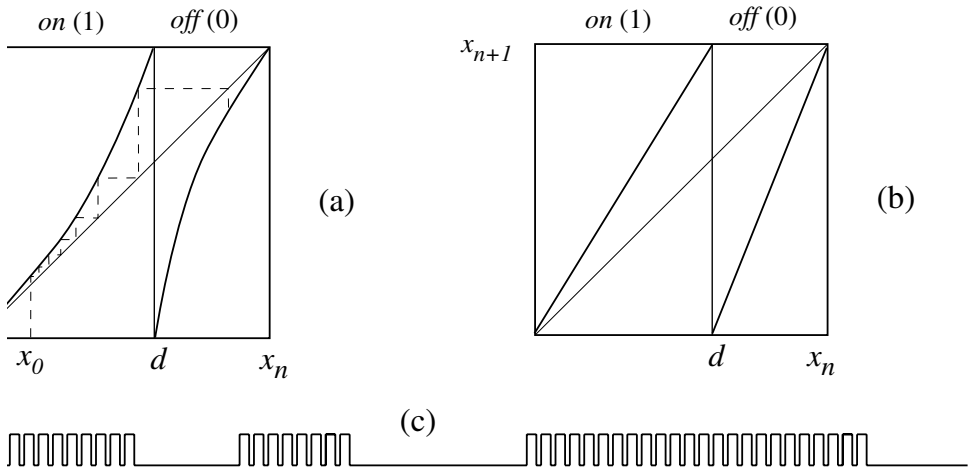

Figure 2: (a) The graph of the map f (Eq. 1) consists of two segments and each has a tangency with the line $y = x$. The iteration of the map f with initial condition x_0 forms a 'web' generating the iterative sequence, or orbit, x_n, where $x_n = f(x_{n-1})$, $n = 1, 2, \ldots$. Note that the tangencies at $x = 0$, 1 give a 'slow' change in the values of the sequence x_n, and therefore the output y_n provides long sequences of consecutive '0's or '1's. (b) Graph of f for $m_1 = m_2 = 1$, this is known as the Bernoulli shift map and it generates *Poisson* traffic. If the map parameters are chosen to satisfy $m_1 = m_2 \in \{1, \frac{3}{2}\}$, then the map generates *SRD* traffic.
(c) The traffic on a packet network is represented by a binary sequence of zeros and ones (Eq. 2).

parameters m_1, $m_2 \in (\frac{3}{2}, 2)$ induce *intermittency* at each of the points $x = 0$ and $x = 1$, by producing tangencies to the diagonal in the graph of f. The orbital 'escape time' in neighbourhoods of 0 and 1 become power law dependent. If this map is iterated a large number of times, the values of x_n will form a non-uniform continuous distribution on the interval $\mathbb{I}$. The parameter d is used as a discriminator, as λ is for the Poisson case. If x_n falls between 0 and d, a packet is generated; and if x_n falls between d and 1, no packet is generated. Thus we have a discrete output map associated with the function in Eq. (1) which is

$$y_n = \begin{cases} 1: & x_n \in [0,\, d] \;-\; \text{packet generated}\,, \\ 0: & x_n \in (d,\, 1] \;-\; \text{no packet generated}\,. \end{cases} \tag{2}$$

The above model, which represents the traffic as a binary sequence is also known as a *packet train* model [27] (see Fig. 2(c)). The intermittency behaviour of the map f induces so-called 'memory' in the digital output y_n giving the long range correlation effects required for the packet traffic. This feature is shown by the slow decay of variance with respect to n, the size of batched output, see Fig. 1(a) and [29]. The power-decay of the variance arises from the small orbital increments of the intermittency map which in turn provides memory in the digital output. An example of this phenomenon is illustrated in Fig. 2(a) where a sequence of the iterated values x near the origin have small increments. This effect is stronger for orbits passing even closer to $x = 0$. The time of escape (i.e. into the region $x > d$) of an orbit from a neighbourhood of the origin has a power-law dependence on its initial position, [47].
The nonlinear nature of f means that in this case the load $\lambda = \lambda(m_1, m_2, d)$ (i.e. the average value of the output y per iteration) is not equal to d, but is given by $\lambda = \int_0^d \nu(x)\, dx$, where ν is the natural invariant density distribution of the map f on the interval $[0, d]$. The distribution ν has no closed form and is often obtained numerically via the Perron-Frobenius operator [32]. Thus the various statistical

properties of traffic generated in this way are determined by the map's parameters m_1, m_2 and d including the auto-correlation behaviour as we shall see in section 3.3.

3.1 Closed form map

There are two other important models of the above intermittency type which have useful mathematical characteristics not available in the standard model described above. The first extension was introduced by Pruthi [38], and the closed form model appears, at first sight, to be more intractable than the original model in Eq. (1). Essentially, the function in (1) is replaced by

$$x_{n+1} = \begin{cases} \dfrac{x_n}{(1 - c_1 x_n^{m_1-1})^{\frac{1}{m_1-1}}}, & x_n \in [0,\, d], \quad (a) \\ 1 - \dfrac{1 - x_n}{(1 - c_2(1-x_n)^{m_2-1})^{\frac{1}{m_2-1}}}, & x_n \in (d,\, 1], \quad (b) \end{cases} \tag{3}$$

where

$$c_1 = \frac{1 - d^{m_1-1}}{d^{m_1-1}}, \qquad c_2 = \frac{1 - (1-d)^{m_2-1}}{(1-d)^{m_2-1}}. \tag{4}$$

A Taylor expansion of the functions in Eq. 3 (a) and (b), around the points $x = 0$ and $x = 1$ respectively, give the forms of the equation (1) and so the leading intermittency effects are the same in both models with leading exponents m_1 and m_2. A distinct advantage of (3) is that, somewhat remarkably, it has a closed form under composition. If the first branch function is denoted by $f_1(c_1,\, m_1,\, x)$ then the n-th iterate can be shown to satisfy $f_1^n(c_1,\, m_1,\, x) = f_1(nc_1,\, m_1,\, x)$. Similarly the second branch function f_2 satisfies $f_2^n(c_2,\, m_2,\, x) = f_2(nc_2,\, m_2,\, x)$. This enables sojourn times in the "*off*" or "*on*" regions to be calculated explicitly, by solving equations of the type $f_1^n(c_1,\, m_1,\, x) = d$, to obtain the number of iterations to the transition point.

We can use the closed iterative form above to calculate the probability of 'escape' from an intermittency region. Specifically, we consider the probability of a sequence of k-consecutive zeroes for the output y_n of an intermittency map f. We will use the closed form map in Eq. (3) and will assume a random uniform injection into the region $[0, d]$ for $x > d$, see [37].

If the orbit re-enters the interval $[0, d]$ at the point $\bar{x}$, then that determines the sequence length l of zeroes, namely,

$$l(\bar{x}) = \frac{1}{c_1}\left(\frac{1}{\bar{x}^{m-1}} - \frac{1}{d^{m-1}}\right) \tag{5}$$

Let $P(l(x))$ be the probability density for at least length l "zero" sequences. If we are assuming that the initial point density on the interval $[0, d]$ at $\{\bar{x}\}$ is $\hat{P}(\bar{x})$, then $\hat{P}(\bar{x})d\bar{x} = P(l(\bar{x}))dl$.

If we further assume the re-entry density $\hat{P}(\bar{x})$ to be uniformly random, i.e. $\hat{P}(\bar{x}) = \hat{P}$ a constant, then

$$P(l) = \hat{P}\left|\frac{d\bar{x}}{dl}\right| \sim C l^{-\frac{m}{m-1}} \tag{6}$$

for small x with C a constant. This is not the case for re-entry to the interval $[0, d]$ when the full double intermittency is used.

3.2 Piece-wise linear map

A second model which has different mathematical advantages over the original Erramilli model is a piece-wise version. This was originally introduced by Wang [47] and has been used subsequently in applications to realise maps with given autocorrelation profiles [7]. The analogue is constructed with the use of two sequences. The piecewise linear map $p : \mathbb{I} \to \mathbb{I}$ is defined by two monotonic sequences z_i^L and z_i^R monotonic decreasing to zero, with $z_1^L = z_1^R = 0.5$. The map p is then defined by requiring that its graph be piecewise linear with nodal points defined by $(z_i^L,\, z_{i+1}^L)$, and $(1 - z_{i+1}^R,\, 1 - z_i^R)$, $i = 1, 2, \ldots$. The piecewise linear map can replicate the intermittency behaviour at $x = 0, 1$ by choosing the sequences to decay to zero in an appropriate way. If we let $z_i^L \sim i^{-\alpha}$ and $z_i^R \sim i^{-\beta}$ with $\alpha, \beta > 1$, then the exponents for the different types of smooth and piecewise linear map have analogous asymptotic behaviour at the intermittency points. We identify the parameters as $\alpha = 1 + 1/m_1$, and $\beta = 1 + 1/m_2$. The piece-wise linear map has distinct advantages in that it is possible to calculate the invariant measure associated with the map, see [7].

