Of Mice and Models

Marco Ajmone Marsan, Giovanna Carofiglio, Michele Garetto, Paolo Giaccone, Emilio Leonardi, Enrico Schiattarella, and Alessandro Tarello

Dipartimento di Elettronica, Politecnico di Torino, C.so Duca degli Abruzzi 24, Torino, Italy*
firstname.lastname@polito.it

Abstract. Modeling mice in an effective and scalable manner is one of the main challenges in the performance evaluation of IP networks. Mice is the name that has become customary to identify short-lived TCP connections, that form the vast majority of packet flows over the Internet. On the contrary, long-lived TCP flows, that are far less numerous, but comprise many more packets, are often called elephants. Fluid models were recently proved to be a promising effective and scalable approach to investigate the dynamics of IP networks loaded by elephants. In this paper we extend fluid models in such a way that IP networks loaded by traffic mixes comprising both mice and elephants can be studied. We then show that the newly proposed class of fluid models is quite effective in the analysis of networks loaded by mice only, since this traffic is much more critical than a mix of mice and elephants.

1 Introduction

The traffic on the Internet can be described either at the packet level, modeling the dynamics of the packet generation and transmission processes, or at the flow level, modeling the start times and durations of sequences of packet transfers that correspond to (portions of) a service requested by an end-user. Examples of flows can be either the train of packets corresponding to an Internet telephone call, or the one corresponding to the download of a web page. In the latter case, the flow can be mapped onto a TCP connection. This is the most common case today in the Internet, accounting for the vast majority of traffic. The number of packets in TCP connections is known to exhibit a heavy-tailed distribution, with a large number of very small instances (called TCP mice) and few very large ones (called elephants).

Models for the performance analysis of the Internet have been traditionally based on a packet-level approach for the description of Internet traffic and of queuing dynamics at router buffers. Packet-level models allow a very precise description of the Internet operations, but suffer severe scalability problems, such that only small portions of real networks can be studied.

Fluid models have been recently proposed as a *scalable* approach to describe the behavior of the Internet. Scalability is achieved by describing the network and traffic dynamics at a higher level of abstraction with respect to traditional discrete packet-level models. This implies that the short-term random effects typical of the packet-level network behavior are neglected, focusing instead on the longer-term deterministic flow-level traffic dynamics.

* This work was supported by the Italian Ministry for Education, University and Research, through the FIRB project TANGO.

M. Ajmone Marsan et al. (Eds.): QoS-IP 2005, LNCS 3375, pp. 15–32, 2005.

In fluid flow models, during their activity period, traffic sources emit a continuous information stream, the fluid flow, which is transferred through a network of fluid queues toward its destination (also called sink).

The dynamics of a fluid model, which are continuous in both space and time, are naturally described by a set of ordinary differential equations, because of their intrinsic deterministic nature.

Fluid models were first proposed in [1–4] to study the interaction between TCP elephants and a RED buffer in a packet network consisting of just one bottleneck link, either ignoring TCP mice [1–3], or modeling them as unresponsive flows [4] introducing a stochastic disturbance. In this case, fluid models offer a viable alternative to packet-based simulators, since their complexity (i.e., the number of differential equations to be solved) is independent of both the number of TCP flows and the link capacity. In Section 2 we briefly summarize the fluid models proposed in [1–3], which constitute the starting point for our work.

Structural properties of the fluid model solution were analyzed in [5], while important asymptotic properties were proved in [6, 7]. In the latter works, it was shown that fluid models correctly describe the limiting behavior of the network when both the number of TCP elephants and the bottleneck link capacity jointly tend to infinity.

The single bottleneck model was then extended to consider general multi-bottleneck topologies comprising RED routers in [3, 8].

In all cases, the set of ordinary differential equations of the fluid model are solved numerically, using standard discretization techniques.

An alternative fluid model was proposed in [9, 10] to describe the dynamics of the average window for TCP elephants traversing a network of drop-tail routers. The behavior of such a network is pulsing: congestion epochs, in which some buffers are overloaded (and overflow), are interleaved to periods of time in which no buffer is overloaded and no loss is experienced, due to the fact that previous losses forced TCP sources to reduce their sending rate. In such a setup, a careful analysis of the average TCP window dynamics at congestion epochs is necessary, whereas sources can be simply assumed to increase their rate at constant speed between congestion epochs. This behavior allows the development of differential equations and an efficient methodology to solve them. Ingenious queueing theory arguments are exploited to evaluate the loss probability during congestion epochs, and to study the synchronization effect among sources sharing the same bottleneck link. Also in this case, the complexity of the fluid model analysis is independent of both the link capacities and the number of TCP flows.

Extensions that allow TCP mice to be considered are outlined in [9, 10] and in [8]. In this case, since the dynamics of TCP mice with different size and/or different start times are different, each mouse must be described with two differential equations; one representing the average window evolution, and one describing the workload evolution. As a consequence, one of the nicest properties of fluid models, the insensitivity of complexity with respect to the number of TCP flows, is lost.

In [11] a different description of the dynamics of traffic sources is proposed, that exploits *partial* differential equations to analyze the asymptotic behavior of a large number of TCP elephants through a single bottleneck link fed by a RED buffer.

In [12] we built on the approach in [11], showing that the partial differential equation description of the source dynamics allows the natural representation of mice as well as elephants, with no sacrifice in the scalability of the model.

The limiting case of an infinite number of TCP mice is considered in [13], where it is proved that, even in the case of loads lower than 1, deterministic synchronization effects may lead to congestion and to packet losses.

When the network workload is composed of a finite number of TCP mice, normally the link loads (given by the product of the mice arrival rate times the average mice length) are well below link capacities. In this case, the deterministic nature of fluid models leads to predict that buffers are always empty, and this fact contradicts the observations made on real packet networks. This discrepancy is due to the fact that, in underload conditions, the stochastic nature of the input traffic plays a fundamental role in the network dynamics, which cannot be captured by the determinism of fluid models.

In [12] we first discussed the possibility of integrating stochastic aspects within fluid models. We proposed a preliminary solution to the problem, exploiting a hybrid fluid-Montecarlo approach. In [14] we further investigate the possibilities for the integration of randomness in fluid models, proposing two additional approaches, which rely on second-order Gaussian approximations of the stochastic processes driving the network behavior.

