Inferring the background traffic arrival process in the Internet

Péter Hága,* István Csabai,† and Gábor Vattay‡
Department of Physics of Complex Systems, Eötvös Loránd University, Budapest, Hungary
(Received 13 March 2009; revised manuscript received 11 October 2009; published 3 December 2009)

Phase transition has been found in many complex interactivity systems. Complex networks are not exception either but there are quite few real systems where we can directly understand the emergence of this nontrivial behavior from the microscopic view. In this paper, we present the emergence of the phase transition between the congested and uncongested phases of a network link. We demonstrate a method to infer the background traffic arrival process, which is one of the key state parameters of the Internet traffic. The traffic arrival process in the Internet has been investigated in several studies, since the recognition of its self-similar nature. The statistical properties of the traffic arrival process are very important since they are fundamental in modeling the dynamical behavior. Here, we demonstrate how the widely used *packet train technique* can be used to determine the main properties of the traffic arrival process. We show that the packet train dispersion is sensitive to the congestion on the network path. We introduce the packet train stretch as an order parameter to describe the phase transition between the congested and uncongested phases of the bottleneck link in the path. We find that the distribution of the background traffic arrival process can be determined from the average packet train dispersion at the critical point of the system.

DOI: 10.1103/PhysRevE.80.066103 PACS number(s): 89.20.Hh, 89.70.−a

I. INTRODUCTION

The Internet is in the focus of increased interest in the last decade and the Internet traffic is growing at an extraordinary pace. Several studies appeared concentrating on the large scale properties of the Internet topology and the characterization of network traffic in the Internet research. The topology and traffic dynamics are the key examples of complex network research, which is the most dynamic field of modern statistical physics. Complex network models are often rooted from some theoretical consideration and often hard to directly map to real life networks. That is why it is so important to show up examples where the model keeps the attributes and structure of the real network and in the meantime, the emergent complex behaviors such as phase transition can be observed and understood from the microstructure of the systems.

Large scale properties of the Internet topology are studied [1,2], routing strategies are investigated [3] and also their relation of traffic congestion phases are identified [4]. Several papers concentrate on the properties of network traffic [5–9], while many authors detected fractal properties in traffic time series [10–12]. Fractal traffic properties and self-similar time series are usually attributed to heavy-tailed distributions of objects at the traffic sources, e.g., sizes of transferred files [13,14] or time delays in user interactions [15]. Recently, the dynamical origins [16–22] of self-similarity also attract increasing attention. Despite the large number of publications that are investigating the fractal properties of network traffic, wide area experimental values for the statistical properties of the background traffic are not available to verify the models. The reason of lacking real wide area experimental values is that the observation of packet flows is based on packet trace collection at a given router [23] and these network routers are usually not accessible due to company policies. Only a limited number of real traffic traces collected at certain points of the network [24–26] are available publicly. A number of approaches try to cope with measuring and inferring network traffic parameters in other way [29,30]. These methods do not collect traces of a certain router, but uses a sender and a receiver node at the two end points of the path and investigates the properties of the network path between them. The sender injects artificial traffic flows or network packets into the network and observes their delay and other parameters at the receiver. Usually the packet delays are used to infer the network and traffic parameters. One of the main advantage of these methods is that they do not require the cooperative behavior of the network administrators, Internet service providers, since the probing is based on injecting standard Internet packets toward the receiver.

Recently, important results appeared [27,28] that are investigating the scaling of flow fluctuations. These results cope with the arrival process of the traffic flows. They found a power law relationship between the fluctuations and the average of the flux (bandwidth) of the traffic. The authors of [27] also studied real Internet traffic traces to determine the scaling of fluctuations. In this paper, we present a similar approach that aims to infer the background traffic arrival process. This is done by the analysis of packet train experiments. We present the power law relation between the probe train length and the value of the *average stretch* that is used to determine the fluctuations of the background traffic flow.

The rest of our paper is organized as follows. In Sec. II we give a brief introduction to the used network model and the packet train measurement method. We describe the fluid model and also the observable packet train dispersion curve and their main properties in Sec. III. We introduce the stretch parameter as an order parameter to describe the phase transition between the congested and uncongested phases in Sec.

*haga@complex.elte.hu
†csabai@complex.elte.hu
‡vattay@complex.elte.hu

IV. In Sec. V, we demonstrate how the average stretch of the packet train converges to the fluid limit and can used for determining the arrival process of the background traffic in the critical point. We show that the convergence of the power law behavior is the key in our inferring method. In Sect. VI, we explain the behavior of the average stretch function in the purely congested and purely uncongested phases in detail.

II. NETWORK AND MEASUREMENT MODEL

Although the details of the methodology seems to be somewhat technical there are few key issues we have to understand to be able to set up a faithful model, we summarize the general framework in this section.

Active probing is a class of measuring methods that are used to infer various characteristics of network paths [29,31]. The common in these methods is that they involve probe packets that are injected into the network, while a receiver side analyzes the observed responses. This general framework admits the determination of the topology of a network [32,33], the link bandwidths on a path [34–36], and the statistics of packet size, packet loss and delay along a route [37,38]. These properties are important for quality of service considerations, because together they determine the rate at which applications can send data on the route.