3.3 Autocorrelation of the map output

We have already seen in Fig. 1 that the movement between strings of the output values '0' and '1' is rapid in trace (b) and much slower in trace (a). The intermittency in traffic maps produces increased sojourn times for the two states. The longer sojourn times are said to introduce *memory* into the output which is reflected in a higher correlation between the output binary sequence and the same sequence with a time-lag k. The *autocorrelation* vector of a sequence is the way in which the memory is measured. Let X_t be a scalar time series of the binary values $\{0, 1\}$ for $t = 0, 1, 2, \ldots$, and suppose the series is stationary. We define the autocorrelation of time lag k by

$$\gamma(k) = \frac{E(X_t X_{t+k}) - E(X_t)E(X_{t+k})}{\sqrt{(\mathrm{Var}(X_t)\mathrm{Var}(X_{t+k}))}}. \tag{7}$$

Let $\mu = E(X_t)$. Note that the values X_t are binary, then $E(X_t^2) = E(X_t) = \mu$ and so $\mathrm{Var}(X_t) = E(X_t^2) - E(X_t)^2 = \mu(1 - \mu)$. Therefore, the autocorrelation can be re-written

$$\gamma(k) = \frac{E(X_t X_{t+k}) - \mu^2}{\mu(1 - \mu)} \tag{8}$$

Given $0 \leq X_t X_{t+k} \leq X_t$, it follows that $\gamma(k) \leq 1$. Note also that if there is no correlation, i.e. the values X_t are independent of each other, then $E(X_t X_{t+k}) = E(X_t)E(X_{t+k})$ and $\gamma(k) = 0$. Thus, in general, we expect that the correlation coefficient $\gamma(k)$ will eventually decay to zero in some way. Two special types of decay are

(a) power-law decay, where $\gamma(k) \sim Ck^{-\beta}$, for some constant C and $\beta > 0$;

(b) exponential decay, where $\gamma(k) \sim C\alpha^{-k}$, for some constants C and $\alpha > 0$.

3.3.1 Bernoulli map decay

The piece-wise linear Bernoulli map is given by $f(x) = 2x \bmod 1$ on the interval [0, 1]. Given the current state is '0', then $0 \leq x < 0.5$. The probability of the transfer '$0 \to 0$' is 0.5, since it requires $0 \leq x < 0.25$, and similarly for '$1 \to 1$'. Thus we can calculate the autocorrelation exactly and we obtain

$$\gamma(k) = \begin{cases} 1 & \text{for } k = 0 \\ 0 & \text{for } k > 0. \end{cases} \tag{9}$$

3.3.2 Intermittency map decay

Only two exact results are known so far for the asymptotic properties of double intermittency maps considered here. Let the piecewise linear map constructed from two sequences $z_i = i^{-\alpha}$ at $x = 0$ and $w_i = 1 - i^{-\beta}$ at $x = 1$, α, $\beta > 1$, then the two intermittencies compete and it can be shown that

$$\gamma(k) \sim K\, k^{-c} \tag{10}$$

where $c = \min\{\alpha, \beta\} - 1$, K constant, [7]. Thus the correlation for the composite map is determined by the heaviest tail in the correlation decay arising from the two competing intermittencies.
A similar result is also available for the differentiable case [33]. In this case, the map is as in Eq. (1) with the extra condition that whenever f iterates across the line $x = d$, the formula is replaced by a random uniform injection. For example, the autocorrelation vector $c(n), n \in \mathbb{Z}^+$, of the output function y is known to have asymptotic behaviour $\gamma(k) \sim Ck^{-\beta}$, with C constant, as $k \to \infty$. The constant $\beta = (2 - m)/(m - 1) \in (0, 1)$, for $m = \max\{m_1, m_2\}$, with m_1, $m_2 \in (\frac{3}{2}, 2)$, (see [22, 33, 7]). Furthermore, the Hurst parameter, H, is given by

$$H = 1 - \frac{\beta}{2} = \frac{3m - 4}{2(m - 1)}, \tag{11}$$

and ranges over the interval $(\frac{1}{2}, 1)$, as required. Thus m_1, $m_2 = 1.5$ corresponds to *Poisson-like* behaviour and as m_1, m_2 are increased towards 2, the behaviour is increasingly long-range dependent, see [7, 33]. If the transition simplification is removed then the result that the heaviest tail dominates in the piece-wise smooth case is still open but the auto-correlation decay is conjectured to remain as in Eq. (11).

3.4 Transmission Control Protocol Dynamics

The dynamics of the packet production model can be extended to incorporate packet window dynamics [13]. If the map is in the '*on*' state, each iteration of the map represents a packet generated. One sojourn period in the '*on*' side of the map represents a whole file. These *files* are then *windowed* using the *slow-start* algorithm, adding another dynamical layer to the system. The *slow-start* algorithm is as follows:
At a given host i in the network, and time $t = n$, there is a current state, $x_i(n)$, and a current window size, $w_i(n)$, for the number of packets that can be sent at time $t = n$. There is also a residual file size, $s_i(n)$, at node i which is given by the number of iterates of f such that $f^{s_i(n)}(x_i(n)) < d$, and $f^{s_i(n)+1}(x_i(n)) > d$.

The source will send $p_i(n) = \min\{w_i(n), s_i(n)\}$ packets. The full dynamics therefore takes the form, (see Erramilli *et al.*, [13]):
For $x(n) < d$, i.e. packet generated-

$$w_i(n+1) = \begin{cases} 1, & \text{if } x_i(n-1) < d, \\ \min\{2w_i(n),\, w_{max}\}, & \text{otherwise,} \end{cases} \tag{12}$$

and $x_i(n+1) = f^{p_i(n)}(x_i(n))$.
For $x_i(n) > d$, i.e. no packet generated - $w_i(n+1) = 0$, and $x_i(n+1) = f(x_i(n))$.
This algorithm applies if all packets in a window are acknowledged before the retransmission timeout (RTO) limit is reached. If packets take longer than this to be acknowledged the window of packets is sent again with the RTO doubled and the window size set to zero. When the map is in the '*off* state, the window size is zero and no packets are sent.
This initial value of RTO is calculated using the exponential averaging method [42]. This method keeps a running average of all round trip times. This average is weighted towards more recent round trips, and is used in calculating the *initial* RTO.

4 Topology and Models of Networks

Many different topologies appear in communication networks. Square lattices, toroidal lattices, meshes and hypercubes arise on multiprocessor computers (e.g. [30]), scale-free networks in the WWW [2] and the Internet [16]. The way that the elements of the network are connected to each other and the nodal degree properties have an impact on its functionality. The representation and study of the connectivity of a network is carried out using concepts from graph theory.
A communications network can be represented by a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes (vertices) and $\mathcal{E}$ is the set of links (edges). The hosts, routers and switches are represented by nodes and the physical connections between them are represented by links. The links can have a direction, but here we are only going to consider undirected links. A node can transfer information to another node in the form of data-packets if there is a link between them. If there is no direct link between the nodes, then a path in the network is the sequence of distinct nodes visited when transferring data-packets from one node to another. We consider networks where there exists at least one path connecting any pair of nodes of the network.
The degree, k, of a node is the number of links which have the node as an end-point [49], or equivalently, the number of nearest neighbours of a node, see Fig. 3(a). The degree of a node is a local quantity. However, the node degree distribution of the entire network gives important information about the global properties of a network and can be used to characterise different network topologies.
If there is very little traffic on the network and if the journey time from one node to its neighbour is in unit time, then given any two nodes and to first approximation, the journey time will be proportional to the length of the journey, or the path length. A path that goes from source node s, to destination node d, in the smallest number of hops is called the shortest-path. The length of the shortest-path $\ell_{s,d}$ is the number of nodes visited when going from s to d. There can be more than one shortest–path between a

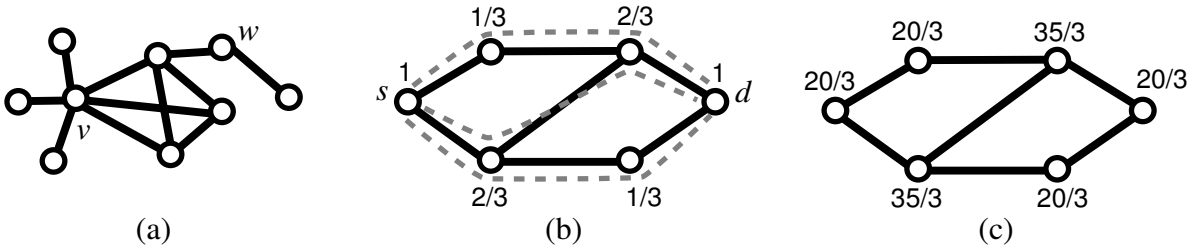

Figure 3: (a) The node degree for node v is 6 and for w, it is 2. (b) Three different shortest paths between s and d. The length of the path is 3. The number above the nodes denotes the proportion of shortest paths $p_{s,d}(w)$ that go through that node. (c) The node medial centrality for the whole network.

pair of nodes. The characteristic path length

$$\bar{\ell} = \frac{1}{S(S-1)} \sum_{s \in \mathcal{V}} \sum_{d \neq s \in \mathcal{V}} \ell_{s,d}, \tag{13}$$

where S is the total number of nodes, is the average shortest-path over all pairs of nodes (see Fig. 3(b)). Sometimes $\bar{\ell}$ is referred to as the *diameter* of the network.

If there is traffic on the network the difference in the journey times of two shortest paths with the same length can be very different. Not all the journeys are equal due to the different patterns of usage of the routes. On a network, there are nodes that are more prominent because they are highly used when transferring packet-data. A way to measure this "importance" is by using the concept of *betweeness centrality* of a node. We will refer to this concept here as *medial centrality* The concept of centrality [18] was introduced in social networks to characterise the prominence of an individual in the *context of the social structure*. Given a source s, and destination d, the number of different shortest-paths is $g(s,d)$. The number of shortest-paths that contain the node w is $g(w;s,d)$. The proportion of shortest-paths, from s to d, which contain node w is

$$p_{s,d}(w) = \frac{g(w;s,d)}{g(s,d)}. \tag{14}$$

Remark: The proportion of shortest-paths and the shortest-path length are related by

$$\ell_{s,d} = \sum_{w \in \mathcal{W}} p_{s,d}(w) - 1, \tag{15}$$

where $\mathcal{W}$ is the set that contains the nodes visited by the shortest paths from s to d.

