Centrum voor Wiskunde en Informatica

REPORT*RAPPORT*

PNA

Probability, Networks and Algorithms

Probability, Networks and Algorithms

A versatile model for TCP bandwidth sharing in networks with heterogeneous users

D. Abendroth, H. van den Berg, M.R.H. Mandjes

REPORT PNA-E0417 NOVEMBER 2004

CWI is the National Research Institute for Mathematics and Computer Science. It is sponsored by the Netherlands Organization for Scientific Research (NWO).
CWI is a founding member of ERCIM, the European Research Consortium for Informatics and Mathematics.

CWI's research has a theme-oriented structure and is grouped into four clusters. Listed below are the names of the clusters and in parentheses their acronyms.

Probability, Networks and Algorithms (PNA)

Software Engineering (SEN)

Modelling, Analysis and Simulation (MAS)

Information Systems (INS)

P.O. Box 94079, 1090 GB Amsterdam (NL)
Kruislaan 413, 1098 SJ Amsterdam (NL)
Telephone +31 20 592 9333
Telefax +31 20 592 4199

ISSN 1386-3711

A versatile model for TCP bandwidth sharing in networks with heterogeneous users

ABSTRACT
Enabled by the emergence of various access technologies (such as ADSL and wireless LAN), the number of users with high-speed access to the Internet is growing rapidly, and their expectation with respect to the quality-of-service of the applications has been increasing accordingly. With TCP being the ubiquitous underlying end-to-end control, this motivates the interest in easy-to-evaluate, yet accurate, performance models for a TCP-based network shared by multiple classes of users. Building on the vast body of existing models, we develop a novel versatile model that explicitly captures user heterogeneity, and takes into consideration dynamics at both the packet level and the flow level. It is described how the resulting multiple time-scale model can be numerically evaluated. Validation is done by using NS2 simulations as a benchmark. In extensive numerical experiments, we study the impact of heterogeneity in the round-trip times on user-level characteristics such as throughputs and flow transmission times, thus quantifying the resulting bias. We also investigate to what extent this bias is affected by the networks' `packet-level parameters', such as buffer sizes. We conclude by extending the single-link model in a straightforward way to a general network setting. Also in this network setting the impact of heterogeneity in round-trip times is numerically assessed.

2000 Mathematics Subject Classification: 60K25, 68M20
Keywords and Phrases: TCP; user heterogeneity; performance; throughput; round-trip times; packet level; flow level; acyclic networks
Note: The work of H. van den Berg and M. Mandjes was partially funded by the Dutch Ministry of Economic Affairs (through its agency SENTER/NOVEM) under the programme `Technologische Samenwerking ICT Doorbraakprojecten', project TSIT 2031 EQUANET. M. Mandjes is also with Korteweg-de Vries Institute, University of Amsterdam, Amsterdam, the Netherlands, and EURANDOM, Eindhoven, the Netherlands. D. Abendroth is currently with Bayerische Motor Werke, München, Germany.

A versatile model for TCP bandwidth sharing in networks with heterogeneous users *

Dirk Abendroth[⋆,○], Hans van den Berg[◇,○], Michel Mandjes[○,•]
⋆ Technical University Hamburg-Harburg, Denickestr. 17, 21073 Hamburg, Germany
○ University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands
◇ TNO Telecom, Brasserplein 2, P.O. Box 5050, 2600 GB Delft, The Netherlands
• CWI, Kruislaan 413, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

Abstract

Enabled by the emergence of various access technologies (such as ADSL and wireless LAN), the number of users with high-speed access to the Internet is growing rapidly, and their expectation with respect to the quality-of-service of the applications has been increasing accordingly. With TCP being the ubiquitous underlying end-to-end control, this motivates the interest in easy-to-evaluate, yet accurate, performance models for a TCP-based network shared by multiple classes of users. Building on the vast body of existing models, we develop a novel versatile model that explicitly captures user heterogeneity, and takes into consideration dynamics at both the packet level and the flow level. It is described how the resulting multiple time-scale model can be numerically evaluated. Validation is done by using NS2 simulations as a benchmark. In extensive numerical experiments, we study the impact of heterogeneity in the round-trip times on user-level characteristics such as throughputs and flow transmission times, thus quantifying the resulting bias. We also investigate to what extent this bias is affected by the networks' 'packet-level parameters', such as buffer sizes. We conclude by extending the single-link model in a straightforward way to a general network setting. Also in this network setting the impact of heterogeneity in round-trip times is numerically assessed.