In this paper, we consider the hybrid fluid-Montecarlo approach proposed in [12], further investigating the impact that different modeling choices can have in different dynamic scenarios. We present numerical results to show that the hybrid fluid-Montecarlo approach is capable of producing reliable performance predictions for networks that operate far from saturation. In addition, we prove the accuracy and the flexibility of the modeling approach by considering both static traffic patterns, from which equilibrium behaviors can be studied, and dynamic traffic conditions, that allow the investigation of transient dynamics.

2 Fluid Models of IP Networks

In this section we briefly summarize the fluid model presented in [1–3] and the extension presented in [12].

Consider a network comprising K router output interfaces, equipped with FIFO buffers, feeding links at rate C (the extension to non-homogeneous data rates is straightforward). The network is fed by I classes of TCP elephants; all the elephants within the same class follow the same route through the network, thus experiencing the same round-trip time (RTT), and the same average loss probability. At time $t = 0$ all buffers are assumed to be empty. Buffers drop packets according to their instant occupancy, as in drop tail buffers, or their average occupancy, as in RED (Random Early Detection [15]) active queue management (AQM) schemes.

2.1 Elephant Evolution Equations

In [1–3], simple differential equations were developed to describe the behavior of TCP elephants over networks of IP routers adopting a RED AQM scheme. We refer to this original model with the name MGT.

Consider the ith class of elephants; the temporal evolution of the average window of TCP sources in the class, $W_i(t)$, is described by the following differential equation:

$$\frac{\mathrm{d}W_i(t)}{\mathrm{d}t} = \frac{1}{R_i(t)} - \frac{W_i(t)}{2}\,\lambda_i(t) \tag{1}$$

where $R_i(t)$ is the average RTT for class i, and $\lambda_i(t)$ is the loss indicator rate experienced by TCP flows of class i. The differential equation is obtained by considering the fact that elephants can be assumed to be always in congestion avoidance (CA) mode, so that the window dynamics are close to AIMD (Additive Increase, Multiplicative Decrease). The window increase rate in CA mode is linear, and corresponds to one packet per RTT. The window decrease rate is proportional to the rate with which congestion indications are received by the source, and each congestion indication implies a reduction of the window by a factor two.

In [12] we extended the fluid model presented in [1–3]. In our approach, that will be named PDFM, rather than just describing the average TCP connection behavior, we try to statistically model the dynamics of the entire population of TCP flows sharing the same path. This approach leads to systems of partial derivatives differential equations, and produces more flexible models, which scale independently from the number of TCP flows.

To begin, consider a fixed number of TCP elephants. We use $P_i(w,t)$ to indicate the number of elephants of class i whose window is $\leq w$ at time t. For the sake of simplicity, we consider just one class of flows, and omit the index i from all variables. The source dynamics are described by the following equation, for $w \geq 1$:

$$\frac{\partial P(w,t)}{\partial t} = \int_w^{2w} \lambda(\alpha,t)\frac{\partial P(\alpha,t)}{\partial \alpha}\,\mathrm{d}\alpha - \frac{1}{R(t)}\frac{\partial P(w,t)}{\partial w} \tag{2}$$

where $\lambda(w,t)$ is the loss indication rate. The intuitive explanation of the formula is the following. The time evolution of the population described by $P(w,t)$ is governed by two terms: i) the integral accounts for the growth rate of $P(w,t)$ due to the sources with window between w and $2w$ that experience losses; ii) the second term describes the decrease rate of $P(w,t)$ due to sources increasing their window with rate $1/R(t)$.

2.2 Network Evolution Equations

In both models, $Q_k(t)$ denotes the (fluid) level of the packet queue in the kth buffer at time t; the temporal evolution of the queue level is described by:

$$\frac{\mathrm{d}Q_k(t)}{\mathrm{d}t} = A_k(t)\,[1 - p_k(t)] - D_k(t) \tag{3}$$

where $A_k(t)$ represents the fluid arrival rate at the buffer, $D_k(t)$ the departure rate from the buffer (which equals C_k, provided that $Q_k(t) > 0$), and the function $p_k(t)$ represents the instantaneous loss probability at the buffer, which depends on the packet discard policy at the buffer. An explicit expression for $p_k(t)$ is given in [2] for RED buffers, while for drop-tail buffers:

$$p_k(t) = \frac{\max(0, A_k(t) - C)}{A_k(t)}\,\mathbb{1}_{\{Q_k(t)=B_k\}} \tag{4}$$

If $T_k(t)$ denotes the instantaneous delay of buffer k at time t, we can write:

$$T_k(t) = Q_k(t)/C_k$$

If $\mathcal{F}_k$ indicates the set of elephants traversing buffer k, $A_k^i(t)$ and $D_k^i(t)$ are respectively the arrival and departure rates at buffer k referred to elephants in class i, so that:

$$A_k(t) = \sum_{i \in \mathcal{F}_k} A_k^i(t)$$

$$\int_0^{t+T_k(t)} D_k(a)\,\mathrm{d}a = \int_0^t A_k(a)\,[1 - p_k(t)]\,\mathrm{d}a$$

$$\int_0^{t+T_k(t)} D_k^i(a)\,\mathrm{d}a = \int_0^t A_k^i(a)\,[1 - p_k(t)]\,\mathrm{d}a$$

which means that the total amount of fluid arrived up to time t at the buffer leaves the buffer by time $t + T_k(t)$, since the buffer is FIFO. By differentiating the last equation:

$$D_k^i(t + T_k(t)) \left(1 + \frac{\mathrm{d}T_k(t)}{\mathrm{d}t}\right) = A_k^i(t)\,[1 - p_k(t)]$$