The *medial centrality* of node w is defined as [24]

$$\mathcal{C}_B(w) = \sum_{s \in \mathcal{V}} \sum_{d \neq s \in \mathcal{V}} p_{s,d}(w) \tag{16}$$

where the sum is over all possible pairs of nodes with $s \neq d$. The medial centrality measures how many shortest paths pass a certain node (see Fig. 3(c)). A node with a large $\mathcal{C}_B$ is "important" because a large amount of packets flow through it, that is, it carries a large traffic load. If this node fails or gets congested, the consequences to the network traffic can be very drastic [24, 55].

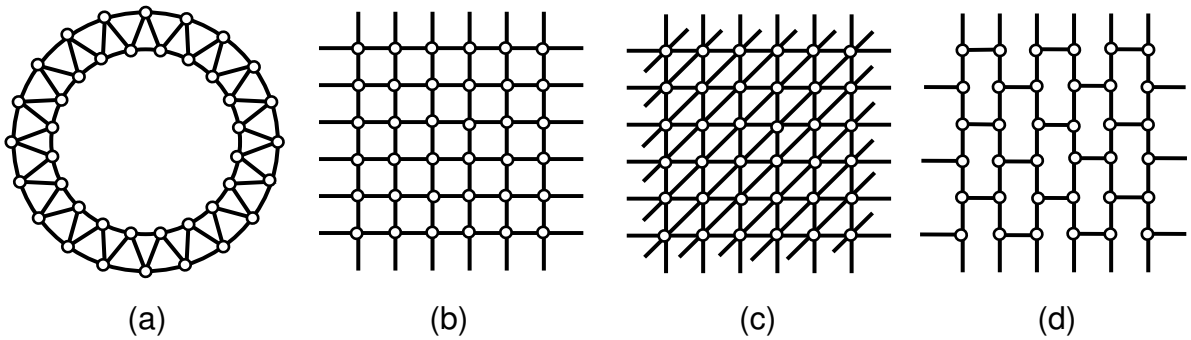

Figure 4: Four regular-symmetric networks. (a) The ring network and (b) the rectangular toroidal network, in which the nodes on one edge of the lattice connect to nodes on the opposite edge, have degree four. (c) The triangular toroidal network has degree six and (d) the hexagonal toroidal network has degree three.

4.1 Regular-Symmetric Networks

In a regular network all nodes have the same degree (see Fig. 4). By symmetry, the medial centrality is constant for all nodes. From Eq. (15) and if $\mathcal{C}_B(w) = c$ then

$$c = \frac{1}{S}\sum_{w\in\mathcal{V}} \mathcal{C}_B(w) = \frac{1}{S}\sum_{s,d} \ell_{s,d} - 1. \tag{17}$$

4.1.1 Toroidal networks

The toroidal rectangular network ($\mathcal{H}$) is based on a square lattice of nodes in which each node has four neighbours with boundary nodes appropriately identified, [41, 19, 36, 51]. The finite rectangular lattice $\mathbb{Z}$ consists of $S = L^2$ nodes. The position of each node in the lattice is given by the coordinate vector $\mathbf{r} = (i, j)$ where i and j are integers in the range 1 to L. The network has periodic boundary conditions throughout, and so each coordinate of (i, j) is effectively reduced - mod $L + 1$ to give a toroidal topology. To measure the distance between a pair of nodes the periodic "Manhattan" metric is used, which measures the sum of vertical and horizontal displacements between two nodes.

The average shortest path can be calculated to be $\bar{\ell} = L/2$.

Useful comparative networks for traffic studies on Manhattan grid are the hexagonal and triangular networks (see Figure 4 (c & d)). Their embeddings in the plane show how we can view the three regular networks as satisfying an edge inclusion property $E(\mathcal{H}) \subset E(\mathcal{R}) \subset E(\mathcal{T})$. Comparative studies of all three types of network show consistent results [4] on the critical loads, and the onset of congestion behaviour.

4.2 Random Networks

Random networks have been used to model communications networks. The reason is that because some of the communication networks tend to have a complex topology and the interactions defining their structure are apparently random.

From a set of nodes, a random network is built by connecting every pair of nodes with probability p. If the total number of nodes is S and if $p > 1/S$ [10] then, with probability 1, the network is fully connected

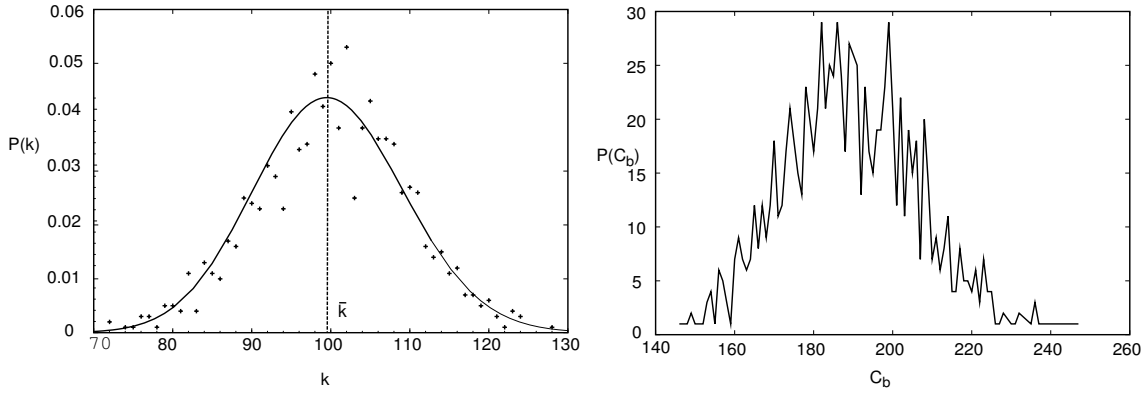

Figure 5: (a) Node degree distribution for a random network with $S = 1000$ and $p = 0.1$, and its approximation using the binomial distribution. (b) Medial centrality distribution of the network.

, or equivalently, there is at least one path connecting any pair of nodes. This is the only case we are going to consider here, as we are interested in connected networks. The degree distribution of a random graph is well approximated by a binomial distribution [10]

$$P(k) = \binom{S-1}{k} p^k (1-p)^{S-1-k}. \tag{18}$$

The degree distribution tends to be concentrated around some "typical" node degree, or average node degree $\bar{k}$, see Fig. 5(a) and [12, 9]. The characteristic path length scales with the size of the network as

$$\bar{\ell}_{\text{rand}} \approx \frac{\ln(S)}{\ln(\bar{k})} \tag{19}$$

.

4.3 Scale-free Networks

Many technological networks are not described by a random or a regular network; instead they are better described by a network where the degree distribution is described by a power law [2, 16] where

$$P(k) \sim Ck^{-\beta}, \tag{20}$$

for $\beta > 1$ and C constant. The probability that a node has k edges connected to it is given by $P(k)$. In practical terms a power law distribution means that the majority of the nodes will have very few neighbours, but there is a very small set of the nodes with a very large number of neighbours (see Fig. 6(a)). Networks with this property are known as *scale-free* because power-laws are free of a characteristic scale, that is, there is no characteristic node degree (see Fig. 6(b)).

The diameter of a scale-free network scales as $\bar{\ell} \sim C \ln(\ln S)$ [11] where S is the size of the network. This is due to the existence of "far–reaching" links which are shortcuts when going from a source to a destination. Any node that contains one of these far-reaching links is going to be highly used when transferring packet-data, that is, its medial centrality is large.

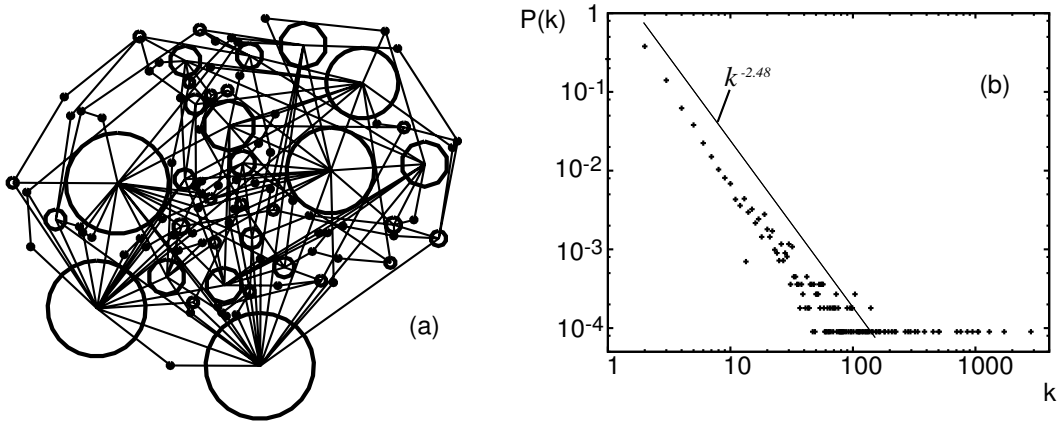

Figure 6: (a) A scale-free network where the size of the node is proportional to its *medial centrality*. In this case is clear that there is a correlation between the node degree and its medial centrality [23]. (b) Degree distribution $P(k)$ vs. node-degree k.

In 1999, Barabási and Albert [6] showed that it is possible to create a scale-free network by using two generic mechanisms: *growth*, the network grows by attaching a new node with m links to m different nodes present in the network; and *preferential attachment*, where new nodes are attached preferentially to nodes that are already well connected. Barabasi and Albert showed further that if the probability that a new node will be connected to node i with degree k_i is

$$\Pi(i) = \frac{k_i}{\sum_j k_j}. \tag{21}$$

then the network has a power law link distribution $P(k) \propto k^{-3}$.