We model the network as *single hop* network. In this case the network packets arrive into a single waiting queue and they are processed by a single server. We assume *first-in-first-out* (FIFO) queuing policy, so the arriving packets have to wait until all the preceding packets are processed before. This waiting is denoted by D_q and called queuing delay. After the queuing the packet arrives to the processing server. During the processing the processing delay $D_{pr}=p/C$ elapses, where p denotes the size of the packet in bits and C denotes the processing capacity of the server counted in Mbits/s. The C capacity is often called as the physical bandwidth of the network. The concurrent traffic flows (either real data transportation or measurement flows) entering to the same network hop has to share the waiting queue and the processing capacity with all the other traffic flows. From measurement point of view we distinguish the *probe traffic* (containing our measurement packets) and the *background traffic* (containing all the other traffic) their rates are measured in Mbits/s. The background traffic can be assumed as constant sized packets (without restricting the model [34]) and their arrival process will be in the focus of our investigation. In general, network models cannot be developed without certain assumptions on the background traffic arrival model. Our work aims to present an estimation method to determine the distribution of the background traffic arrival process with probing methods.

In general, a probe traffic may contain n probe packets of different sizes p_i, $i \in \{1 \ldots n\}$ sent with different interdeparture times (*input spacing*) $\delta_i = t^d_{i+1} - t^d_i$, $i \in \{1 \ldots n-1\}$. The actual choice of p_i and δ_i determines the architecture of the probe stream that may vary according to the particular quantity under investigation (e.g., bandwidth, distribution, or spectrum of end-to-end delays). Traversing the network the probe packets interact with the background traffic as can be seen in the subfigures of Fig. 5. The initial spacing (e.g., the interdeparture time) of probe packets change due to the interactions with the background traffic flows [see Figs. 5(b)–5(d)]. They can also suffer extra queuing delays, since the background traffic together with the probe traffic can be more than the physical capacity of the link [Fig. 5(d)]. The analysis usually based on the arrived packets spacing (*output spacing*) $\delta'_i = t^a_{i+1} - t^a_i$, $i \in \{1 \ldots n-1\}$. In most of the applications it is customary to send constant sized packets regularly ($p_i = p$ and $\delta_i = \delta$), this probe pattern is called *packet train*. Other probe patterns are also common in the networking practice. For instance to measure bottleneck bandwidth one can send packet-pairs with an interpair time chosen randomly from an exponential distribution, while the packets in a pair are sent in a back-to-back fashion [36]. Another example is the packet tailgating technique of [37], where pairs consisting of a packet with the highest possible size immediately followed by a packet with the smallest possible size are sent to measure the bandwidth of each link on a path. In this paper, we use only packet trains (e.g., regularly sent uniform sized packets), that can be used for measure the packet train dispersion.

III. PACKET DISPERSION IN CASE OF FLUID TRAFFIC

In our study we use packet trains to explore the dispersion curve. We inject packet trains of constant sized packets with the same input spacing between them [see Fig. 5(a)] and we interested in the average output spacing of the packets as a function of the input spacing. The input spacing between the packets of the probe train is same for all the probe packets and denoted by δ. Thus the average input spacing is $\delta_n = \delta$. The δ'_n output spacing [see Figs. 5(b)–5(d)] of the probe train that consist n packets are written

$$\delta'_n = \frac{1}{n-1} \sum_{i=1}^{n-1} \delta'_i . \tag{1}$$

The general formula for the packet train dispersion function is

$$\delta'_n = f(\delta_n). \tag{2}$$

The exact form of $f(\,)$ depends of the details of the applied background traffic model [34]. In our study, it is enough to model the $f(\delta)$ dispersion curve in the fluid approximation of the background traffic. In this approximation, the background traffic is considered as a fluid of infinitely small packets. In this case, the average bandwidth of the background traffic is kept constant, while the average packet size of the background traffic goes to zero [39].

Let us consider a single-hop network with C physical capacity, B background traffic rate. Using the notation introduced in Sec. II the δ input and $\delta' = \delta'_2$ output spacing for a packet train containing $n=2$ probe packets can be written:

$$\delta = t^d_2 - t^d_1 = (t_0 + \delta) - (t_0), \tag{3}$$

$$\delta' = t_2^a - t_1^a = \left(t_0 + \delta + D + D_q + \frac{p}{C}\right) - \left(t_0 + D + \frac{p}{C}\right), \quad (4)$$

where t_0 denotes the staring time of the experiment, D is the link delay, D_q is queuing delay and p/C is a processing delay of the probe packet.

In the case of the probe traffic rate is smaller than the free capacity of the server ($p/\delta < C-B$) the queuing delay will be $D_q=0$, thus the separation of the two probe packets remain as it was at the time of starting:

$$\delta' = \delta. \quad (5)$$

In the case of the probe traffic rate exceeds the free capacity of the server ($C-B < p/\delta$) the second probe packet suffers extra delay while the accumulated background traffic is processed, that is the D_q queuing delay cannot be neglected: $\delta' = \delta + D_q$. The background traffic starts accumulating right after the arrival of the first probe packet and go on till the second probe packet arrives at the waiting queue. The queue length can be formulated as the difference of the amounts of the incoming ($p+B\delta$) and outgoing ($C\delta$) traffic. With a division of C processing rate we can get the queuing delay $D_q=(p+B\delta-C\delta)/C$. With an algebraic transformation we get:

$$\delta' = \frac{p}{C} + \frac{B}{C}\delta \quad (6)$$

for the δ' output spacing as a function of the B background traffic rate, the C capacity, the p probe packet size and the δ input spacing.