2.3 Source-Network Interactions

Consider the elephants of class i. Let $k(h,i)$ be the h-th buffer traversed by them along their path P_i of length H^i. The RTT $R_i(t)$ perceived by elephants of class i satisfies the following expression:

$$R_i\left(t + g_i + \sum_{h=1}^{H^i} T_{k(h,i)}(t_{k(h,i)})\right) = g_i + \sum_{h=1}^{H^i} T_{k(h,i)}(t_{k(h,i)}) \tag{5}$$

where g_i is the total propagation delay[1] experienced by elephants in class i, and $t_{k(h,i)}$ is the time when the fluid injected at time t by the TCP source reaches the h-th buffer along its path P_i. We have:

$$t_{k(h,i)} = t_{k(h-1,i)} + T_{k(h-1,i)}(t_{k(h-1,i)}) \tag{6}$$

Now the instantaneous loss probability experienced by elephants in class i, $p_i^F(t)$, is given by:

$$p_i^F(t) = 1 - \prod_{h=1}^{H^i} \left[1 - p_{k(h,i)}(t_{k(h,i)})\right]$$

[1] Equation (5) comprises the propagation delay g_i in a single term, as if it were concentrated only at the last hop. This is just for the sake of easier reading, since the inclusion of the propagation delay of each hop would introduce just a formal modification in the recursive equation of $t_{k(h,i)}$.

By omitting the class index from the notation, the loss rate indicator $\lambda(w,t)$ in our PDFM model can be computed as follows:

$$\lambda\left(w, t+R(t)\right) = \frac{w\, p^F(t)}{R(t)} \tag{7}$$

where $w/R(t)$ is the instantaneous emission rate of TCP sources, and the source window at time $t+R(t)$ is used to approximate the window at time t. Intuitively, this loss model distributes the lost fluid over the entire population, proportionally to the window size.

Finally, in both models:

$$A_k(t) = \sum_i \sum_q r^i_{qk} D^i_q(t) + \sum_i e^i_k \frac{W_i(t)}{R_i(t)} N_i \tag{8}$$

where $e^i_k = 1$ if buffer k is the first buffer traversed by elephants of class i, and 0 otherwise; r^i_{qk} is derived by the routing matrix, being $r^i_{qk} = 1$ if buffer k immediately follows buffer q along P_i.

2.4 Mice

The extension of fluid models to a finite population of mice was provided only in [12], thus, here we refer only to the PDFM model.

The model of TCP mice discussed in this section extends the PDFM model reported in [12], and takes into account the effects of the sources maximum window size W_{max}, of the TCP fast recovery mechanism which prevents from halving the window more than once in each round-trip time, of the initial slow-start phase up to the first loss, and of time-outs.

We assume flow lengths to be exponentially distributed, with average L. Thanks to the memoryless property of the exponential distribution, we can write:

$$\begin{aligned}\frac{\partial P_{nl}(w,t)}{\partial t} = &-\frac{1}{R(t)}\frac{\partial P_{nl}(w,t)}{\partial w} - \frac{1}{R(t)L}\int_1^w \alpha \frac{\partial P_{nl}(\alpha,t)}{\partial \alpha}\,\mathrm{d}\alpha \\ &- \int_1^w \lambda(\alpha,t)\frac{\partial P_{nl}(\alpha,t)}{\partial \alpha}\,\mathrm{d}\alpha + \frac{1}{R(t)}P_l(w,t)\end{aligned} \tag{9}$$

$$\begin{aligned}\frac{\partial P_l(w,t)}{\partial t} = &\int_1^{\min(2w,W_{max})} \lambda(\alpha,t) p_{NTO}(\alpha)\frac{\partial P_{nl}(\alpha,t)}{\partial \alpha}\,\mathrm{d}\alpha \\ &+ \int_1^{\min(2w,W_{max})} \lambda(\alpha,t) p_{NTO}(\alpha)\frac{\partial P_s(\alpha,t)}{\partial \alpha}\,\mathrm{d}\alpha \\ &+ \int_1^{W_{max}} \lambda(\alpha,t) p_{TO}(\alpha)\frac{\partial P_{nl}(\alpha,t)}{\partial \alpha}\,\mathrm{d}\alpha \\ &+ \int_1^{W_{max}} \lambda(\alpha,t) p_{TO}(\alpha)\frac{\partial P_s(\alpha,t)}{\partial \alpha}\,\mathrm{d}\alpha - \frac{1}{R(t)}P_l(w,t)\end{aligned} \tag{10}$$

$$\frac{\partial P_s(w,t)}{\partial t} = -\frac{w}{R(t)}\frac{\partial P_s(w,t)}{\partial w} - \frac{1}{R(t)L}\int_1^w \alpha \frac{\partial P_s(\alpha,t)}{\partial \alpha}\,\mathrm{d}\alpha - \int_1^w \lambda(\alpha,t)\frac{\partial P_s(\alpha,t)}{\partial \alpha}\,\mathrm{d}\alpha + \gamma(t) \quad (11)$$

where $P_{nl}(w,t)$ represents the window distribution of TCP sources in congestion avoidance mode which have not experienced losses in the last round-trip time; $P_l(w,t)$ represents the windows distribution of TCP sources in congestion avoidance mode which have experienced losses in the last round-trip time; $P_s(w,t)$ represents the window distribution of TCP sources in slow-start mode; and W_{max} represents the maximum window size. Finally, $p_{TO}(w) = 1 - p_{NTO}(w) = \min(1, 3/w)$ represents the probability that packet losses induce a time-out of TCP transmitters having window size w, as proposed in [16].

The left hand side of (9) represents the variation of the number of TCP sources in congestion avoidance which at time t have window less than w and have not experienced losses in the last round-trip time. The negative terms on the right hand side of the same equation indicate that this number decreases because of the window growth (first term), of connection completions (second term), and of losses (third term). The fourth positive term indicates an increase due to the elapsing of a round trip time after the last loss.

Similarly, the left hand side of (10) represents the variation of the number of TCP sources in congestion avoidance which at time t have window less than w and have experienced losses in the last round-trip time. The positive terms on the right hand side indicate that this number increases because of losses that either do not induce or do induce a timeout (respectively, first and second, and third and fourth terms) for connections that either were in slow-start mode (second and fourth terms) or had not experienced losses in the last round-trip time (first and third terms). The fifth negative term indicates a decrease due to the elapsing of a round trip time after the last loss.