4.3.1 The Internet

In 1999, an analysis of the Internet topology by Faloutsos *et. al* [16] suggested that the distribution of node degrees of the Internet decays as a power-law $P(k) \propto k^{-y}$, with $y = 2.22$. In 2002, Subramanian *et. al* [44] reported that the Internet features a tier structure, where at the top of this tier is the core of the network. It is well established that a realistic model of the Internet topology should generate a power law topology with a core structure. There exist network models that produce power law networks, e.g. Barabási and Albert model [6] and Inet-3.0 [50] to mention just two of them, but they do not reproduce the core structure of the Internet [54].

A node of a network is considered "rich" if it contains a large number of links or equivalently it has a large node-degree. The core of the Internet consist of a set of nodes which have a large node degree. We refer to these nodes as the *rich-club*. In the Internet, the members of the rich-club are very well connected to each other. This means that there are a large number of alternative routing paths between the club members where the average path length inside the club is very small (1 to 2 hops). The rich-club acts as a super traffic hub and provides a large selection of shortcuts. Hence scale-free models without the rich-club structure may underestimate the efficiency and flexibility of the traffic routing in the Internet. Conversely, networks without the rich-club may over-estimate the robustness of the network to a node

attack, [1, 24] where the removal of a small percentage of its richest club members can break down the network integrity.
It is possible to build network topology models which will generate a rich club. Recently the Interactive-Growth (IG) model [53] was introduced as a way to generate networks that contain a rich-club. The model is a modification of the Barabási and Albert model and it reflects the evolution of the Internet, with the addition of new nodes and the addition of new links between existing nodes. The network is generated by starting with m_0 nodes connected through $m_0 - 1$ links. At each time–step, one of the following two operations is performed: 1) with probability $p \in (0, 1)$, $m < m_0$ new links are added between m pairs of nodes chosen from the existing nodes, and, 2) with probability $1 - p$, one new node is added and connected to m existing nodes.

4.4 Model of Networked Data Traffic

The model considered here has been studied by several authors [36, 19, 21, 20, 41, 45, 51, 3, 5]. The network consists of two types of nodes; router nodes that store and forward packets; host nodes that store and forward packets and are also sources and sinks of traffic. Given the network has S nodes, and a density $\rho \in [0, 1]$ of hosts then ρS is the total number of hosts. The host nodes are randomly distributed in the network.

- Traffic generation: A host creates a packet whose destination is another host. A host creates a packet using either a uniform random distribution (Poisson) or a LRD distribution defined by a chaotic map. Each source generates its traffic independently of the other sources.
- Queue: Each node keeps a queue of unlimited length where the packets are stored. Any packet that is generated is put at the end of the host's queue. If a packet arrives at a router it is put at the end of the hierarchical router's queue. The packet is not queued when it arrives at its destination node.
- Routing: Each node picks a packet at the head of the queue and forwards the packet to the next node. From the source/destination information that each packet carries, the forwarding is done by one of the following routing algorithms:
  1. For regular networks:
     - a neighbour closest to the destination node is selected
     - if more than one neighbour is at the minimum distance from the destination, the link to which the smallest number of packets has been forwarded is selected
     - if more than one of these link shares the same minimum number of packets forwarded then a random selection is used
  2. for general networks:
     - From the state of the queues and the number of hops from source to destination, the routing is done by minimising the time–delay by considering the number of hops and the size of the queues that a packet visits when crossing the network.

The process of packet generation, hop movement, queue movement and updating of the routing table occurs at one time step.

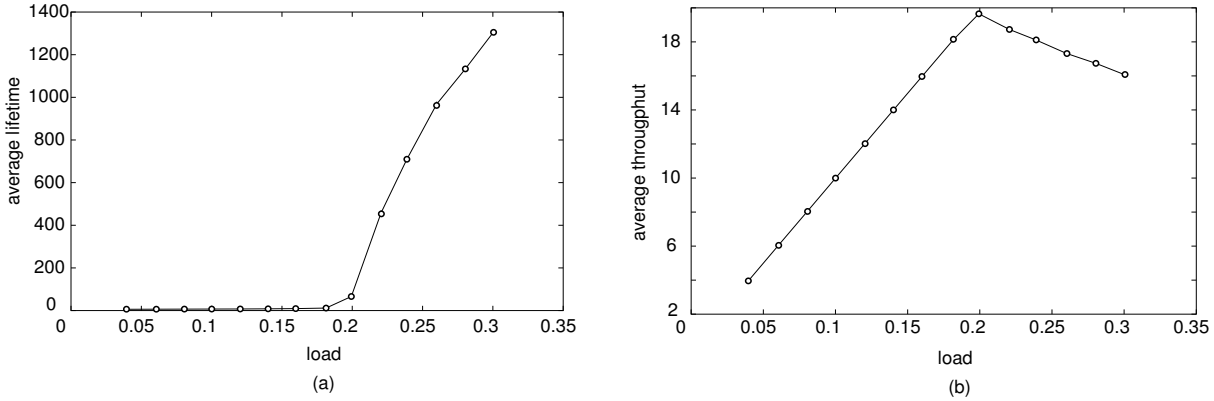

Figure 7: (a) The average lifetime of a packet increases rapidly after the congestion point. (b) Average throughput of a Manhattan network with $S = 100$ and $\rho = 1$. As the offered load increases the average throughput increase until it reaches the critical point $\lambda_c \approx 0.2$. At this load the network is congested.

5 Congestion

The time that elapses between the creation of a packet at source s, from its creation to its destination d, is known as the delay time $\tau_{s,d}$. The average of the delay

$$\bar{\tau}_T = \frac{1}{S(S-1)} \sum_{s,d;\tau_{s,d} \leq T} \tau_{s,d},$$

which is the average for all the packets up to time T, is an important quantity with which to assess the performance of a network. If the traffic load presented to the network is low and the queues on the nodes are empty, then, to a first approximation, the average delay is proportional to the average number of nodes that the packets visit when travelling. As the traffic load increases, the queues at the nodes start to build up, the average delay time will increase accordingly. If the traffic load increases even further, then at the critical load λ_c, the queues of some nodes will grow very quickly and the average delay time will dramatically increase or even diverge. At this critical load, we consider that the network is congested. This critical behaviour is also noticed in the network throughput. The throughput is defined as the number of packets reaching their destination per unit time per host. Starting from a low load, the throughput increases proportionally as the increase of the load, until congestion is reached. At this point the network has its maximum throughput (see Fig. 7).

If the number of packets at node i at time t is denoted by $n_i(t)$, then the total number of packets in the network is

$$N(t) = \sum_{i=1}^{S} n_i(t) = \sum_{i=1}^{S} Q_i(t). \tag{22}$$

If the network is not congested, and the average delay time is finite, then the average number of packets on the network, $\bar{N} = (1/T) \lim_{T \to \infty} N(t)$, is also finite. At the congestion point, the queues of the congested nodes increases rapidly and this implies that the total number of packets on the network continues to increase.

5.1 Mean Field Approximation

5.1.1 Total distance formulation

A simple way of estimating λ_c, [51], is to look for the total distance that all the packets at time t have to travel to reach their destination. In the congested phase if there are queues at all nodes, then the change in total distance is $D(N_{t+1}) - D(N_t)$, where $N(t)$ is the number of packets in the queues at time t, and $D(N_t)$ is the aggregated distance of all packets from their destination at time t. The increase in the number of packets per unit of time is $\rho\lambda S$. If $\bar{\ell}$ is the average path length, then the overall added distance is $\rho\lambda S\bar{\ell}$. By contrast, the aggregated distance is reduced by S given that every packet at the head of the queue moves one step closer to its destination. Thus the change in total distance to destination between time t and $t+1$ is

$$D(N_{t+1}) - D(N_t) = \rho\lambda\bar{\ell}S - S. \tag{23}$$

The critical load λ_c occurs when the total distance no longer decreases, $D(N_{t+1}) - D(N_t) = 0$, which implies

$$\lambda_c = \frac{1}{\rho\bar{\ell}}. \tag{24}$$

This relation refines the results of [41] and [19].

5.1.2 Time-delayed formulation

Another possible way to determine the critical load is to use Little's law [20, 41]: *"The average number of customers in a queueing system is equal to the average arrival rate of customers to that system, times the average time spent in the system"* [28]. Little's law is a flow conservation law which can be restated as: in a steady state, the number of delivered packets is equal to the number of generated packets, or

$$\frac{d\,N(t)}{d\,t} = \rho\lambda S - \frac{N(t)}{\tau(t)}. \tag{25}$$

where $\rho\lambda S$ is the average arrival rate to the queues per unit of time, $\tau(t)$ is the average time spent in the system, and $N(t)/\tau(t)$ is the number of packets delivered per unit of time.
Remark: Little's law does not depend on the arrival distribution of packets to the queue or the service time distribution of the queues. Also it does not depend upon the number of queues in the system or upon the queueing discipline within the system. The law holds only when a steady state exists below the critical load, as shown in Fig. 8 where the load rates have been normalised.
If the load is low, the queues at the nodes tend to be empty and the average delay time is the average shortest path $\bar{\ell}$, that is $\tau \approx \bar{\ell}$. For higher loads the transit time can be approximated by the average shortest path plus the average time, $\mathcal{T}$, that a packet spends in the queues

$$\tau \approx \bar{\ell} + \mathcal{T}(N(t), \bar{\ell}). \tag{26}$$

5.1.3 Criticality for Regular-Symmetric Networks

If the traffic is evenly distributed in the network and the network is not congested, then there exists a steady state solution N^* for the number of packets on the network. Each queue, on average, contains

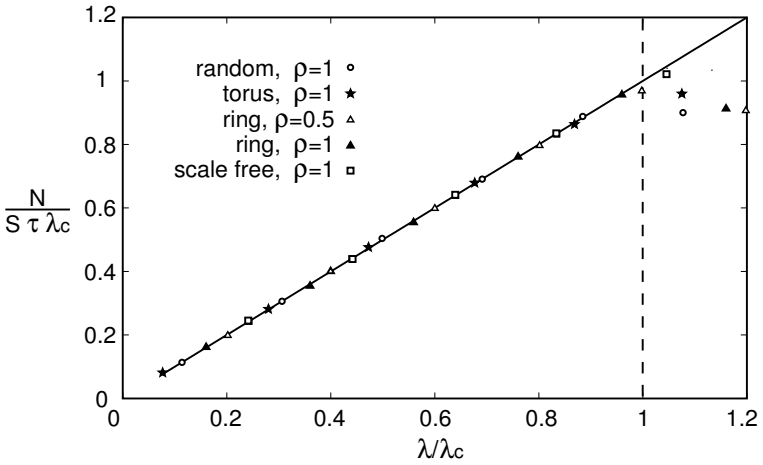

Figure 8: Verification of Little's law for different networks. In all cases the networks have $S = 100$ nodes. The coordinates have been normalised by dividing them by the critical load of the corresponding network. For the ring network, the law was verified for two different densities of nodes. The law does not hold if the load is greater than the critical load (marked with a vertical hash line).