The two cases can be written together and called as the *fluid model* of the packet dispersion:

$$\delta' = \begin{cases} \frac{p}{C} + \frac{B}{C}\delta & \delta \le \delta_c \\ \delta & \delta \ge \delta_c, \end{cases} \quad (7)$$

where $\delta_c = p/(C-B)$, which separates the congested and uncongested phases of the network. This model of dispersion is able to capture the transition between two important phases of network operation and can be extended easily for probe trains consisting $n>2$ number of packets. For general probe trains containing n packets the form of the dispersion curve is same as Eq. (7) with substituting $\delta=\delta_n$ and $\delta'=\delta'_n$ average values.

The dispersion curves observed in packet level simulations or in real world experiments deviate from the result of the fluid model. The source of the deviation is that the assumption of infinitely small packets is not hold in real world experiments. The fluid model describes correctly the dispersion curve for asymptotically small and large δ input spacings, but it cannot handle the transitional region between the congested and uncongested phases. Besides its weakness in real world applications the fluid model gives us the lower bound of the real dispersion curves [39], which are very useful in our investigation.

Here, we just note that based on diffusive approximation a *granular model* of the dispersion curve for $n=2$ can be derived in closed form [34], even for the general case, where the background traffic packet sizes are finite values. The granular model describes the observed packet-pair dispersion curves correctly even in their transitional region, but the closed form result is only available for $n=2$ packet-pair scenarios. In our work, we focus on the deviation of the observed dispersion curve and the dispersion curve based on the fluid model as a function of the probe train length n and, thus, the granular model cannot be applied here.

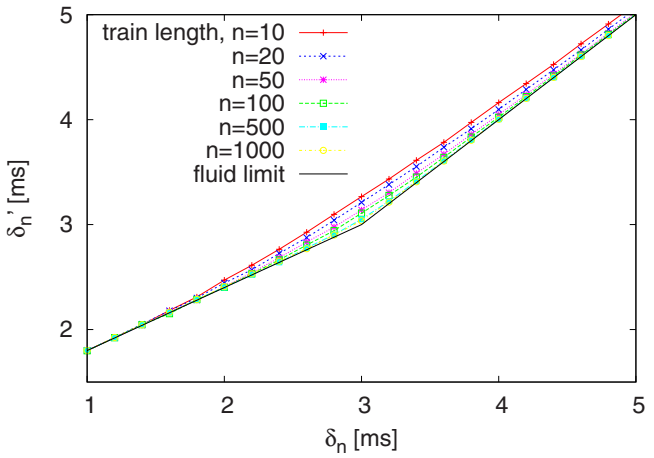

FIG. 1. (Color online) The average output spacing curves for different train lengths.

IV. PHASE TRANSITION

In our study, we investigate the convergence of the deviation of the observed dispersion curve from the fluid limit as a function of the packet train length we use the fluid-model equations. Next, we show results of packet level simulations. Since the simulation considers both the probe and background traffic packet sizes, the observable dispersion corresponds to the real wide area experiments and not agrees with the fluid model. In Fig. 1, the dispersion curves $\langle \delta'_n \rangle = \langle f(\delta_n) \rangle$ for different train lengths are shown. Each data point represents an average for 600 packet train measurement events. The physical bandwidth $C=10$ Mbps, the average background traffic rate $B=6$ Mbps, background traffic packet size $P=12000$ bit, and the size of the probe packets $p=12000$ bit are the same for both sets of parameters. We plot the average output spacing δ' as a function of the input spacing δ. The different symbols represent the results of the packet level simulation with train length from $n=10$ to 1000. In these simulations, we used a Poisson arrival process for the background traffic. As a limiting case and a lower bound, we also show the curve in fluid approximation. The deviation of the observed curves from the fluid limit can be clearly seen, while this deviation decreasing with the increasing train length. The characteristic breakpoint is at the values of $p/(C-B)=3$ ms. Now, we introduce a new parameter called *relative stretch* to describe the average change of the probe packet spacing of the train traversing the network for a given probe train length n:

$$\Delta = \frac{\delta'_n - \delta_n}{\delta'_n}. \quad (8)$$

This quantity serves as an order parameter that describes the phase transition between the congested and uncongested

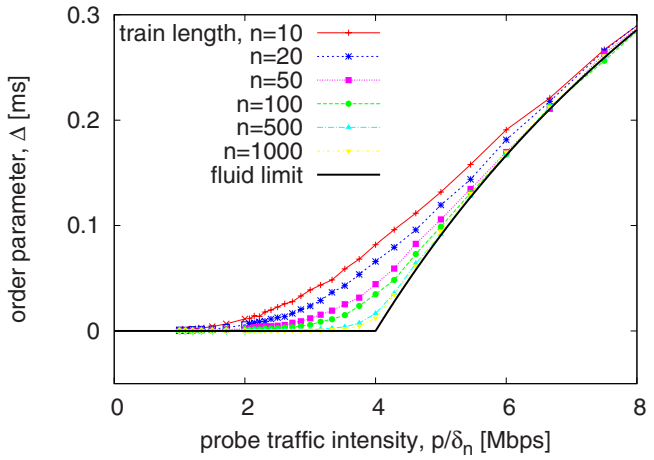

FIG. 2. (Color online) The order parameter ($\triangle$) as a function of probe traffic intensity (p/δ_n) for different train lengths.