Key words: TCP, user heterogeneity, performance, throughput, round-trip times, packet level, flow level, acyclic networks

*The work of H. van den Berg and M. Mandjes was partially funded by the Dutch Ministry of Economic Affairs (through its agency SENTER / NOVEM) under the programme 'Technologische Samenwerking ICT Doorbraakprojecten', project TSIT 2031 EQUANET. M. Mandjes is also with Korteweg-de Vries Institute, University of Amsterdam, Amsterdam, the Netherlands, and EURANDOM, Eindhoven, the Netherlands.

1 Introduction

Enabled by enhanced access technologies such as 'fiber to the home', ADSL, wireless LANs or UMTS, the number of users with high-speed access to the Internet is increasing rapidly. At the same time, more and more Internet applications require some sort of minimum quality-of-service (QoS), expressed in terms of delay, throughput, etc. With most of the data transfers relying on TCP as underlying end-to-end transport protocol, the performance of TCP-controlled networks has become a prominent theme in networking research.
TCP has been designed to support efficient and reliable transmission of elastic data flows, tolerating some variations in the throughput. In particular, based on implicit information about the level of network congestion (round-trip time, packet loss) TCP increases or decreases the sending rate in order (to attempt) to provide a fair share of the network resources to all users. However, it is clear that heterogeneous user behavior might lead to asymmetries in the experienced performance. For instance, one may wonder to what extent it pays off to have a higher access rate or a shorter round-trip time than the other TCP flows. It is this relation between user heterogeneity and flow-level performance (throughput, flow transfer times) that is studied in this paper.
A study on the impact of user heterogeneity could be done relying exclusively on simulation tools (like the NS2 [15] TCP simulator), but such an approach has its well known inherent limitations; in particular, (basic) simulation methods hardly allow for doing sensitivity analysis, as it may already be rather time-consuming to get a reliable estimate for a single parameter instance. Therefore we have chosen to set up a model that allows for an analytical approach, or, when its evaluation turns out to be too complicated, a hybrid approach in which the role played by simulation is minimized.

As motivated by the above, TCP's widespread use and complex behavior have triggered the search for simple and transparent, yet accurate, mathematical techniques for performance analysis. Many performance models have been proposed, which can be roughly divided into *packet-level models* and *flow-level models*.
Packet-level models describe the detailed dynamics of TCP, related to the evolution of the flows' window sizes. The occurrence of packet loss provides the users with information on the current congestion level: during periods of low utilization TCP expands the flows' window sizes (and transmission rates), whereas these are reduced when packet loss is experienced. Evidently, the detailed behavior of a link fed by various classes of TCP flows, is intrinsically difficult to capture, particularly due to these complex interactions between the source behavior and the congestion level. However, by assuming a constant number of greedy (i.e., persistent) flows, and after imposing some additional simplifications, explicit expressions were obtained for the flows' throughput, as a function of the packet loss probability; early references are Kelly [17], Mathis *et al.* [23], and Padhye *et al.* [26]. Noticing that in return the packet loss probability of a bottleneck link is a function of the offered load (and hence essentially also of the throughput), we obtain two equations in two unknowns, and as a result the throughput can be found in a rather elementary way,

see e.g. [3, 8, 19]. Some other papers that explicitly model TCP's window dynamics, and which at the same time allow user heterogeneity, are [12, 13, 24]; it is noted that the use of such detailed models tends to be severely limited by scalability constraints.
Where packet-level models consider a constant set of persistent flows feeding into a link, flow-level models explicitly focus on the dynamics of the number of flows present. In other words: flow-level models focus on a somewhat less detailed time-scale than packet-level models, namely the time-scale at which flows arrive and depart, see for instance [9, 22]. It is assumed that, at the moment such a flow-level transition takes place, the allocation of the transmission rates to the individual flows adapts instantly. This enables the use of *processor sharing* (PS) queues, addressed in great generality by Cohen [10]. Over the past years this class of models has gained ground as a generic description of TCP's flow-level dynamics, as advocated in, e.g., [5, 18, 25].