N^*/S packets and the delay can be approximated by

$$\tau(t) \approx \bar{\ell}\left(1 + \bar{Q}\right) = \bar{\ell}\left(1 + \frac{N^*}{S}\right) \tag{27}$$

where on average, a packet visits $\bar{\ell}$ queues with average load $\bar{Q}$ and $\mathcal{T} \approx \bar{\ell}\bar{Q}$. From the steady state solution ($d\,N(t)/d\,t = 0$) the total number of packets in the system is

$$N^* = \frac{\rho\lambda\bar{\ell}S}{1 - \rho\lambda\bar{\ell}} = \frac{\gamma S}{1 - \gamma} \tag{28}$$

where $\gamma = \rho\lambda\bar{\ell}$ is the normalised load and N^* is the steady state solution.
The average traffic load generated at node i is

$$\lambda_i = \left(\frac{1}{\rho\bar{\ell}(1 + S/N^*)}\right). \tag{29}$$

As the traffic load increases, the number of packets in the network increases accordingly. At the congestion point the number of packets on the network diverges, $N^* \to \infty$, and the critical load is

$$\lambda_c = \frac{1}{\rho\bar{\ell}}, \tag{30}$$

which is the same as Eq. (24).

5.1.4 General networks

Fig. 9(a) shows, in phase space, that the transition from the free flow phase to the congested phase is well approximated by Eq. (30) for the case of regular-symmetric networks but not for scale-free networks.

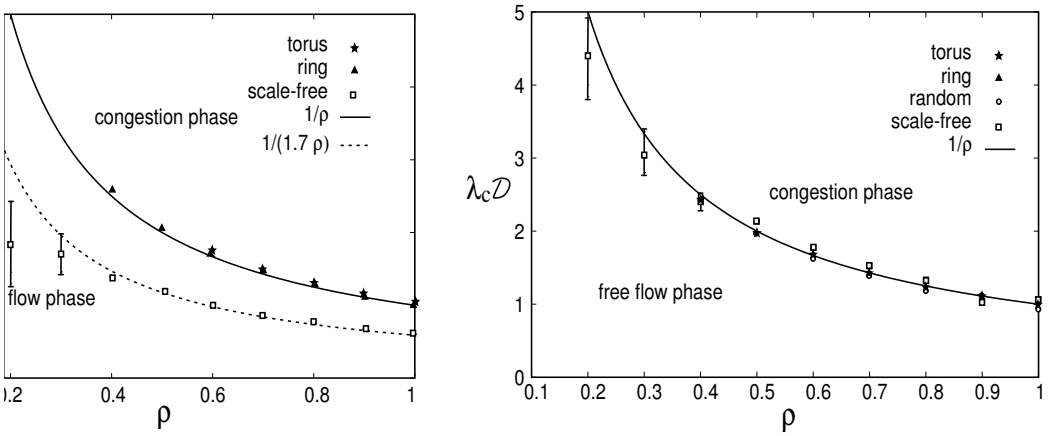

Figure 9: (a) Phase diagram showing the change from free–flow to congestion. For the square toroidal and the regular-symmetric network the phase transition is given by $\ell_c\lambda_c = 1/\rho$. For $\rho > 0.4$, the phase transition of the scale free network is well approximated by $\ell_c\lambda_c = 1/(1.7\rho)$. (b) Renormalised phase transition for different networks topologies. The four networks have $S = 100$.

Equation (30) gives a good approximation to the critical load because the average queue size for all nodes is very similar ($\bar{Q} \approx N^*/S$).

In a scale-free network, due to the disproportionate importance of some of the nodes, if these nodes get congested more readily and hence, all the network gets congested. An alternative approach is to use the medial centrality [34] to characterise the node usage. If

$$\hat{\mathcal{C}}_B(w) = \frac{\mathcal{C}_B(w)}{\sum_{v\in\mathcal{V}} \mathcal{C}_B(v)} \tag{31}$$

is the normalised medial centrality then the average queue size at node i can be approximated by

$$\bar{Q}_i \approx \hat{\mathcal{C}}_B(n_i) N^*. \tag{32}$$

An example of this approximation is shown in Fig. 10.

It is possible to approximate the typical travel time of a packet by adding the average time that the packets spends in the queues that it visits, that is

$$\tau^*(s,d) \approx \sum_{v\in\mathcal{R}(s,d)} \hat{\mathcal{C}}_B(v) N^* \tag{33}$$

where the sum is over all nodes that the packet visits. The set $\mathcal{R}(s,d)$ is the subset of nodes obtained from the routing table. The total delay time can be approximated by considering

$$\tau(t) \approx \bar{\ell} + \frac{1}{S(S-1)} \sum_{s\in\mathcal{V}} \sum_{d\neq s\in\mathcal{V}} \sum_{v\in\mathcal{R}(s,d)} \hat{\mathcal{C}}_B(v)\, N(t) = \bar{\ell} + \mathcal{D}N(t) \tag{34}$$

where

$$\mathcal{D} = \frac{1}{S(S-1)} \sum_{s\in\mathcal{V}} \sum_{d\neq s\in\mathcal{V}} \sum_{v\in\mathcal{R}(s,d)} \hat{\mathcal{C}}_B(v). \tag{35}$$

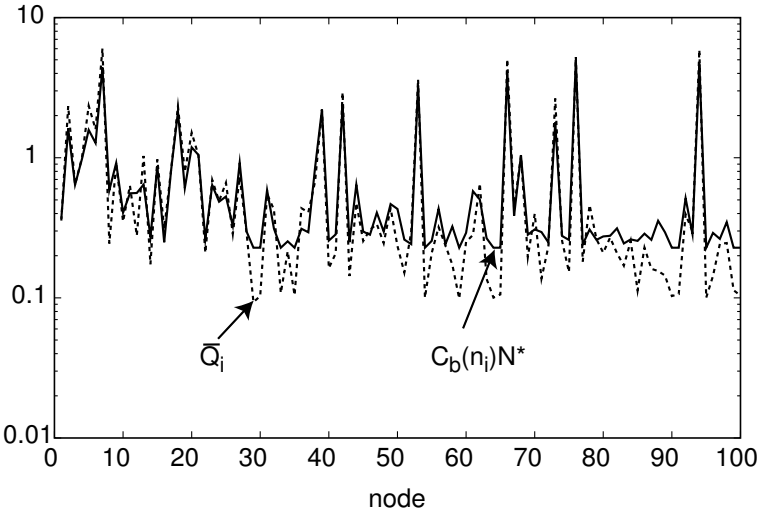

Figure 10: Comparison between the average queue size (dotted line) and its approximation using Eq. (32) (solid line). The network is a scale-free network with $S = 100$.

The approximation to the critical load is given by

$$\lambda_c = \frac{1}{\rho S \mathcal{D}}, \tag{36}$$

(cf. Eq. (30)). As shown in Fig. 9(b), this last equation gives a very good approximation of the phase transition from free flow phase to congested phase for the regular-symmetric and scale-free networks. It is not difficult to see why Eq. (36) also works for regular-symmetric networks. If the medial centrality measures how many routes use a particular node, then all nodes are utilised to the same degree in the regular case. The constant value for the medial centrality for each node is the same (see Eq. (17)) and is given by

$$\hat{\mathcal{C}}_B(w) = \frac{\mathcal{C}_B(w)}{\sum_{v \in \mathcal{V}} \mathcal{C}_B(v)} = \frac{1}{S}. \tag{37}$$

By substituting this last expression in Eq. (36), we find $\mathcal{D} = \bar{\ell}/S$ and the expression for the critical load obtained in Eq. (30) is recovered.

Equation (36) also reproduces behaviour observed by Fuks and Lawniczak [19] in a rectangular network. The addition of few random links provides a shortcut between distant parts of the network, but can quickly become congested. A fuller addition of links such as transferring from a square to triangular lattice has the more conventional effect of increasing the critical load, [4].

5.2 *LRD* at criticality

It is known that at the critical load the traffic statistics change from *SRD* to *LRD* [41]. The change of correlations from *SRD* (exponential) to *LRD* (power law) at criticality in phase transitions is a standard feature, and this phenomena also occurs in road traffic congestion.