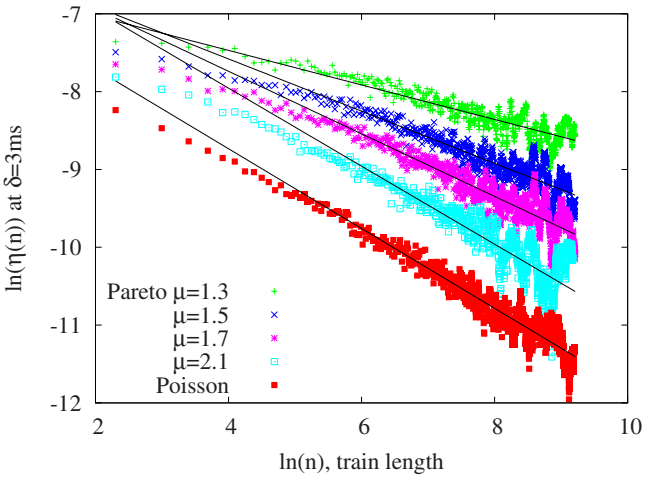

FIG. 3. (Color online) The deviation of the average output spacing from the fluid limit as a function of the train lengths for Poisson and Pareto arrival processes. All the 5 data sets are taken at the critical point ($\delta_c=3$ ms) of the dispersion curves.

phases [40]. In Fig. 2, the order parameter Δ can be seen as a function of the probing rate (probe traffic intensity) p/δ_n. It can be clearly seen that below the critical point ($p/\delta_c=4$ Mbps, which corresponds to $\delta_c=3$ ms as above) the observed Δ order parameter is zero for the fluid limit and above the critical point the order parameter is increasing up to 1. The order parameter for the packet train measurements shows finite size scaling phenomena, as the train length grows the deviation from the fluid limit disappears. In the next section, we investigate this convergence at the critical point.

V. DETECTING THE ARRIVAL PROCESS IN THE CRITICAL POINT

In this section, we introduce the *average stretch* η to investigate the deviation of the average output spacing from the fluid limit as a function of the train length n in the critical point. This scenario is presented in Fig. 5(c) when the probe traffic rate and the background traffic rate together just fill the server capacity, but not overload it. We expect that the properties of the convergence of the δ'_n to the δ'_{fl} are related to the background traffic arrival process, since its characteristics determine the deviation in $n=2$ packet-pair scenarios [39].

The average stretch can be written:

$$\eta(n) = \langle \delta'_n \rangle - \delta'_{fl}, \tag{9}$$

where δ'_n is the average output spacing for the n length packet train and δ'_{fl} is the output spacing calculated by the Eq. (7) fluid approximation. The $\eta(n)$ value can be considered as

$$\eta(n) = \langle \xi \rangle = \langle \delta'_n - \delta'_{fl} \rangle, \tag{10}$$

where the ξ values are independent and identically distributed (iid) random variables that are depend on the δ_n input spacing and the statistical properties of the background traffic. Based on the generalized central limit theory, one can show that

$$\eta(n) \sim n^{\alpha}. \tag{11}$$

If ξ has finite variance then the sum tends toward the normal distribution and $\eta(n) \sim n^{-0.5}$ as $n \to \infty$. If ξ has infinite variance then the sum tends to the Pareto distribution with $\eta(n) \sim n^{\alpha}$ as $n \to \infty$, where α is related to the shape parameter of the Pareto distribution. In our case the α exponent describes the arrival process of the background traffic. The arrival process is a fundamental property of the network traffic that describes whether the traffic flows are independent or rather correlated in time. Based on the $\eta(n)$ function we are able to distinguish between the typical arrival processes and we can determine also the shape parameter of the Pareto distribution in the case of. With some algebraic work it can be shown that

$$\alpha = \begin{cases} -0.5 & \text{for Poisson and Weibull arrival processes} \\ \dfrac{1}{\mu} - 1 & \text{for Pareto arrival process,} \end{cases} \tag{12}$$

where μ is the shape parameter of the Pareto distribution. The Pareto distribution shows heavy-tailed properties in the case $1<\mu<2$. Otherwise, if $\mu>2$ the observable behavior is rather Poisson-like, with $\alpha=-0.5$.

The power law behavior makes it possible to infer the background traffic arrival process. As a next step, we can investigate the exponents of the power law relation between the average stretch and the train length. Several simulated scenarios were performed with different kind of background traffic arrival processes. Besides the arrival process the other important parameters are set to be the same: $C=10$ Mbps, $B=6$ Mbps, $P=12000$ bit, and $p=12000$ bit. We used Poisson, Weibull and Pareto arrival processes to model both short range and heavy-tailed distributions. The shape parameter of the Pareto distribution was varied in order to study the effect of the heavy-tailed behavior.