It is clear that TCP's packet-level and flow-level have a strong mutual dependency, which motivates the attempts to develop a unified approach. In this respect we mention the pioneering work of Gibbens *et al.* [14], who consider the network extension of the single-link packet-level models mentioned above, enabling them to compute the throughputs for any given number of flows simultaneously present at the various routes through the network. Then they weigh these throughputs with an *a priori* supposed distribution for this number of 'concurrent flows' (Poisson, geometric), in order to 'emulate' the flow level. Though reasonable, a rigorous justification of assuming these specific flow-level distributions was lacking. This motivated why Lassila *et al.* [21] have considered ways to *derive* (i.e., to 'endogeneously determine') the flow-level distribution from the model, rather than to exogeneously impose a distribution. It is noted that [21] has succeeded in doing so in a single-link setting with homogeneous input (i.e., flows stem from a common distribution, and have the same round-trip times and access rates).
In the present paper we build on the results of [14, 21], but we add a number of substantial enhancements. Relative to [21], a first contribution of our work is that we allow for heterogeneous input at the flow level: several classes are distinguished (characterized by flow arrival rate, flow size, round-trip time, and access rate). Then the procedure is that we first use packet-level models to compute the (per-class) throughputs for a fixed number of flows present, and then we use these as the input for a PS-type of flow-level model. It is noted that the PS model that arises in this setting has the flavor of a so-called *discriminatory processor sharing* (DPS) system, which is in general notoriously hard to study analytically, see, e.g., [11]. A second improvement over [21] is that our multiple time-scale framework lends itself to being extended to network settings. The model can be used to assess the effect of user heterogeneity in often highly complex multi-link situations. Compared to [14] the major improvement is that we, as in [21], *derive* the flow-level distribution (i.e., the joint distribution of the number of 'concurrent flows' of the various types), rather than that we impose *a priori* some distribution.
It is noted that, particularly in the situation that the number of user classes grows large, it may become extremely time-consuming to numerically solve the flow level. For those situations we propose a 'hybrid' method: the packet-level is solved numerically, and the

resulting throughputs are used as input in the *simulation* of the flow level. It is noted that such an approach is still substantially faster than detailed time-scale (NS2) simulations, as only the flow-level jumps need to be simulated rather than the full packet-level dynamics. We remark that a somewhat similar hybrid approach was developed in [4]; there the simulated jumps correspond to congestion epochs.

The remainder of this paper is organized as follows. In Section 2 we introduce our packet / flow level model for heterogeneous TCP-controlled traffic transmitted over a buffered link. Section 3 presents a number of qualitative validation experiments. Quantitative validation, as reported in Section 4, is done by using the simulator NS2 [15] as a benchmark. Also a number of other experiments are performed, with a strong focus on the impact of the heterogeneity in round-trip times on the throughputs experienced by the various user classes. We also give some further comments on the relation with discriminatory processor sharing. Finally, Section 5 describes the extension of the single-link model to acyclic multiple-link network scenarios. Importantly, it is shown that 'packet-level parameters' such as buffer sizes, do have a significant impact on the way TCP allocates bandwidth. Section 6 summarizes the results and concludes the paper.

2 Integrated packet/flow level modelling approach

In this section we introduce a mathematical model that describes a buffered network link fed by a fluctuating set of heterogeneous TCP flows. As argued in the introduction, a model that captures all the details is far too complex to analyze.
A commonly used resort is to rely on *time-scale decomposition*: distinguish between multiple time-scales, solve these separately, and integrate them into the performance measures of interest. We here follow such a decomposition approach, by decoupling the *packet level* and the *flow level*. Recall that the packet level describes the performance when the link is used by a fixed set of persistent flows, whereas the flow level describes the fluctuations of the number of flows simultaneously present. Interestingly, the decomposition allows us to analyze the effect of typical packet-level parameters (buffer size, round-trip times, etc.) on flow-level performance (the average number of flows in the system, flow transfer times, etc.). It is stressed that our specific focus is on assessing the impact of user heterogeneities on the performance. To this end, we will introduce m classes of flows.