The change from *Poisson* to *LRD* has been observed by analysing the time series for delays for 1 hop and 24 hop routes, and also by looking at the time series of average host and router queues lengths measured at each time tick. As mentioned before, the Hurst parameter distinguishes *SRD* traffic from *LRD*. The parameter can be obtained by considering the so-called *rescaled data*. More precisely, let X_t, $t \in \mathbb{Z}^+$ denote a discrete time series. The *adjusted range* is defined as

$$R(t,k) = \max_{0\le i\le k} R_i(t,k) - \min_{0\le i\le k} R_i(t,k), \tag{38}$$

where

$$R_i(t,k) = \sum_{l=1}^{t+i} X_l - \sum_{l=1}^{t} X_l - \frac{i}{k}\left(\sum_{l=t+1}^{t+k} X_l - \sum_{l=1}^{t} X_l\right) \tag{39}$$

The quantity $R(t,k)$ is normalized by the translated sample standard deviation

$$S(t,k) = \sqrt{k^{-1}\sum_{l=t+1}^{t+k} (X_l - \bar{X}_{t,k})^2} \tag{40}$$

where $\bar{X}_{t,k} = k^{-1}\sum_{l=t+1}^{t+k} X_l$. The *R/S* statistic is then defined to be

$$R/S = \frac{R(t,k)}{S(t,k)} \tag{41}$$

and it is fitted to the equation

$$\ln E[R/S] = a + H \ln k, \tag{42}$$

with H interpreted as the Hurst parameter. We considered separate *R/S* plots for 1 and 24 step journeys in a 32×32 network grid with 164 hosts for Poisson, and for *LRD*, sources for a range of load values. The network used had the same parameters as that used for Fig. 7 so the onset of congestion at a load of 0.3 was expected. At the smaller loads the network remains free of congestion. This means that time series of packet delays are stationary, and the original data may be used in measuring a value of H. For the higher values of the load, including $\lambda = 0.3$, congestion does occur, leading to an upward trend in delay times and queue sizes. In this case, the data are weighted to remove the trend, creating a stationary series. We observe that for $\lambda = 0.2$, $H = 0.5$ for both path lengths, indicating that very little *LRD* is present. This is corroborated by investigating the probability distributions of delays which both have the characteristic shape of exponentially decaying delay times. However, for $\lambda = 0.29$ H values are higher for both path lengths at the onset of congestion. The *R/S* plots show a kink with the steeper part of the curve corresponding to $H = 0.8$ in both cases. Hence the longer delays and lower frequencies do show significant *LRD*, but this is not seen at any other load value. Values for the three higher load values show very similar H values. Note that in these cases, the time series was weighted to remove the upward trend and the H values are all close to 0.5. This lack of any *LRD* seems to be caused by the phase change to the congested region above λ_c. The probability distributions for these higher values show delays shifting towards the length of the run (1×10^6 time ticks as the network becomes more and more congested). Here 1 and 24 step delays have a long tailed distribution, but this is caused by the non-stationarity of the data, not power-law autocorrelation decay.

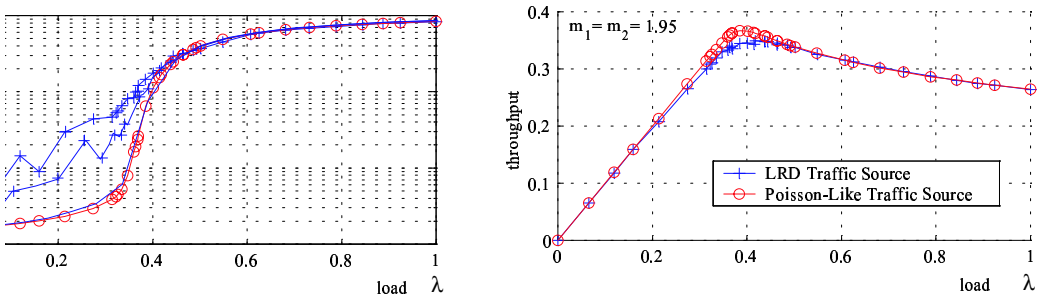

Figure 11: (a) Average packet lifetimes are plotted as a function of the load λ for Poisson sources and also for *LRD* sources with increasing average lifetime in the pre-congestion phase as m increases through $m_1 = m_2 = 1.5, 1.8,$ and 1.95. (b) Corresponding throughput for Poisson and *LRD* traffic are plotted as a function of the load λ for $m = 1.95$. Note that the lower peak value in throughput for the *LRD* traffic sources reflects the longer average lifetimes below the critical point. The peak differences diminish to zero as m is decreased to the Poisson-like value $m = 1.5$.

5.3 Congestion and *LRD* traffic

In the previous section, it was noted that *LRD* arose from interaction within the network and was not intrinsic to the traffic sources. In this section the Poisson-like sources are replaced with *LRD* sources, modelled using chaotic maps.

Fig. 11 gives a comparison of the onset of congestion in two otherwise identical Manhattan networks with host density $\rho = 0.164$, one Poisson sourced, and the other *LRD* sourced for different values of the Hurst parameter. The values of the intermittency parameters $m_1 = m_2 = m$ are kept equal in each case for simplicity. The critical load for this network is $\lambda = 0.3$.

Fig. 11(a) shows the average lifetime, or *end-to-end delay*, of a packet plotted against load λ, the average number of packets generated per host per unit time. As has been seen previously, there is a phase transition from the free phase in which lifetimes remain small to a congested phase in which lifetimes increase rapidly. Fig. 11(a) shows clear evidence of the earlier onset of congestion in the *LRD* traffic in comparison with Poisson traffic produced at the same rate. The data for Fig. 11 are shown for various values of $m_1 = m_2$. Values of $m_1 = m_2$ close to the maximum value of $m_1 = m_2 = 2$ give the highest degree of intermittency and hence the greatest contrast with Poisson traffic sources (corresponding to $m_1 = m_2 = 1.0$ in the intermittency model). The highest value used in our simulations, $H = 0.975$ has been observed in statistical investigations of real network traffic data.

Fig. 11(b) shows throughput versus load. The peak in the throughput occurs at the critical point. The network therefore reaches its peak efficiency at the critical point. The peak throughput is slightly lower for the *LRD* sources, emphasising the longer lifetimes of packets. Although the throughput is only slightly reduced, the average lifetimes increase by up to a factor of 10. This earlier onset appears to be the most important feature of *LRD* congestion within the context of the model, and has significant implications for shared backbone data network infrastructures.

The average queue sizes were shown to be closely related to the average lifetime of packets in [51]. The difference is clear for the average queue size for a rectangular network with *Poisson* and *LRD* traffic. In the *LRD* case the queues get congested more readily and as mentioned before this translates into longer lifetimes of the packets. High values of H for queue length distributions at hosts and routers have been

measured at all post critical loads for Poisson sources indicating the presence of strong network-induced *LRD* (see [52]).

6 Control of queue sizes

The simplest way to control packet traffic is to limit the length of queues [4]. As grid bar charts of node queue size have shown by Woolf *et al.* [52], long queues in the network invariably occur at hosts.
Valverde and Solé [45] introduced a *local* control mechanism where hosts modify their rates of packet release depending on the detected *local* rate of congestion. The sources adjust their local packet release by querying how busy their neighbours are. The number of congested neighbours a node can have is

$$\zeta = \sum_{v \in \mathcal{G}} \theta(n_v(t)) \tag{43}$$

where $\mathcal{G}$ is the set containing the neighbours of node, $n_v(t)$ is the number of packets in node v at time t and θ is the Heaviside function $\theta(x > 0) = 1$ and $\theta(x \leq 0) = 0$. The control mechanism consists of adjusting the rate of packet production $\lambda_i(t)$ using

$$\lambda_i(t+1) = \begin{cases} \min\{1, \lambda_i(t) + \mu\}, & \zeta < 4 \\ 0 & \zeta \geq 4. \end{cases} \tag{44}$$

If the neighbouring nodes are not congested the load increases at a rate of μ, and drops to zero if all the neighbours are congested. This control mechanism is inspired by the "additive increase/multiplicative decrease" [43] used in *TCP* packet networks. Valverde and Solé studied this control method in a Manhattan network with *Poisson* traffic and unlimited queue capacity. It was noticed that the packet release and the density of hosts satisfies the relation $\lambda \sim \rho^{-1}$ (see Eq. 24). On average, the packet release reaches a steady critical state. This does not mean that the local traffic creation rates are all similar, as the congestion levels vary widely between nodes. An interesting observation made by Valverde and Solé is the existence of a synchronization effect in the congestion state of distinct nodes.

6.1 Control in scale-free networks using *TCP*

The above model is an oversimplified packet network as the topology of a real network is, of course, not a regular network. Also, the model does not allow for packets dropped in transit (as occurs in the Internet) because of the unlimited queues on the nodes. In this section we introduce a more realistic model by considering a scale-free network generated using the *IG* algorithm with a power law decay index of 2.22. Similarly to the previous simulations, each node is designated as either a host or a router. Routers have a single routing queue that receives packets in transit across the network, and releases them back onto the network at a rate governed by the connectivity of the node. The difference with the previous examples is that for this network the simulation is of the fixed-increment time advance type rather than next-event time advance. This allows the routing queue service rate to be set as Ck^{α} packets per time tick of the simulation, where k is the degree of the router node, and C constant. This means that nodes with larger degrees produce more traffic. The index α has been chosen to be between 1 and 2 here. Hosts

have identical routing queues and function in the same way, but additionally act as sources. They have transmit buffers that hold packets generated by *LRD* and Poisson traffic sources until they have been acknowledged. The exact mechanism for this is described below.

The control mechanism uses a simplified version of *TCP* Reno [43] as the network protocol. This is the predominate protocol used on the Internet at present. This version is derived from that described in Erramilli *et al* [13]. It concentrates on the slow-start mechanism because in real networks this not only affects all connections, but is also the dominant effect for most connections. It also has a more marked effect on the network dynamics than congestion avoidance.