Finally, the left hand side of (11) represents the variation of the number of TCP sources that, at time t, have window less than w and are in slow-start mode. The negative terms on the right hand side indicate that this number decreases because of the window growth (first term), of connection completions (second term), and of losses (third term). The fourth positive term indicates an increase due to the generation of new connections, that always open in slow-start mode with window size equal to 1.

We wish to stress the fact that (9)-(11) provide quite a powerful tool for an efficient representation of TCP mice, since a wide range of distributions (including those incorporating long range dependence) can be approximated with a good degree of accuracy by a mixture of exponential distributions [17].

3 The Role of Randomness

The fluid models presented so far provide a deterministic description of the network dynamics at the flow level, thus departing from the traditional approach of attempting a probabilistic description of the network at the packet level by means of stochastic models, such as continuous-time or discrete-time Markov chains and queueing models.

Table 1. Correspondence of input parameters and dynamic variables in the original fluid model and in the transformed model.

	Unchanged	Multiplied by η
Input parameters	g, L	N, B, C
	p_max	$\gamma(t,l), \gamma(t), min\_th, max\_th$
Dynamic variables	$R(t), \lambda(w,t), p(t), W(t)$	$Q(t), D(t), A(t), P(w,t), P_{max}(t)$
	$p^F(t), T(t), \bar{p}_L, \bar{w}(t)$	$P_O(w,t), P_L(w,t), P(w,t,l), P_s(w,t,l), P_s(w,t)$

Deterministic fluid models were proven to correctly represent the asymptotic behavior of IP networks when the number of active TCP elephants tends to infinity [7]. Indeed, when considering scenarios with only elephants, randomness, which is completely lost in fluid models, plays only a minor role, because links tend to be heavily congested, and the packet loss rate is determined by the load that TCP connections offer in excess of the link capacity.

Deterministic fluid models also exhibit nice invariant properties, as proven in [5, 18]. In particular, the whole set of equations describing our PDFM model is invariant under the linear transformation of parameters and variables summarized in Table 1. The top rows of Table 1 report the transformations which map the original network parameters into those of the transformed network, being $\eta \in I\!R^+$ the multiplicative factor applied to the model parameters. Basically, the transformed network is obtained from the original network by multiplying by a factor η the number of elephants and the arrival rate of mice γ, as well as all transmission capacities and buffer dimensions. Table 1 in the bottom rows also reports the transformations which relate the modeled behavior of the original network to that of the transformed network. These invariance properties are extremely interesting, since they suggest that the behavior of very large systems can be predicted by scaling down network parameters and studying small-scale systems. This result was confirmed by simulation experiments reported in [5, 18], in case of heavily congested links.

However, deterministic fluid models are not suitable for the study of network scenarios where link capacities are not saturated. In particular, fluid models fail to correctly predict the behavior of networks loaded by a finite number of TCP mice only. This fact can be explained by looking at a network with just a single bottleneck link of capacity C, loaded by TCP mice that open according to a stationary process with rate γ, and consist of a unidirectional transfer of data with arbitrary distribution and mean size L. The average utilization ρ of the bottleneck link can be expressed as

$$\rho = \frac{\gamma\, L}{C} \tag{12}$$

Clearly, ρ must be smaller than 1 in order to obtain a stable system behavior. Neglecting pathological behaviors which may appear for particular initial conditions, as shown in [13], for all values $\rho < 1$ a deterministic fluid model predicts that the buffer feeding the link is always empty, so that queueing delays and packet loss probabilities are equal to zero. This is due to the fact that deterministic models consider only the first moment of the packet arrival process at the buffer, $A(t)$, and of the opening process of TCP mice, $\gamma(t)$. By so doing, $A(t)$ remains below the link capacity C, and the buffer

remains empty. Since the loss probability is zero, no retransmissions are needed, and the actual traffic intensity on the link converges to the nominal link utilization ρ computed by (12). This prediction is very far from what is observed in either *ns-2* simulation experiments or measurement setups, that show how queuing delays and packet losses are non-negligible for a wide range of values $\rho < 1$. This prediction error is essentially due to the fact that, in underload conditions, randomness plays a fundamental role that cannot be neglected in the description of the network dynamics. Indeed, randomness impacts the system behavior at two different levels:

- **Randomness at Flow Level** is due to the stochastic nature of the arrival and completion processes of TCP mice. The number of active TCP mice in the network varies over time, and the offered load changes accordingly.
- **Randomness at Packet Level** is due to the stochastic nature of the arrival and departure processes of packets at buffers. In particular, the burstiness of TCP traffic is responsible for high queueing delays and sporadic buffer overloads, even when the average link utilization is much smaller than 1.

3.1 The Hybrid Fluid-Montecarlo Approach

The approach we propose to account for randomness consists in transforming the deterministic differential equations of the fluid model into stochastic differential equations, which are then solved using a Montecarlo approach.

More in detail, we consider two levels of randomness:

- **Randomness at Flow Level.** The deterministic mice arrival rate $\gamma(t)$ in (10) is replaced by a Poisson counter with average $\gamma(t)$. The deterministic mice completion process can be replaced by an inhomogeneous Poisson process whose average at time t is represented by the sum of the second terms on the right hand side of (9) and (11). As opposed to the mice arrival process, which is assumed to be exogenous, the mice completion process depends on the network congestion: when the packet loss probability increases, the rate at which mice leave the system is reduced.
- **Randomness at Packet Level.** The workload emitted by TCP sources, rather than being a continuous deterministic fluid process with rate $W_i(t)N_i/R_i(t)$, can be taken to be a stochastic point process with the same rate. Previous work in TCP modeling [19, 20] showed that an effective description of TCP traffic in networks with high bandwidth-delay product is obtained by means of a batched Poisson process in which the batch size is distributed according to the window size of TCP sources. The intuition behind this result is that, if the transmission time for all packets in a window is much smaller than the round trip time, packets transmissions are clustered at the beginning of each RTT. This introduces a high correlation in the inter-arrival times of packets at routers, that cannot be neglected, since it heavily impacts the buffer behavior. By grouping together all packets sent in a RTT as a single batch, we are able to capture most of the bursty nature of TCP traffic. This approach is very well suited to our model, which describes the window size distribution of TCP sources.