2.1 Modelling assumptions

Each flow class i, for $i = 1, \ldots, m$, is characterized by four parameters. (i) The rate at which flows of class i are initiated is denoted by λ_i; it is assumed that these arrivals follow a Poisson process, or, in other words, that the interarrival times are exponential with mean λ_i^{-1} (units of time). (ii) Flows of class i have an exponential size with mean μ_i^{-1} (packets); it is assumed throughout this paper that packets are equally sized. (iii) The 'physical' round-trip time of a class-i packet, i.e., due to propagation and all other non-congestion-dependent factors, is given by RTT_i^0 time units (in other words: the queueing

delay in the buffer is *not* incorporated in RTT_i^0). (iv) The access rate of a class-i user is given by R_i (packets per time unit). Notice that the maximum window size $W_{\max,i}$ and the round-trip time also put a limit on the users' transmission rate: $R_i \leq W_{\max,i}/\mathrm{RTT}_i^0$.
The link, to be interpreted as a *bottleneck* link, is characterized by its service speed (or link rate) C (expressed in packets per time unit) and buffer size B (packets). The service discipline is *first-in-first-out*. The m traffic classes share these common network resources. For reasons of stability, the incoming load is limited to the system's capacity: $\sum_{i=1}^{m} \lambda_i/\mu_i < C$.
Now we subsequently describe the packet level and flow level, and describe how these allow the computation of the performance measures of our interest.

2.2 Packet level

In the packet-level model there is a *fixed* number n_i of *persistent* TCP flows of class i, for $i = 1, \ldots, m$. The main objective of the packet-level analysis is to compute, for a given vector $\bar{n} := (n_1, \ldots, n_m)$, the throughputs $t_1, \ldots, t_m$ of the various classes.
In our approach we rely on a relation between the mean throughput, the packet-loss probability, and the round-trip time, that was derived for the case of just one class of TCP connections, see e.g. [17, 23, 26]; in our study we will rely on the formula given in [17]. Suppose there are n flows with round-trip time RTT and access rate R, who experience a packet-loss probability p. Then the (total) throughput is, during the congestion-avoidance phase, approximated by

$$t(n) = \min\left\{nR, \frac{n}{\mathrm{RTT}}\sqrt{\frac{2(1-p)}{p}}\right\};$$

evidently, the left-hand argument of the min function simply limits throughputs to the access rate R. By borrowing the above formula, we obtain for our case of m heterogeneous traffic classes:

$$t_i(\bar{n}) = \min\left\{n_i R_i, \frac{n_i}{\mathrm{RTT_i}}\sqrt{\frac{2(1-p_i)}{p_i}}\right\}, \qquad i = 1, \ldots, m. \tag{1}$$

It is clear that the round-trip time RTT_i consists of its fixed component RTT_i^0, increased by the queueing delay experienced by class i. With δ_i denoting the mean queueing delay of class i, we could write $\mathrm{RTT}_i = \mathrm{RTT}_i^0 + \delta_i$. Regarding loss and delay, it seems reasonable to assume that there will not be too much discrepancy between the classes, which allows us to write $\delta_i = \delta$ and $p_i = p$, for $i = 1, \ldots, m$.
Having expressed the throughput in terms of the packet loss and delay, we have to find loss and delay as a function of the throughput; clearly, having these relations at our disposal, we are able to compute the $t_i(\bar{n})$. To this end, we make the approximation that, at the packet level, packets (of equal size) arrive at the buffer according to a Poisson process. Hence, we can approximate the packet-loss probability and the mean delay by those of

the corresponding M / D / 1 / B queue. In self-evident notation:

$$p \equiv p(\bar{n}) = p_{\mathrm{M/D/1}}\left(\sum_{i=1}^{m} t_i(\bar{n}), C, B\right); \qquad \delta \equiv \delta(\bar{n}) = \delta_{\mathrm{M/D/1}}\left(\sum_{i=1}^{m} t_i(\bar{n}), C, B\right). \quad (2)$$

Techniques for computing (or approximating) both the loss probability and mean delay in M / D / 1 queues are described in, e.g., [27, Section 15.1]; the mean delay is given by their Eq. (15.1.2), whereas the loss probability could be accurately approximated by exponential expansion in the spirit of their Eq. (15.1.6).