As in our previous sections, a double intermittency map is used as the basis for each *LRD* traffic source, and a uniform random number generator for each Poisson source. However, they are used in a slightly different way. One sojourn period in the '*on*' side represents a whole file which is then *windowed* using the *TCP* slow start algorithm: at the start of the file the window size is set to 1. Only a single packet is sent at that time tick. When a packet reaches its destination an acknowledgement is returned to the source: once this packet has been acknowledged, the window size is doubled and the next two packets are sent. When both these packets have been acknowledged, the window size is doubled again and a new window of packets is sent. The doubling process is repeated until the end of the file or the maximum window size (a fixed value). This is described mathematically in section 3.4 on *TCP* dynamics.

The routing algorithm uses a pre-calculated look-up table of shortest paths. All links between nodes are assumed to have unit length. At each time step packets are forwarded from the head of each routing queue. If an acknowledgement packet reaches its destination, this triggers the release of the next window of packets from that host.

A necessary modification of the traffic model was to make server strengths reflect the degree(and importance) of the node in the network . This has significant effects in increased throughput without overload, see [52].

Fig. 12 shows a comparison of different server strengths as α is varied. Two types of source are considered. The same *IG* network with the same pattern of hosts is used. Results from *LRD* sources and Poisson sources differ greatly. Throughputs (Figs 12(c) and 12(d)) at the lower server strengths (α equal to 1 and 1.1) are qualitatively similar: the throughput matches the load up to a threshold and then levels out. However, this threshold is more than 50% higher for the Poisson sources. If the exponent α is increased to a value of 1.5 this threshold can no longer be seen; throughputs for the two source types are similar. When servers are strongest ($\alpha = 2.0$) the situation is reversed: the network with *LRD* sources has a higher throughput, able to handle the maximum load applied to it without becoming overloaded.

Figs. 12(a) and 12(b) show average lifetimes for the two source types. At low loads behaviour is similar to that seen in our earlier work with regular lattices and open-loop protocols: lifetimes for Poisson sources are much less than for *LRD*. In the case of Poisson sources there is a similar transition from the free state to the congested state. At increasing server strengths the transition becomes less pronounced. This is due to the increased network capacity which slows the onset of congestion.

Comparisons have been made between the different topologies and packet transport algorithms to give four combinations [51, 52]:

– scale-free(produced using the *IG* model) with a closed-loop algorithm (*TCP*).

– Manhattan with an open-loop algorithm.

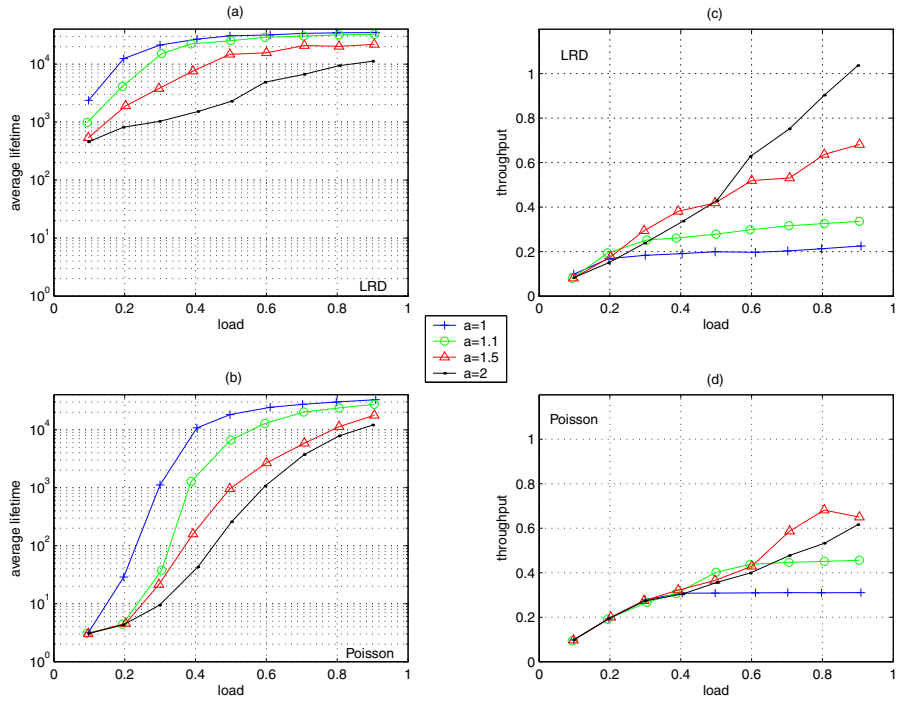

Figure 12: The number of packets that can be served at each time instant is increased according to a power law n^α, where n is the degree of the node and $\alpha = 1, 1.1, 1.5, 2$.

– scale-free(produced using the *IG* model) with an open-loop algorithm.

– Manhattan network with with a closed-loop algorithm.

The network had 1024 nodes in both cases and the host density was taken to be 589/1024. The index α of the server strength $0.25n^\alpha$ was fixed at 1 for all four combinations. The host distribution was random for the Manhattan network; by comparison the *IG* network connectivity 1 and 2 nodes were selected to be hosts.

The closed-loop *TCP* algorithm increases lifetimes for both regular and scale free networks. The requirement of *TCP* that packets be acknowledged before the next window of packets is sent is very restrictive in the sense that new windows cannot be sent by hosts more frequently than the round trip times (RTT) or ping times. Congestion is reacted to immediately because it causes an increase in RTT's and makes sources back off. Since file sizes are not that great and window sizes are often reset to 1 due to the round-trip time maximum RTO limit, throughput can never be that high. This results in packets being delayed in the transmit buffer and is the primary cause of increased packet lifetimes. By contrast the open-loop algorithm does not react to congestion and allows unlimited queues to build up at routers. For sufficiently high loads open-loop networks become congested and lifetimes approach those of the closed-loop networks.In fact a Manhattan network with an open-loop algorithm has longer lifetimes than an *IG* network with a closed-loop algorithm primarily because of the much shorter average path lengths in the *IG* network, see [52]. Identical networks using open-loop algorithms have higher throughputs; the *IG* network performs more efficiently for both types of algorithm. Similar behaviour is observed for Poisson sources.

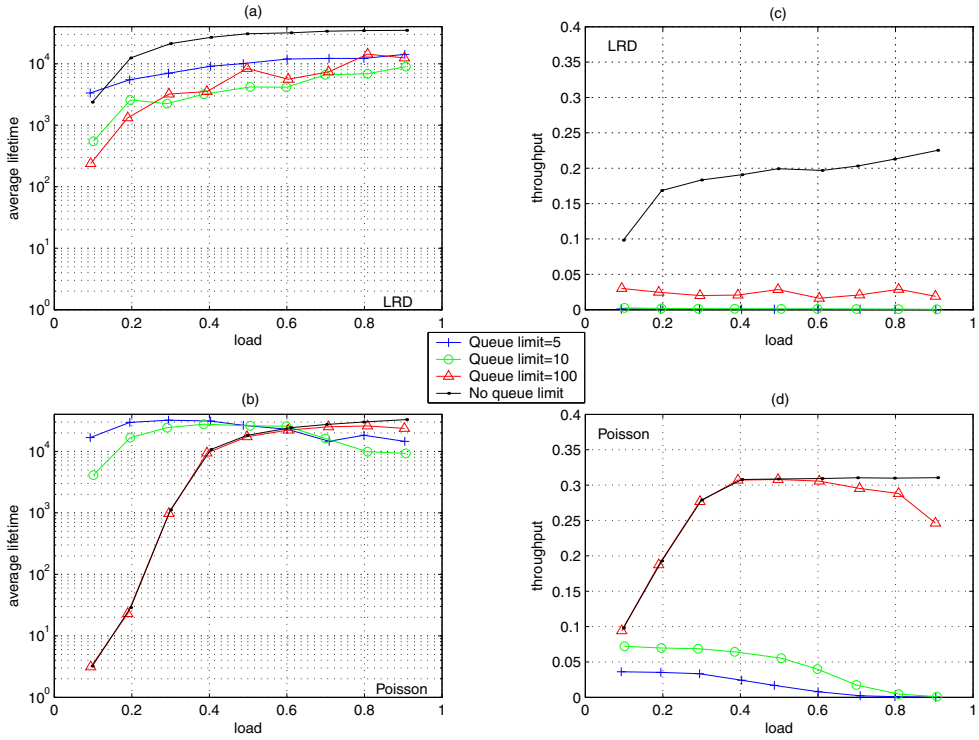

Figure 13: Effect of varying queue length limit on average packet life time and throughput for *LRD* and Poisson sources.

Thus, regular open-loop simulations are fundamentally different to the closed-loop simulations of *IG* networks and so detailed comparisons are not very useful.

In Fig. 13 packet dropping of real networks by limiting the routing queue lengths. Very severe packet loss has been modelled here in order to test the extreme situation. Real networks generally suffer much less packet loss. For *LRD* sources Fig. 13(a) average lifetimes are greatly reduced when the queue limit is decreased. When queue lengths are limited, packets are dropped at the routers and therefore re-sent more frequently. This causes shorter waits in the transmit buffer. This can be seen most clearly at the very low queue length limit. In the case of Poisson sources Fig. 13(b) average lifetimes behave quite differently. Lifetimes peak at a load of 0.3 for the queue limit of 5. This peak shifts to higher values as the queue limit increases. When there is no queue limit the lifetimes are the same in regular networks. Fig. 13(c) shows throughputs for *LRD* sources. These are greatly reduced when a queue limit is applied. At the smaller queue limits throughput is close to zero. In the case of Poisson sources (Fig. 13(d)) throughput is similar for the queue limit of 100, but also much less at the lower queue limits. However, throughputs are still much higher than for the *LRD* sources. This is caused by the shorter queues in the case of Poisson sources. Most queue lengths are less than a 100, meaning that this limit has little effect. The same *IG* network has been used, but hosts are now selected randomly with the same density of 589/1024. Results are similar. The network with randomly placed hosts always performs slightly better than the one with hosts placed at the low degree hosts. Average lifetimes are lower and throughput slightly higher. This shows that the inclusion of a small number of rich nodes in fact makes little difference.