In Figs. 3 and 4 we present the average stretch as a func-

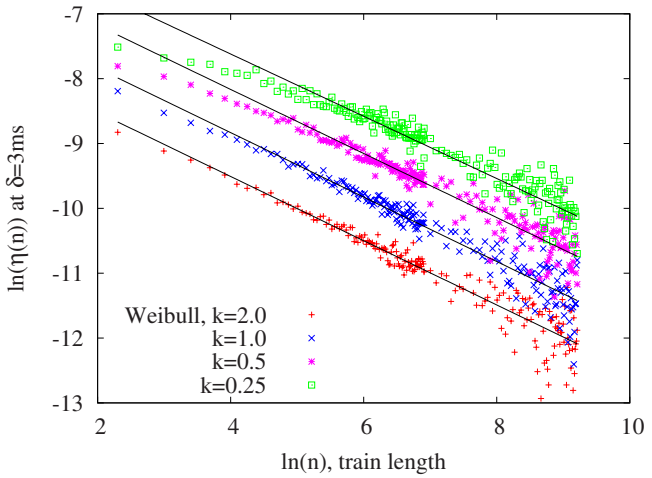

FIG. 4. (Color online) The deviation of the average output spacing from the fluid limit as a function of the train lengths for Weibull arrival process. All the data sets are taken at the critical point (δ_c=3 ms) of the dispersion curves.

tion of the probe packet train length for different arrival processes. The both axis have logarithmic scale to show the power law relation between the quantities. In this figure, the system is studied in the critical point of the phase transition δ_c=3 ms, p/δ_c=4 Mbps. From the slopes of the fitted linear it can be seen that different exponents belong to different arrival process. The fitted exponents are summarized in Table I to compare them with the calculated exponents. The calculation of the exponents are based on Eq. (12). The fitted exponents match to the calculated theoretical values, which shows that with packet train measurements one can infer the arrival process of the background traffic in the bottleneck queue.

VI. EXPLANATION OF THE STRETCH FUNCTION AT $\delta \neq \delta_c$

In the previous section, we presented how the arrival process of the background traffic can be determined in the critical point of the phase transition. In this section we provide the explanation of the observed average stretch for the $\delta > \delta_c$ and $\delta < \delta_c$ cases. The exact running of the $\eta(n)$ function depends on the distance of the actual δ and δ_c value, which distance is related to the δ input spacing, the B background traffic rate and also the C physical capacity of the network. In case of $\delta < \delta_c$ the probe traffic exceeds the free capacity of the network that leads to overloading the server and queuing in the buffers. In this case, the probe and background traffic packets are leaving the investigated network as a continuous flow. If $\delta > \delta_c$ the probe traffic rate is less then the long term average of the free capacity of the network, but on short term (in the characteristic length of a probe train) the background traffic fluctuations can cause temporally overload in the network with high probability. This probability decreases as the average probe traffic rate is decreased (as the δ probe packet separation is increased).

The described scenarios are presented in Fig. 5, where we defined the packet sequence numbers in the probe train in the order of the leaving of the sender from 1 to n. Four cases can be recognized: a) the injected packet train with well defined packet spacing and without background traffic, b) the probe train with background traffic in the uncongested phase, c) the packet train at the critical point (described in Sec. V. and d) the fully loaded line in the congested phase. In the uncongested phase [Fig. 5(b)] there is a k^{th} packet that separates the probe train into two parts in the following way. The first part (consisting of k number of packets) called *unchanged subtrain* that does not suffer stretch and a second part (built up by $n-k$ packets) called *stretched subtrain* with increased train length due to the background traffic burst entered between the packets of the second part. The limiting k^{th} packet can be clearly seen in the center of Fig. 5(b), while the unchanged subtrain takes the right part (from 1 to k) and the stretched subtrain takes the left part (from k to n) of the schematic probe train.

TABLE I. Comparison of the fitted and calculated exponents for different traffic arrival processes. The fitted values match very well to the calculated ones, which allows us to infer the background traffic arrival process from these kind of experiments. The data sets and fitted lines are the same as in Figs. 3 and 4.

	Calculated exponent	Fitted exponent
Poisson process	−0.50	−0.51
Weibull, k=2.0	−0.50	−0.50
Weibull, k=1.0	−0.50	−0.49
Weibull, k=0.5	−0.50	−0.50
Weibull, k=0.25	−0.50	−0.48
Pareto, μ=2.1	−0.50	−0.50
Pareto, μ=1.7	−0.41	−0.40
Pareto, μ=1.5	−0.33	−0.33
Pareto, μ=1.3	−0.23	−0.22

A. Uncongested phase: $\delta > \delta_c$

Although in the case of $\delta > \delta_c$ the input spacing region corresponds to the uncongested phase, the δ'_n dispersion can be larger than δ'_{fl}. This deviation can be clearly seen in Fig. 1 around δ=4 ms. On average the distance between the probe packets are larger than what the background traffic packets can fill. This could lead to the false conclusion that the probe train length is not changing as the train interacts with the background traffic. This conclusion not holds, since the background traffic bursts entering between the last few probe packets increase in the probe train length if the size of the background traffic burst exceeds the size of the gaps between the last probe packets. In that case, the free capacity remaining between the lasts probe packets is smaller than the background traffic rate of a traffic burst. This phenomena can be recognized in Fig. 5(b).