The assumption of Poisson arrivals at the packet level has been very common in the literature, see e.g. [3, 14, 19]. Given the fact that many flows are multiplexed, and in the absence of a detailed description of the packet arrival process, we have chosen to do so. It can obviously not be justified that this choice yields the best match with the actual queueing behavior. We emphasize, however, that the methodology presented here does not critically rely on this Poisson assumption; if there is a reason to assume that some other arrival process provides a better match, one could employ the corresponding queueing model instead.

It is clear that inserting (2) into (1) yields a fixed-point problem of m non-linear equations with m unknowns (i.e., $t_1, \ldots, t_m$; we suppress the $\bar{n}$ for convenience). Now consider the right-hand side of (1) as a function of t_i, for fixed t_j, $j \neq i$. Evidently, both the mean queueing delay δ and the loss probability p increase in t_i. It can be verified easily that this implies that the right-hand side of (1) decreases in t_i, and has limit 0. As the left-hand side (i.e., the identity function) increases from 0 to ∞, we conclude that there is, for given t_j, $j \neq i$, a fixed point.

2.3 Flow level

The flow level describes the dynamics related to arrivals and departures of flows (i.e, TCP connections). When there are $\bar{n}$ flows in the system, we assume that class i is served at a rate $s_i(\bar{n}) := t_i(\bar{n})(1 - p(\bar{n}))$, where $t_i(\bar{n})$ and $p(\bar{n})$ follow from the packet-level fixed-point equations; $s_i(\bar{n})$ is often referred to as the class-i 'goodput'. All individual flows of type i are assumed to get a 'fair share' $s_i(\bar{n})/n_i$ of the class-i goodput. It is remarked that our time-decomposition entails that we implicitly assume that, when the number of active flows changes, service rate adaptation takes place instantly. Put differently, our approach neglects the rate fluctuations at the packet level (which are due to the window-size dynamics, in response to packet losses).

As the flow sizes were assumed to be exponential, we can model the flow-level dynamics by a (continuous-time) Markov chain. The fact that any flow gets a fair share of the per-class goodput, gives our model the flavor of a processor-sharing system. The single-class system can be solved explicitly; see [21], relying on the results of [10]. The user heterogeneity, as present in our model, however, makes the analysis essentially more difficult. In particular, as mentioned in the introduction, the flow-level model that arises in this setting has the flavor of a DPS system, a class of models that is in general hard to study analytically.

We now give the transition rates of the continuous-time Markov chain governing the flow level, with state space $\mathbb{N}_0^m$. The transition rate corresponding to an arrival of a type-i flow, denoted by $q(\bar{n}+e_i \mid \bar{n})$, is clearly equal to λ_i; here $e_i := (0,\ldots,0,1,0,\ldots,0)$, where the 1 is put on the ith position. It can be verified easily that the transition rate from $\bar{n}$ to $\bar{n}-e_i$, i.e., $q(\bar{n}-e_i \mid \bar{n})$, equals $\mu_i s_i(\bar{n})$ (under the proviso that $n_i > 0$; otherwise $\bar{n}-e_i$ clearly does not belong to the state space). Let $Q(\bar{n})$ denote the total transition rate out of state $\bar{n}$:

$$Q(\bar{n}) := \sum_{i=1}^{m} (\lambda_i + \mu_i s_i(\bar{n})).$$

Hence, for any state $\bar{n}$ we can solve the packet-level fixed-point equations to find the transition rates at the flow level. These transition rates determine the equilibrium distribution at the flow level; this long-run distribution is found by solving the balance equations

$$\pi(\bar{n})Q(\bar{n}) = \sum_{i=1}^{m} \pi(\bar{n}+e_i)q(\bar{n} \mid \bar{n}+e_i) + \sum_{i=1}^{m} \pi(\bar{n}-e_i)q(\bar{n} \mid \bar{n}-e_i), \tag{3}$$

in conjunction with $\sum_{\bar{n}} \pi(\bar{n}) = 1$.