References

[1] R. Albert, H. Jeong, and A. Barabási. Error and attack tolerance of complex networks. *Nature*, 406:378–381, 2000.

[2] R. H. Albert, A. Jeong, and A.-L. Barabási. Diameter of the world wide web. *Nature*, 401:130–131, 1999.

[3] D.K. Arrowsmith, R.J. Mondragon, J.M.Pitts, and M. Woolf. Internet packet traffic congestion. *Nonlinear Dynamics for Coding Theory and Network Traffic, in the IEEE Proceedings of Systems and Circuits*, 3:746–749, 2003.

[4] D.K. Arrowsmith, R.J. Mondragón, J.M. Pitts, and M.Woolf. Phase transitions in packet traffic on regular networks: a comparison of source types and topologies,. *Preprint*, 2004.

[5] D.K. Arrowsmith and M.Woolf. Modelling of TCP packet traffic in a large interactive growth network. *IEEE Proceedings of Systems and Circuits*, 5:477–480, 2004.

[6] A. L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 266:509–512, 1999.

[7] M. Barenco and D.K. Arrowsmith. The autocorrelation of double intermittency maps and the simulation of computer packet traffic. *Jnl of Dynamical Systems*, 19(1):61–74, 2004.

[8] J. Beran. Statistics of long memory processes,. *Monographs on Stats and Appl. Prob.*, 61, 1994. Chapman & Hall.

[9] B. Bollobás. Degree sequences of random graphs. *Discrete Maths.*, 33:1–19, 1981.

[10] B. Bollobás. *Random Graphs.* Academic Press, 1985.

[11] R. Cohen and S.Havlin. Scale–free Networks are Ultrasmall. *Phys. Rev. Lett.*, 90(5):058701, 2003.

[12] P. Erdös and A. Rényi. On Random Graphs I. *Publ. Math. Debrecen*, 6:290–297, 1959.

[13] A. Erramilli, M. Roughan, D. Veitch, and W. Willinger. Self-similar traffic and network dynamics. *Proc. of the IEEE*, 90(5):800–819, May 2002.

[14] A. Erramilli, R. P. Singh, and P. Pruthi. Chaotic Maps as Models of Packet Traffic. *In Proc. ITC 14, The Fundamental Role of Teletraffic in the Evolution of Telecommunication Networks*, pages 329–338, 1994.

[15] A. Erramilli, R. P. Singh, and P. Pruthi. An application of deterministic chaotic maps to model packet traffic. *Queueing Systems*, 20:171–206, 1995.

[16] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On Power-Law Relationships of the Internet Topology. *Proc. ACM/SIGCOMM, Comput. Commun. Rev.*, 29:251–262, 1999.

[17] H. H. Fowler and W. Leland. Local Area Network Traffic Characteristics, with Implications for Broadband Network Congestion Management. *IEEE Jour. on Sel. Areas in Comm*, 9:1139–1149, September 1991.

[18] L.C. Freeman. A set of measures of Centrality based on Betweeness. *Sociometry*, 40:35–41, 1977.

[19] H. Fukś and A. T. Lawniczak. Performance of data networks with random links. *Mathematics and Computers in Simulation*, 51:103–119, 1999.

[20] H. Fukś, A.T. Lawniczak, and S. Volkov. Packet delay in data network models. *ACM Transactions on Modeling and Computer Simulation*, 11(3), 2001.

[21] K. Fukuda, H. Takayasu, and M. Takayasu. Origin of critical behavior in Ethernet traffic. *Physica A*, pages 289–301, 2000.

[22] A. Giovanardi, G. Mazzini, and R. Rovatti. Chaos based self–similar traffic generators. *Proc. NOLTA*, pages 747–750, 2000.

[23] K. I. Goh, E. Oh, K. Kahng, and D. Kim. Betweenness centrality correlation in social networks. *Phys. Rev. E*, 67:017101, 2003.

[24] P. Holme and B. J. Kim. Vertex Overload Breakdown in Evolving Networks. *Physical Review E*, 65:066109, 2002.

[25] C. Huang, M. Devetsikiotis, I. Lambadaris, and R. Kaye. Fast Simulation for Self-Similar Traffic in ATM Networks. In *Gateway to Globalization, 1995 IEEE International Conference on Communications*, volume 1, pages 438–444, Seattle, Washington, 1995. IEEE ICC95.

[26] B. E. Huberman and R. M. Lukose. Social dilemmas and Internet congestion. *Science*, 277:535–537, 1997.

[27] R. Jain and S. A. Routhier. Packet trains: Measurements and a new model for computer network traffic. *IEEE Journal on Selected Areas*, 4:986–995, 1986.

[28] L. Kleinrock. *Queueing Systems. v. 1. Theory.* John Wiley & Sons, 1975.

[29] W. E. Leland, M. S. Taqqu, W. Willinger, and D. Wilson. On the Self-Similar Nature of Ethernet Traffic (Extended Version). *IEEE/ACM Trans on Networking*, 2, No 1(1):1–15, Feb. 1994.

[30] H. Li and M. Maresca. Polymorphic-torus network. *IEEE Transs Comp.*, 38(9):1345–1351, 1989.

[31] B. Mandelbrot. Self-Similar Error Clusters in Communication Systems and the Concept of Conditional Stationarity. *IEEE Trans. On Communication Technology*, COM-13:71–90, March 1965.

[32] J. L. McCauley. *Chaos, Dynamics and Fractals and Algorithmic Approach to Deterministic Chaos.* Cambridge University Press, 1995.

[33] R. J. Mondragón. A Model of Packet Traffic using a Random Wall Model. *Int. Jou. of Bif. and Chaos*, 9 (7):1381–1392, 1999.

[34] R. J. Mondragón. Congestion and centrality. *in preparation*, 2004.

[35] I. Norros. Studies on a model for connectionless traffic, based on fractional brownian motion. *Conf. On Applied Probability in Engineering, Computer and Communication Sciences*, Paris:16–18, June 1993.

[36] T. Ohira and R. Sawatari. Phase Transition in Computer Network Traffic Model. *Physical Review E*, 58:193–195, 1998.

[37] Y. Pomeau and P. Maneville. Intermittent transition to turbulence in dissipative dynamical systems. *Commun. Math. Phys.*, 74:189–197, 1980.

[38] P. Pruthi. *An application of chaotic maps to packet traffic modelling.* PhD thesis, KTH, Stockholm, Sweden, Oct. 1995.

[39] P. Pruthi and A. Erramilli. Heavy-Tailed ON/OFF Source Behaviour and Self-Similar Traffic. *ICC 95*, June 1995.

[40] M. Roughan and D. Veitch. A Study of the Daily Variation in the similarity of Real Data Traffic. *Tech. Rep. 0070, SERC, Software Engineering Research Centre, Level 3, 110 Victoria St, Carlton Vic, 3053, Australia*, 1997.

[41] R. V. Solé and S. Valverde. Information transfer and phase transitions in a model of internet traffic. *Physica A*, 289:595–605, 2001.

[42] W. Stallings. *Data and Communications.* Prentice Hall Int. Inc., 2000.

[43] W. R. Stevens. *TCP/IP Illlustrated*, volume Volume 1: The Protocols. Addison-Wesley, 1994.

[44] L. Subramanian, S. Agarwal, J. Rexford, and R. H. Katz. Characterizing the Internet Hierarchy from Multiple Vantage Points. *Proc. of INFOCOM 2002*, June, 2002.

[45] S. Valverde and R.V. Solé. Self-organized critical traffic in parallel computer networks. *Physica A*, 312:636, 2002.

[46] D. Veitch. Novel Models of Broadband Traffic. In *Proc. Globecom '93*, pages 362–368, Murray River, Australia, 1992. 7th Australian Teletraffic Research Seminar.

[47] X-J Wang. Statistical physics of temporal intermittency. *Phys Rev A*, 40(11):664761, 1989.

[48] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson. Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level. *IEEE/ACM Transactions on Networking*, 5(1):71–86, 1997.

[49] R.J. Wilson. *Introduction to Graph Theory.* Longman Group Limited, Whistable Kent, Great Britain, 1972. ISBN:0-582-44762-3.

[50] J. Winick and S. Jamin. Inet–3.0 Internet Topology Generator. *Tech. Report UM-CSE-TR-456-02, University of Michigan*, 2002.

[51] M. Woolf, D.K. Arrowsmith, R.J. Mondragón, and J. M. Pitts. Optimization and phase transition in a choatic model of data traffic. *Physics Reviews E*, 66:056106, 2002.

[52] M. Woolf, D.K. Arrowsmith, S. Zhou, R.J.Mondragón, and J.M. Pitts. Dynamical modelling of *tcp* packet traffic on scale-free networks. *submitted to Phys Rev E*, 2004.

[53] S. Zhou and R. J. Mondragón. Towards Modelling the Internet Topology – The Interactive Growth model. In *Proc. of ITC18*, 2003.

[54] S. Zhou and R. J. Mondragón. Accurately modelling the internet topology. *submitted to Physical Review E*, 2004.

[55] S. Zhou and R. J. Mondragón. Redundancy and robustness of the *as*-level internet topology and its models. *IEE Electronic Letters*, 40(2):151–152, January 2004.