To determine the probe train stretch in the uncongested phase we have to investigate the effect of the background traffic entering between the last l probe packets of the probe train. This framework enables the background traffic to enter between any two probe packets with sequence number larger

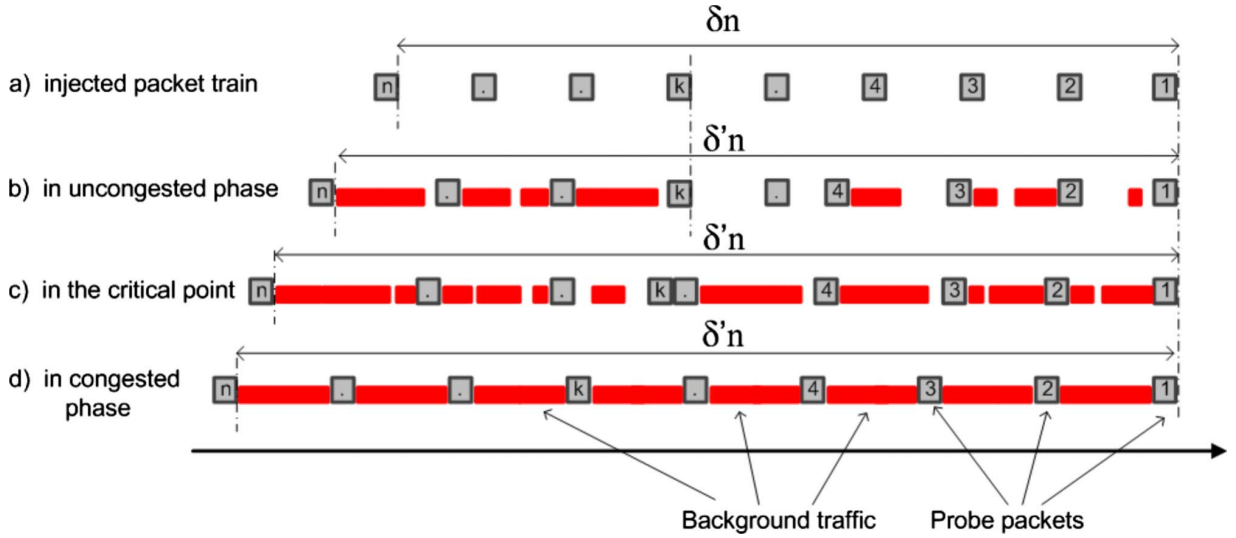

FIG. 5. (Color online) Graphical representation of the probe packet train dispersion: a) at departing from the sender; interring with the background traffic b) in uncongested case; c) in the critical point; d) in congested traffic conditions.

than l, thus, the formula describes the situation for l includes all the situations for sequence numbers that are smaller than l. In this case the sequence number of the packet that separates the unchanged and stretched subtrains is $k=n-l$. The formula for the average stretch for general l values can be written in the following way

$$\begin{aligned}\eta(n) &= \delta'_n - \delta'_{fl} \\ &= \frac{1}{n-1}\sum_{i=1}^{n-1}\delta'_i - \delta'_{fl} \\ &= \frac{1}{n-1}[(n-l-1)\delta + \delta'_{l-1} - (n-1)\delta_{fl}] \\ &= \frac{1}{n-1}\left[(n-l-1)\delta + \sum_{j=n-l-1}^{n-1}\left(\frac{p}{C}+\frac{p_{Bj}}{C}\right) - (n-1)\delta\right] \\ &= \frac{1}{n-1}(S-l\delta) \\ &= \frac{1}{n-1}\cdot K \sim n^{-1}, \end{aligned} \tag{13}$$

where the output spacing is the sum of $n-l-1$ times of the initial spacing δ for the first $n-l$ packets and the δ'_{l-1} output spacing that consists the sum of the probe packet spacings for the last l probe packets (packets from $n-l$ to n). The exact value of δ'_{l-1} probe packet spacing is not important right now, but we notice that this is build up by the probe packet service time p/C and the service time of the jammed background traffic p_{Bj}/C for the last l probe packets. The sum of these service times is denoted by S that is independent from the n packet train length. The sum of $(n-l-1)$ δ and $(n-1)\delta'_{fl}$ is $l\delta$ independent from n, since $\delta'_{fl}=\delta$ if $\delta > \delta_c$ from Eq. (7). The final result is that for large n value the $\eta(n)$ average stretch scales as n^{-1}.

In Fig. 6, the results from packet level simulations can be seen. We plotted the $\eta(n)$ average stretch as the function of the n probe train length. We investigated this relation using several kinds of background traffic arrival processes with different parameters. One can see that the values follow power law functions with a slope of $\alpha=-1$ for all the different arrival processes, which confirms the above analytical result and can be summarized as

$$\langle \eta(n) \rangle \sim n^{-1} \quad \text{for} \quad \delta > \delta_c \tag{14}$$

independent from the kind of arrival process.

B. Congested phase: $\delta < \delta_c$

Next, the congested phase is investigated, where $\delta < \delta_c$. In this case, the probe traffic and the background traffic together exceed the physical capacity of the link, thus the packets are

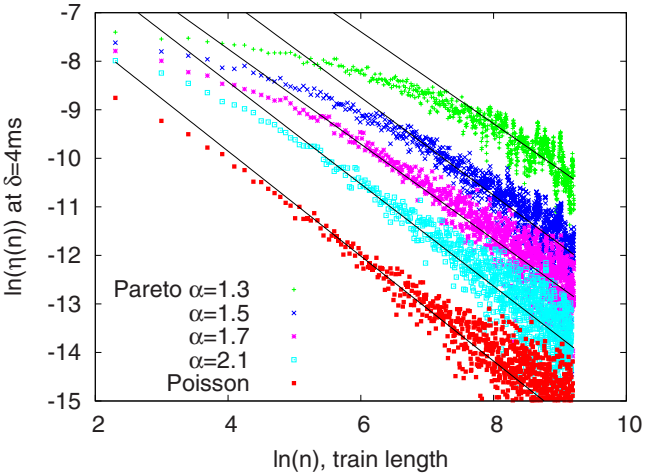

FIG. 6. (Color online) The deviation of the average output spacing from the fluid limit as a function of the train lengths for different arrival processes. The data sets are taken above the critical point, at $\delta_c=4$ ms of the dispersion curves.