An important observation is that the addition of randomness in the model destroys the invariance properties described in Section 3. This can be explained very simply by considering that the loss probability predicted by any analytical model of a finite queue

with random arrivals (e.g., an M/M/1/B queue) depends on the buffer size, usually in a non linear way. Instead, according to the transformations of Table 1, the loss probability should remain the same after scaling the buffer size. In underload conditions, it is clearly wrong to assume that the packet loss probability does not change with variations in the buffer size.

4 Results for Mice

In this section we discuss results for network scenarios comprising TCP mice. First, we investigate the impact of the source emission model in a scenario where only mice are active. Second, we study the impact of the flow size distribution. Third, we investigate the invariance properties of the network when mice are present. Finally we study a dynamic (non-stationary) scenario in which a link is temporarily overloaded by a flash crowd of new TCP connection requests.

4.1 Impact of the Source Emission Model

Consider a single bottleneck link fed by a drop-tail buffer, with capacity equal to 1000 packets. The link data rate C is 1.0 Gbps, while the propagation delay between TCP sources and buffer is 30 ms. In order to reproduce a TCP traffic load close to what has been observed on the Internet, flow sizes are distributed according to a Pareto distribution with shape parameter equal to 1.2 and scale parameter equal to 4.

Using the algorithm proposed in [17], we approximated the Pareto distribution with a hyper-exponential distribution of the 9-th order, whose parameters are reported in Table 2. The resulting average flow length is 20.32 packets. Correspondingly, 9 classes of TCP mice are considered in our model. The maximum window size is set to 64 packets for all TCP sources. Experiments with loads equal to 0.6, 0.8 and 0.9 were run; however, for the sake of brevity, we report here only the results for load equal to 0.9.

Fig. 1 compares the queue lengths distributions obtained with *ns-2*, and with the stochastic fluid model. While in the model the flow arrival and completion processes have been randomized according to a non-homogeneous Poisson process (see Section 3.1), different approaches have been tried to model the traffic emitted by sources:

- *Poisson:* the emitted traffic is a Poisson process with time-varying rate;
- *Det-B:* the emitted traffic is a batch Poisson process with time-varying rate and constant batch size, equal to the instantaneous average TCP mice window size;
- *Exp-B:* the emitted traffic is a batch Poisson process with time-varying rate and exponential batch size, whose mean is equal to the instantaneous average TCP mice window size;
- *Win-B:* the emitted traffic is a batch Poisson process with time-varying rate, in which the batch size distribution is equal to the instantaneous TCP mice window size distribution.

The last three approaches were suggested by recent results about the close relationship existing between the burstiness of the traffic generated by mice and their window size [19].

Table 2. Parameters of the hyper-exponential distribution approximating the Pareto distribution.

Prob.	mean length
$7.88\ 10^{-1}$	6.48
$1.65\ 10^{-1}$	23.26
$3.70\ 10^{-2}$	80.65
$8.34\ 10^{-3}$	279.7
$1.87\ 10^{-3}$	970.2
$4.22\ 10^{-4}$	3376
$9.46\ 10^{-5}$	11862
$2.10\ 10^{-5}$	43086
$4.52\ 10^{-6}$	176198

If we use a Poisson process to model the instants in which packets (or, more precisely, units of fluid) are emitted by TCP sources, the results generated by the fluid model cannot match the results obtained with the *ns-2* simulator, as can be observed in Fig. 1. Instead, the performance predictions obtained with the fluid model become quite accurate when the workload emitted by TCP sources is taken to be a Poisson process with batch arrivals. The best fitting (confirmed also by several other experiments, not reported here for lack of space) is obtained for batch size distribution equal to the instantaneous TCP mice window size distribution (case *Win-B*). Note that our proposed class of fluid models naturally provides the information about the window size distribution, whereas the MGT model provides only the average window size.

Table 3 reports the average loss probability, the average queue length, and the average completion time for each class of TCP mice, obtained with *ns-2*, with the *Poisson* and the *Win-B* models. The *Poisson* model significantly underestimates the average queue length and loss probability, thus producing an optimistic prediction of completion times. The *Win-B* model moderately overestimates the average queue length and loss probability, as pointed out in [19]. However, for very short flows, completion time predictions obtained with the *Win-B* model are slightly optimistic; this is mainly due to the fact that an idealized TCP behavior (in particular, without timeouts) is considered in the model.

4.2 Impact of the Flow Size

We now discuss the ability of our model to capture the impact on the network behavior of the flow size variance.

We consider three different scenarios, in which flow lengths are distributed according to either an exponential distribution ("Distr.1"), or hyper-exponentials of the second order ("Distr.2" and "Distr.3"). For all three scenarios, we keep the average flow size equal to 20.32 (this is the average flow size used in the previous subsection), and we vary the standard deviation σ. Detailed parameters of our experiments are reported in Table 4.

Table 5 shows a comparison between the results obtained with the *Win-B* model and with *ns-2*. As in previous experiments, the model moderately overestimates both the average loss probability and the average queue length. The discrepancies in the average completion times between model and *ns-2* remain within 10%.

Fig. 2, which reports the queue length distributions obtained by the model in the three scenarios, emphasizes the significant dependency of the queue behavior on the flow size variance. This dependency is mainly due to the complex interactions between the packet level and the flow level dynamics which are due to the TCP protocol.

4.3 Impact of the Link Capacity

Finally, we discuss the effect on performance of the link capacity. The objective of this last study of networks loaded with TCP mice only is to verify whether the performance of networks which differ for a multiplicative factor in capacities show some type of invariance, like in the case of elephants.

More precisely, we wish to determine whether the queue length distribution exhibits any insensitivity with respect to the bottleneck link capacity, for the same value of the traffic intensity. This curiosity is motivated by the fact that in many classical queuing models (e.g. the M/M/1 queue, possibly with batch arrivals) the queue length distribution depends only on the average load, not on the server speed.

We consider the third scenario ("Distr.3") of the previous experiment, we fix the traffic load at 0.9, and we study four different networks, in which the bottleneck capacity is equal to 10 Mbps, 100 Mbps, 1 Gbps, 10 Gbps, respectively.