In practice, there are several principal problems when computing the equilibrium distribution at the flow level by solving the above balance equations, particularly when the dimension m is large. In the first place, the state space is (countably) infinite, so to do numerical computations one has to apply some truncation. In the second place, the linear system of equations could be ill-conditioned, and may consequently lead to numerical problems; it is noted that usually many elements of the state space are (at the flow level) extremely rare, leading to small equilibrium probabilities.

To overcome these problems, we propose, in case solving the system (3) leads to numerical problems, to use simulation at the flow level. In more detail, the procedure would be the following. Start the simulation at some $\bar{n}^{(0)}$, compute the throughputs $t_i(\bar{n}^{(0)})$ for $i = 1,\ldots,m$, then determine the transition rates $q(\bar{n}^{(0)}+e_i \mid \bar{n}^{(0)})$ and $q(\bar{n}^{(0)}-e_i \mid \bar{n}^{(0)})$, simulate a transition of the continuous-time Markov chain to a new state $\bar{n}^{(1)}$, etc.; the procedure of computing throughputs, determining transition rates and simulating a transition can be repeated until some stopping criterion is satisfied.

For the sake of completeness, we mention that it is well known that a transition of the Markov chain (from state $\bar{n}$) is most efficiently simulated by sampling the time until the transition from an exponential distribution with mean $Q(\bar{n})^{-1}$, and by letting the process jump to state $\bar{n}+e_i$ with probability $q(\bar{n}+e_i \mid \bar{n})/Q(\bar{n})$ and to $\bar{n}-e_i$ with probability $q(\bar{n}-e_i \mid \bar{n})/Q(\bar{n})$, for $i = 1,\ldots,m$.

The procedure can be significantly accelerated when one *stores* the throughputs $t_i(\bar{n})$ during the simulation. In other words, the packet-level fixed point for a state $\bar{n}$ needs to be calculated only if the process visits $\bar{n}$ for the first time.

As mentioned earlier, usually a significant part of the state space corresponds to extremely small equilibrium probabilities, which hardly contribute to the (flow-level) performance measures of our interest. Hence, also from this point of view simulating the flow level has important advantages over numerically solving the balance equations.

The procedure described above, in which simulation is used at the flow level, could be characterized as a *hybrid* approach, between computation and simulation. An alternative could be to simulate *both* packet-level and flow-level dynamics, for instance by using NS2. Our approach, however, is substantially faster. Clearly, the main difference lies in the fact that our integrated approach does not need to simulate the detailed packet-level dynamics; these are summarized by the (computed) throughput values. In this respect there is an interesting similarity with the hybrid simulation / computation approach proposed in [4]; an important difference is that in our approach the flow-level fluctuations (flow arrivals, flow departures) are the jump epochs, whereas in [4] this role is played by the congestion epochs.

After having determined the equilibrium distribution $\pi(\bar{n})$ by using the approach described above, it is straightforward to obtain the mean number of active flows of class i:

$$n_i^\star = \sum_{\bar{n}} n_i \pi(\bar{n}),$$

$i = 1, \ldots, m$. Using Little's law, the mean transfer delay D_i for a flow of class i equals $D_i = n_i^\star / \lambda_i$.

3 Qualitative validation

In this section we present results from a number of first experiments. These are meant as a 'sanity check', in that we verify whether the model shows the type of bias we expect (for instance: one expects that the flows with the longer round-trip time receive the smaller goodput). We mention that Matlab was used for solving the m fixed-point packet-level equations, obtained by inserting (2) into (1).