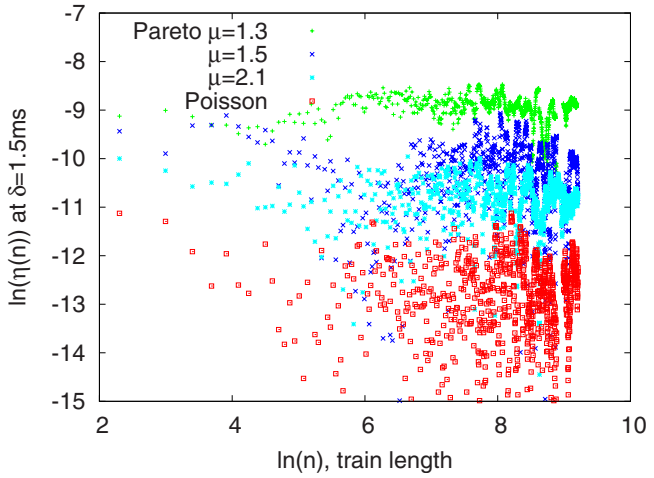

FIG. 7. (Color online) The deviation of the average output spacing from the fluid limit as a function of the train lengths for different arrival processes. The data sets are taken below the critical point, at $\delta=1.5$ ms of the dispersion curves.

leaving the queue in back-to-back manner. This means that the probe packets and the background traffic packets constitute a continuous flow without any gaps in between. Although the background traffic rate is fluctuating on short time scale, the traffic bursts can be neglected in our analysis. This is because even the large traffic bursts are built up by individual packets and thus the bursts not served as single traffic units, but as series of separate packets. In a general and realistic scenario the packet sizes are similar in the background traffic and the probe traffic. The order of the packets in the queue is determined by their order of arrival and as the packets of the probe train are arriving regularly, since their δ spacing is constant, the background traffic packets can only be injected in between. In the case of congestion this rule leads to an alternate packet order in the queue where every probe packet is followed by constant number of background traffic packets and vice versa. This phenomena can be seen in Fig. 5(d). This property eliminates the effect of high fluctuations of the background traffic and the background traffic jammed between the probe packets can be treated as a constant traffic amount. In this case, the average stretch then will be calculated from the total length of the probe train, in the following way

$$\begin{aligned}\eta(n) &= \delta_n' - \delta_{fl}' \\ &= \frac{1}{n-1}\sum_{i=1}^{n-1}\delta_i' - \delta_{fl}' \\ &= \frac{1}{n-1}\sum_{j=1}^{n-1}\left(\frac{p}{C}+\frac{p_{Bj}}{C}\right) - \frac{p}{C} - \delta\frac{B}{C} \\ &= \frac{1}{n-1}\sum_{j=1}^{n-1}\frac{p_{Bj}}{C} - \delta\frac{B}{C} \\ &= \frac{1}{n-1}\frac{(n-1)\cdot p_B}{C} - \delta\frac{B}{C} \\ &= \frac{p_B}{C} - \delta\frac{B}{C} \approx K \sim n^0, \end{aligned} \tag{15}$$

where δ_{fl}' is substituted from Eq. (7) and we can approximate $p_{Bj}=p_B$ due to the previously described property of the congested traffic. Finally, we can see that the average stretch in independent from the n probe train length.

In Fig. 7, the results from packet level simulations can be seen. We plotted the $\eta(n)$ average stretch as the function of the n probe train length. We investigated this relation by using several kinds of background traffic arrival processes with different parameters. One can see that the $\eta(n)$ values are independent of the actual choice of n probe train length for all the different arrival processes, which confirms the above results.

For the regions of δ input spacing values the $\eta(n)$ average stretch can be written

$$\eta(n) \sim \begin{cases} n^0 & \delta < \delta_c \\ n^\alpha & \delta = \delta_c \\ n^{-1} & \delta > \delta_c \end{cases}, \tag{16}$$

where the appropriate values for α is described by Eq. (12) formula. One can see that the α parameter of the background traffic arrival process plays an important role only in the critical point of the phase transition and its value can be estimated by the slope of the $\eta(n)$ function.

VII. CONCLUSION

Based on the packet-train measurement technique, we presented an approach to infer internal properties of the Internet traffic. We presented the packet train stretch as an order parameter, which describes the phase transition between the congested and uncongested phases of the bottleneck link. We studied the scaling phenomena at the critical point and we found that the exponent of the power law function arising in the average stretch function is closely related to the background traffic arrival process. We showed that the observed exponents can be calculated and the theoretical results match very well to the observed power law functions. It makes us possible to infer the qualitative and some of the quantitative parameters of the background traffic arrival process with end-to-end packet train technique.