The results of the fluid model, depicted in Fig. 3, show that, in general, the queue length distribution exhibits a dependency on the link capacity. The packet level behavior, indeed, strongly depends on flow level dynamics, which cause a slowly varying modulation of the arrival rate at the packet level. The flow level dynamics, however, do not scale up with the capacity of the system, since the random variable which represent the number of active flows has a coefficient of variation which decreases as we increase the system capacity (consider, for example, the Poisson distribution of the number of active flows proposed in [17]).

Nevertheless, when the capacity of the system becomes very large (in the considered example, greater than 1 Gbps) the dependence of the queue distribution on capacity tends to vanish, and the queueing behavior becomes indeed independent from the link capacity. This phenomenon was confirmed by *ns-2* simulations.

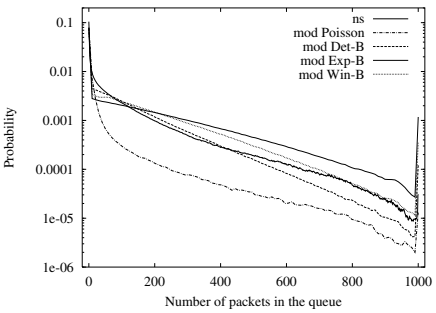

Fig. 1. Queue length distribution for single drop tail bottleneck, varying the random process modeling the workload emitted by the TCP sources; comparison with *ns-2* simulator.

Table 3. Average loss probability (ALP), average queue length (AQL) and average completion times (ACT) in seconds of the nine classes of mice for the setup of Section 4.1.

	ALP	AQL	ACT[s]
Poisson	$1.23\ 10^{-6}$	17.61	0.0932, 0.129, 0.169 0.274, 0.613, 1.68 5.48, 19.2, 73.7
win Poisson	$1.22\ 10^{-4}$	143.12	0.0991, 0.138, 0.187 0.297, 0.658, 1.92 6.29, 22.4, 95.1
ns-2	$5.34\ 10^{-5}$	101.63	0.104, 0.160, 0.219 0.327, 0.661, 1.83 6.10, 21.3, 87.0

Table 4. Parameters of the three flow length distributions.

	σ	mean length 1	mean length 2
Distr. 1	20.32	20.32	-
Distr. 2	28.89	6.48	80.65
Distr. 3	215.51	6.48	3376.24

This behavior is mainly due to the fact that when the capacity becomes very large, the coefficient of variation of the number of active flows becomes small. As a consequence, the effects of the flow level dynamics on the network performance tend to become negligible, and the packet-level behavior resembles that of a single server queue loaded by a stationary Poisson (or batched Poisson) process, for which the queue length distribution is independent of the server capacity.

To confirm this intuition, we solved the fluid model by eliminating the randomness at the flow level (i.e., in the flow arrival and departure processes), and we observed that the dependency on the capacity disappears.

We would like to remark, however, that the flow length distribution plays a major role in determining the system capacity above which the queue length distribution no longer depends on the system capacity – the invariance phenomenon appears at higher data rates when the variance of the flow length distribution increases.

4.4 Flash Crowd Scenario

We now study a non-stationary traffic scenario, in which a network section is temporarily overloaded by a flash crowd of new TCP connection requests.

Consider a series of two links at 100 Mb/s, as shown in Figure 4. The first link is fed by a drop-tail buffer, while the second link is fed by a RED buffer. Both buffers can store up to 320 packets. As shown in Figure 4, three classes of TCP mice traverse this network section. Mice of class 1 traverse only the first link, mice of class 2 traverse only the second link, and mice of class 3 traverse both links. The length of mice in all three classes is geometrically distributed with mean 50 packets. Mice of classes 1 and 3 offer a stationary load equal to 45 and 50 Mb/s, respectively. Mice of class 2 offer a

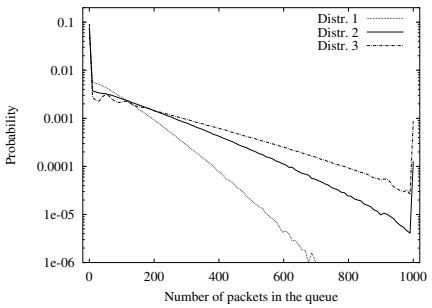

Fig. 2. Queue size distribution for single drop tail bottleneck, varying the flow length distribution.

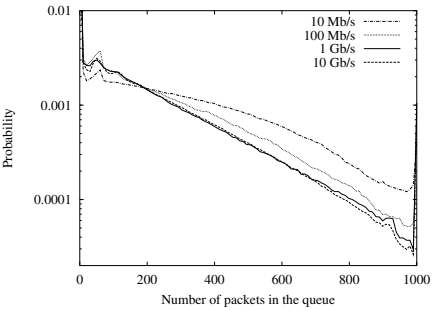

Fig. 3. Queue size distribution for single drop tail bottleneck, varying the bottleneck capacity.

time-varying load that follows the profile shown in Figure 5, and determine a temporary overload of the second queue between $t_1 = 50$ s and $t_2 = 95$ s.

Figures 6 and 8 report, respectively, the queue lengths and the source window sizes versus time, averaging values obtained during intervals of duration 1 s for a better representation. For the sake of comparison, Figures 7 and 9 report the same performance indices obtained with *ns-2*. First of all, it is necessary to emphasize the fact that the reported curves refer to sample paths of the stochastic processes corresponding to the network dynamics, not to averages or moments of higher order; thus, the curves are the result of particular values taken by the random variables at play, and the comparison between the *ns-2* and the model results requires a great amount of care. In spite of this fact, we can observe a fairly good agreement between the *ns-2* and the model predictions. For example, in both cases it is possible to see that congestion arises at the second queue as soon as the offered load approaches 1, i.e., around $t = 50$ s. After a while, congestion propagates back to the first queue, due to retransmissions of mice of class 3, which cross both queues. At time $t = 100$ s, when the offered load at the second buffer decreases below 1, congestion ends at the second queue, but persists until $t = 160$ s at the first queue, due to the large number of active mice of class 3.