Our first experiment focuses on the impact of heterogeneity in the mean flow size on the packet-level performance; we take $m = 2$ classes. For different values of $\bar{n} = (n_1, n_2)$ we have computed the goodputs $s_1(\bar{n})$ and $s_2(\bar{n})$. Figure 1.A shows the 'normalized goodputs' $s_i(\bar{n})/C$. We have chosen the mean file sizes $1/\mu_1 = 500$ and $1/\mu_2 = 1000$ [packets]; in this and the following experiments the flow arrival rates are such that both classes are responsible for half of the load. All other class-specific parameters (round-trip times, access rates) are the same for both classes.
It is clear that the heterogeneity of flow sizes does not have any influence on the goodputs for fixed $\bar{n}$. This follows directly from the fact that the buffer content in the M / D / 1 queue only depends on the *load* (and not on the mean flow sizes). We indeed find this symmetry in the graph; in the third experiment, we investigate whether there is still symmetry when also the flow level is taken into account. In contrast to the above, a heterogeneity in round-trip times *does* have a strong impact, also at the packet-level, as becomes clear in the next experiment.

The second experiment assesses the impact of heterogeneous round-trip times on the packet-level performance. Now the round-trip time is the only class-specific parameter

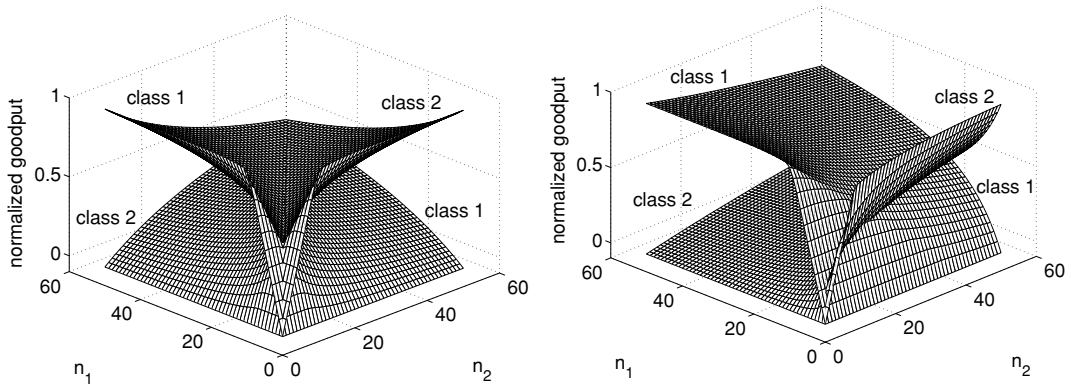

Figure 1: Packet-level performance: goodputs. A. (left) 'Normalized goodput' per class, packet level, two different mean flow sizes (500 and 1000 [packets]), access rates 1 [Mbit/sec.], round-trip times 100 [msec.], buffer size 10 [packets]. B. (right) 'Normalized goodput' per class, packet level, two different round-trip times (50 and 200 [msec.]), access rates 1 [Mbit/sec.], mean flow size 500 [packets], buffer size 10 [packets].

that is different for both classes. The results are plotted in Figure 1.B. We conclude that the surface shapes are no longer symmetric, as expected. The goodput surface for class 2 ($\mathrm{RTT}_2^0 = 200$ msec.) increases 'slower' than that of class 1 ($\mathrm{RTT}_1^0 = 50$ msec.), due to its larger round-trip time.

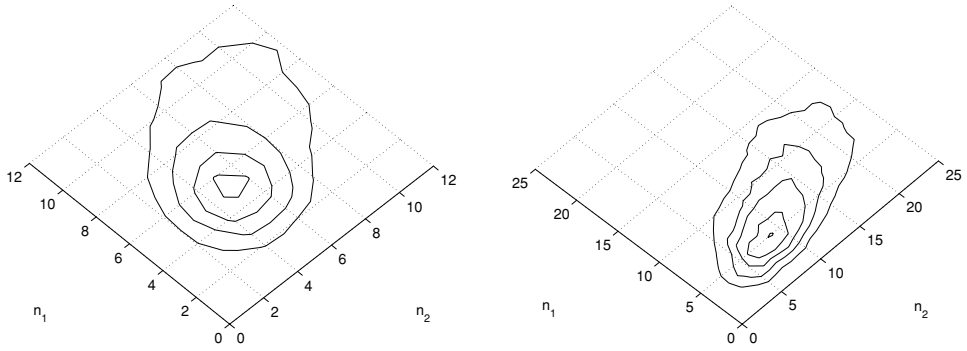

Figure 2: Impact of user heterogeneity on flow level: contour lines of state probabilities under 90% load, flow level. A. (left) heterogeneity of flow sizes. B. (right) heterogeneity of round-trip times.