As a future work we are planning to infer the traffic arrival process in wide geographic area in Europe with the ETOMIC infrastructure [31]. With this measurement infrastructure we will be able to collect large number of estimates frequently, which will helps us in the study of the spatiotemporal structure of the background traffic arrival processes.

ACKNOWLEDGMENTS

The authors thank the partial support of the National Office for Research and Technology (Grant No. NAP 2005/KCKHA005), the EU ICT MOMENT Collaborative Project (Grant Agreement No. 215225), the EU ICT OneLab2 Integrated Project (Grant agreement No. 224263) and the National Science Foundation OTKA 7779.

[1] R. Pastor-Satorras and A. Vespignani, *Evolution and Structures of the Internet: A Statistical Physics Approach* (Cambridge University Press, Cambridge, U.K., 2004).
[2] S.-H. Yook, H. Jeong, and A.-L. Barabási, Proc. Natl. Acad. Sci. U.S.A. **99**, 13382 (2002).
[3] P. Echenique, J. Gomez-Gardenes, and Y. Moreno, Phys. Rev. E **70**, 056105 (2004).
[4] D. De Martino, L. Dall'Asta, G. Bianconi, and M. Marsili, Phys. Rev. E **79**, 015101(R) (2009).
[5] R. D. Smith, e-print arXiv:0807.3374.
[6] S. Gábor and I. Csabai, Physica A **307**, 516 (2002).
[7] G. Mukherjee and S. S. Manna, Phys. Rev. E **71**, 066108 (2005).
[8] P. Echenique, J. Gomez-Gardenes, and Y. Moreno, Europhys. Lett. **71**, 325 (2005).
[9] F. Liu, X. Shan, Y. Ren, and J. Zhang, Physica A **328**, 341 (2003).
[10] I. Csabai, J. Phys. A **27**, L417 (1994).
[11] M. Takayasu, H. Takayasu, and T. Sato, Physica A **233**, 824 (1996).
[12] A. Feldmann, A. C. Gilbert, W. Willinger, and T. G. Kurtz, ACM SIGCOMM Comput. Commun. Rev. **28**, 5 (1998).
[13] W. E. Leland, M. S. Taqqu, W. Willinger and D. V. Wilson, IEEE/ACM Trans. Netw. **2**, 1 (1994).
[14] K. Park, G. Kim and M. Crovella, in *Proceedings of the 1996 International Conference on Network Protocols (INCP '96)* (IEEE Computer Society, Washington, DC, 1996), p.171.
[15] M. Crovella and A. Bestavros, IEEE/ACM Trans. Netw. **5**, 171 (1996).
[16] R. G. Clegg and M. M. Dodson, Phys. Rev. E **72**, 026118 (2005).
[17] L. Kocarev and G. Vattay, *Understanding Complex Systems* (Springer, New York, 2005).
[18] A. Erramilli, R. P. Singh, and P. Pruthi, Proceedings of the 14th ITC (unpublished).
[19] P. Hága, P. Pollner, G. Simon, I. Csabai, and G. Vattay, Nonlinear Phenomena in Complex Systems **6**, 814 (2003).
[20] A. Veres, Zs. Kenesi, S. Molnar, and G. Vattay, Comput. Commun. Rev. **30**, 243 (2000).
[21] A. Fekete and G. Vattay, Globecom, 2001.
[22] L. Guo, M. Crovella, and I. Matta, Tech. Rep. BUCS-TR-2000–017, Computer Science Dep., Boston University, 2000.
[23] Mark Crovella and Balachander Krishnamurthy, (unpublished).
[24] Passive Measurement and Analysis, National Laboratory for Applied Network Research.
[25] The MOME Project Consortium, http://www.ist-mome.org/.
[26] K. Cho, K. Mitsuya and A. Kato, USENIX 2000, San Diego, CA, June 2000.
[27] J. Duch and A. Arenas, Phys. Rev. Lett. **96**, 218702 (2006).
[28] S. Meloni, J. Gómez-Gardeñes, V. Latora, and Y. Moreno, Phys. Rev. Lett. **100**, 208701 (2008).
[29] J. Strauss, D. Katabi, and F. Kaashoek, ACM IMC—Internet Measurement Conference, (unpublished).
[30] M. Barthelemy, B. Gondran, and E. Guichard, Physica A **319**, 633 (2003).
[31] D. Morato *et al.*, IEEE TRIDENTCOM 2005, p. 283–289, Trento, Italy, Feb. 23–25 2005.
[32] A. Vázquez, R. Pastor-Satorras, and A. Vespignani, Phys. Rev. E **65**, 066130 (2002).
[33] Cooperative Association for Internet Data Analysis (CAIDA), http://www.caida.org/Tools.
[34] P. Hága, K. Diriczi, G. Vattay, and I. Csabai, Comput. Netw. **51**, 683 (2007).
[35] A. B. Downey, ACM SIGCOMM, 1999.
[36] R. L. Carter and M. E. Crovella, Perform. Eval. **27-28**, 297 1996.
[37] K. Lai and M. Baker, ACM SIGCOMM 2000, Stockholm, Sweden, 2000.
[38] J.-C. Bolot, Journal of High-Speed Networks **2**, 305 (1993).
[39] X. Liu, K. Ravindran, B. Liu and D. Loguinov, ACM Internet Measurement Conference,15, 918 (2007)..
[40] A. T. Lawniczak and X. Tang, Eur. Phys. J. B **50**, 231 (2006).