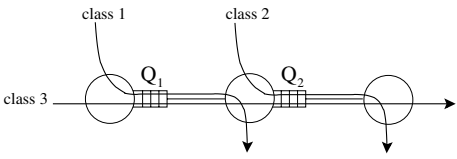

Fig. 4. Network topology of the flash crowd scenario.

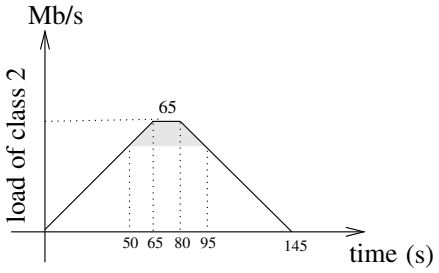

Fig. 5. Temporal evolution of the load produced by the flash crowd.

Table 5. Average loss probability (ALP), average queue length (AQL) and average completion times (ACT) in seconds of the different classes of mice for the setup of Section 4.2, having introduced random elements.

	ALP	AQL	ACT[s]
Distr. 1 (model)	0.0	63.9	0.131
Distr. 1 (ns)	0.0	40.9	0.128
Distr. 2 (model)	$4.01\ 10^{-5}$	123	0.0985, 0.185
Distr. 2 (ns)	$9.00\ 10^{-6}$	98.0	0.0878, 0.191
Distr. 3 (model)	$3.29\ 10^{-4}$	167	0.0999, 2.01
Distr.3 (ns)	$1.23\ 10^{-4}$	142	0.0915, 1.85

For the sake of comparison, in this case we also show in Figure 10 the predictions generated by the fully deterministic fluid model. The fact that in this case the network becomes overloaded allows the deterministic fluid model to correctly predict congestion. However, the deterministic fluid model predicts a behavior at the first queue after the end of the overload period which is different from *ns-2* observations. Indeed, the fluid model predicts a persistent congestion for the first queue, which gets trapped into a "spurious" equilibrium point. This phenomenon is not completely surprising, and confirms the results presented in [13], where it was recently proved that asymptotic mean

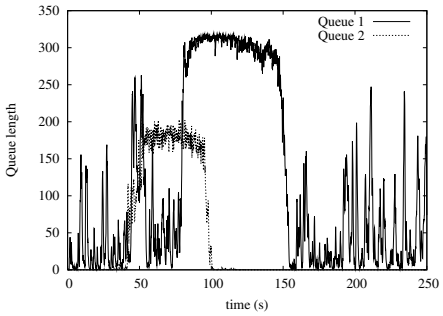

Fig. 6. Queue size evolution obtained with the Fluid-Montecarlo approach.

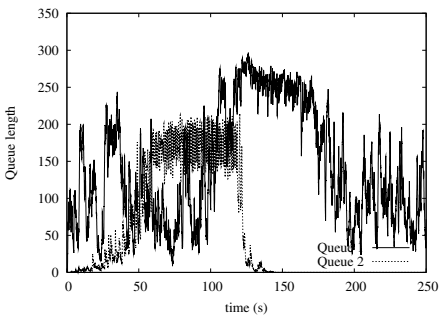

Fig. 7. Queue size evolution obtained with *ns-2*.

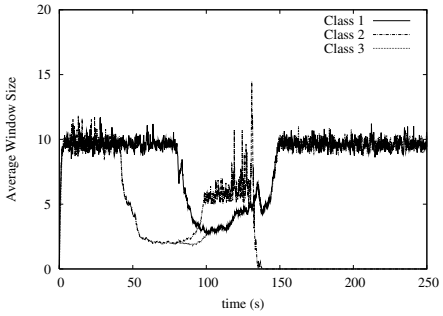

Fig. 8. Window size evolution obtained with the Fluid-Montecarlo approach.

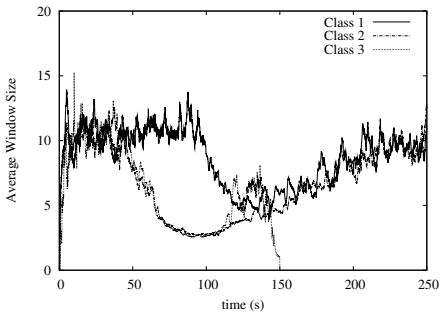

Fig. 9. Window size evolution obtained with *ns*.

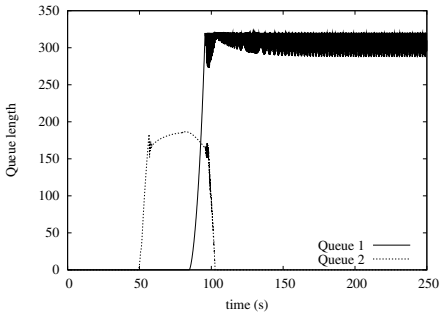

Fig. 10. Queue size evolution obtained with the deterministic fluid model.

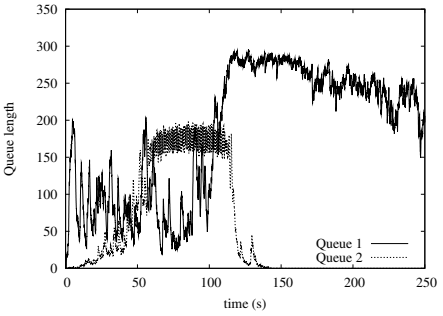

Fig. 11. *ns-2* sample path showing persistent congestion at queue 1.

field models of networks loaded by a large number of short-lived TCP flows may exhibit permanent congestion behaviors also when the load is less than 1. The existence of two equilibrium points at loads close to 1, one stable and the other unstable, was also described in [21]. This pseudo-chaotic dynamics of deterministic fluid models reflects a behavior that can be observed in *ns-2* simulations. See for example the *ns-2* sample path reported in Figure 11, for the same scenario. In this case congestion at the first queue persists for a very long time, until random fluctuations bring down the system to the stable operating point.

5 Conclusions

In this paper we have defined a class of fluid models that allows reliable performance predictions to be computed for large IP networks loaded by TCP mice and elephants, and we have proved the accuracy and the flexibility of such models under static and dynamic traffic scenarios comprising just mice.

The choice of traffic patterns comprising just mice in the model validation is due to the fact that the effectiveness of fluid models in the performance analysis of IP networks loaded by just elephants or by mixes of elephants and mice was already proved in previous works. The key characteristic of traffic patterns comprising elephants that makes them suitable to a fluid analysis lies in the fact that elephants (being modeled as infinite-length flows) bring the utilization of (some) network elements close to saturation, so that performance can be studied with a set of deterministic differential equations defining the fluid model of the system.

On the contrary, traffic patterns comprising just mice, induce a fixed load on the network elements, which normally remain well below their saturation point. This is what actually happens in the Internet today (because elephants may be large, possibly huge, but cannot be of infinite size), so that defining fluid models that can cope with this situation is of great relevance.

The fact that network elements are far from saturation, makes the standard deterministic fluid models useless for performance analysis, because they predict all buffers to be deterministically empty, and all queuing delays to be deterministically zero. This

is not what we experience and measure from the Internet. The reason for this discrepancy lies in the stochastic nature of traffic, which is present on the Internet, but is not reflected by the standard deterministic fluid models.

The fluid model paradigm that we defined in this paper is capable of accounting for randomness in traffic, at both the flow and the packet levels, and can thus produce reliable performance predictions for networks that operate far from saturation.

The accuracy and the flexibility of the modeling paradigm was proved by considering both static traffic patterns, from which equilibrium behaviors can be studied, and dynamic traffic conditions, that allow the investigation of transient dynamics.

References

1. V.Misra, W.Gong, D.Towsley, "Stochastic Differential Equation Modeling and Analysis of TCP Window Size Behavior", *Performance'99*, Istanbul, Turkey, October 1999.
2. V.Misra, W.B.Gong, D. Towsley, "Fluid-Based Analysis of a Network of AQM Routers Supporting TCP Flows with an Application to RED", *ACM SIGCOMM 2000,* Stockholm, Sweden, August 2000.
3. Y.Liu, F.Lo Presti, V.Misra, D.Towsley, "Fluid Models and Solutions for Large-Scale IP Networks", *ACM SIGMETRICS 2003,* San Diego, CA, USA, June 2003.
4. C.V.Hollot, Y.Liu, V.Misra and D.Towsley, "Unresponsive Flows and AQM Performance," *IEEE Infocom 2003*, San Francisco, CA, USA, March 2003.
5. R.Pan, B.Prabhakar, K.Psounis, D.Wischik, "SHRiNK: A Method for Scalable Performance Prediction and Efficient Network Simulation", *IEEE Infocom 2003*, San Francisco, CA, USA, March 2003.
6. S.Deb, S.Shakkottai, R.Srikant, "Stability and Convergence of TCP-like Congestion Controllers in a Many-Flows Regime", *IEEE Infocom 2003*, San Francisco, CA, USA, March 2003.
7. P.Tinnakornsrisuphap, A.Makowski, "Limit Behavior of ECN/RED Gateways Under a Large Number of TCP Flows", *IEEE Infocom 2003*, San Francisco, CA, USA, March 2003.
8. M.Barbera, A.Lombardo, G.Schembra, "A Fluid-Model of Time-Limited TCP flows", to appear on *Computer Networks*.
9. F.Baccelli, D.Hong, "Interaction of TCP Flows as Billiards", *IEEE Infocom 2003*, San Francisco, CA, USA, March 2003.
10. F.Baccelli, D.Hong, "Flow Level Simulation of Large IP Networks", *IEEE Infocom 2003*, San Francisco, CA, USA, March 2003.
11. F.Baccelli, D.R.McDonald, J.Reynier, "A Mean-Field Model for Multiple TCP Connections through a Buffer Implementing RED", *Performance Evaluation,* vol. 49 n. 1/4, pp. 77-97, 2002.
12. M.Ajmone Marsan, M.Garetto, P.Giaccone, E.Leonardi, E.Schiattarella, A.Tarello, "Using Partial Differential Equations to Model TCP Mice and Elephants in Large IP Networks", *IEEE Infocom 2004,* Hong Kong, March 2004.
13. F.Baccelli, A.Chaintreau, D.Mc Donald, D.De Vleeschauwer, "A Mean Field Analysis of Interacting HTTP Flows", *ACM SIGMETRICS 2004*, New York, NY, June 2004.
14. G.Carofiglio, E.Leonardi, M.Garetto, M.Ajmone Marsan, A.Tarello, "Beyond Fluid Models: Modelling TCP Mice in IP Networks with Non-Stationary Random Traffic", submitted for publication.
15. S.Floyd, V.Jacobson, "Random Early Detection Gateways for Congestion Avoidance", *IEEE/ACM Transactions on Networking*, vol. 1, n. 4, pp. 397-413, August 1993.

16. J.Padhye, V.Firoiu, D.Towsley, and J.Kurose, “Modeling TCP Throughput: A Simple Model and its Empirical Validation,” *ACM SIGCOMM’98 - ACM Computer Communication Review*, 28(4):303–314, September 1998.
17. A.Feldmann, W.Whitt, “Fitting Mixtures of Exponentials to Long-Tail Distributions to Analyze Network Performance Models”, *IEEE Infocom 97*, Kobe, Japan, April 1997.
18. M.Ajmone Marsan, M.Garetto, P.Giaccone, E.Leonardi, E.Schiattarella, A.Tarello, ”Using Partial Differential Equations to Model TCP Mice and Elephants in Large IP Networks”, Technical report available at `http://www.telematics.polito.it/garetto/papers/TON-2004-leonardi.ps`
19. M.Garetto, D.Towsley, “Modeling, Simulation and Measurements of Queuing Delay Under Long-Tail Internet Traffic”, *ACM SIGMETRICS 2003,* San Diego, CA, USA, June 2003.
20. M.Garetto, R.Lo Cigno, M.Meo, M.Ajmone Marsan, “Modeling Short-Lived TCP Connections with Open Multiclass Queuing Networks”, *Computer Networks Journal*, vol. 44, n. 2, pp. 153–176, February 2004.
21. M.Meo, M.Garetto, M.Ajmone Marsan, R.Lo Cigno, “On the Use of Fixed Point Approximations to Study Reliable Protocols over Congested Links”, *IEEE Globecom 2003*, San Francisco, CA, December 2003.