The third experiment illustrates the impact of user heterogeneity on the flow-level performance. In particular, we consider the contour lines of the state probabilities. The results in Figure 2 are based on the corresponding packet-level results, which were visualized in the previous figures. In fact, it is remarked that the case of exponentially distributed

flow sizes with different mean values (at a common round-trip time) still belongs to the class of Generalized Pprocessor Sharing that was introduced in [10], which explains the symmetry in case of two different flow sizes (with common round-trip times). The shape of the contour lines in case of two different round-trip times (with same mean flow size) tends strongly towards the n_2 axis. Hence, despite the fact that both classes correspond to the same offered load, the difference in round-trip times makes the flow-level performance highly asymmetric, as is captured nicely by our model.

4 Quantitative validation; impact of user heterogeneity

This section presents the quantitative validation of our packet/flow level approach by using NS2 simulations; we have chosen to use NS2 [15] as a benchmark, motivated by the fact that it mimicks TCP at a detailed time-scale. Hence, comparing results obtained by using our methodology with results obtained under NS2, we assess to what extent it is justified to replace the packet-level dynamics by a static bandwidth allocation according to the rates $s_i(\bar{n})$ (when there are $\bar{n}$ flows in the system), as is done in our approach.
We also perform a number of further experiments assessing the impact of heterogeneity on the performance bias, with emphasis on the impact of packet-level parameters (such as the buffer size).

4.1 Scenarios

We consider the single bottleneck scenario introduced in Section 2, with $m = 2$ traffic classes. Recall that the number of concurrently active flows within each class (n_1 and n_2) is varying over time due to flow arrivals and departures.
In our experiments we have chosen the loads per class 1 and 2 to be equal, i.e., $\lambda_1/\mu_1 = \lambda_2/\mu_2 = \rho/2$, where $\rho < C$ denotes the system load. We concentrate on the heterogeneity with respect to round-trip times and file sizes; we assume that all connections have the same access link rate, i.e., $R_1 = R_2$. We set the link rate C equal to 10 [Mbit/s]. All transmitted packets have a size of 1500 [bytes] (cf. MTU size).
In practical situations round-trip times vary, roughly speaking, over a range of 50 to 200 msec. – pings within Europe, as well as pings from Europe to other continents have been answered with round-trip times 40 msec. up to 180 msec. Therefore we have chosen to consider round-trip times of 50, 100, and 200 msec. We study the influence of small and large buffer sizes, i.e., 10 and 50 [packets], and access rates, i.e., 1 and 2 Mbit/sec. In the sequel we provide a comparison of results for a full permutation of all the above parameters, where, in addition, both classes have different round-trip times. In view of the results of the previous section, we have not varied the mean file size; we have set this to 500 [packets] for all flows. Recall that we took a constant packet size of 1500 bytes.

4.2 Simulation in NS2

As mentioned, NS2 will serve as benchmark to validate our approach. For every parameter setting, the NS2 simulation result is based on 50 independent replicas. The graphs in Section 4 show mean values and confidence intervals (of confidence 95%). Each replica corresponds to 300 sec. simulated time (i.e., in 'real time'). In order to suppress the effect of the initial transient, we cut off 50 sec. at the beginning of each simulation, i.e., the gross simulation time per replica is 350 sec.
The 'physical' round-trip times RTT_1^0 and RTT_2^0 used in our model are, in the NS2 scenario, translated into delays at the access link and at the bottleneck link. These delays contribute twice to RTT_1^0 and RTT_2^0: once for the packet transmission, and another time for the transmission of the acknowledgment. The delay at the bottleneck link is set to a fixed value of 5 ms. The difference in round-trip times between the classes is realized by an appropriate choice of the access link delays.

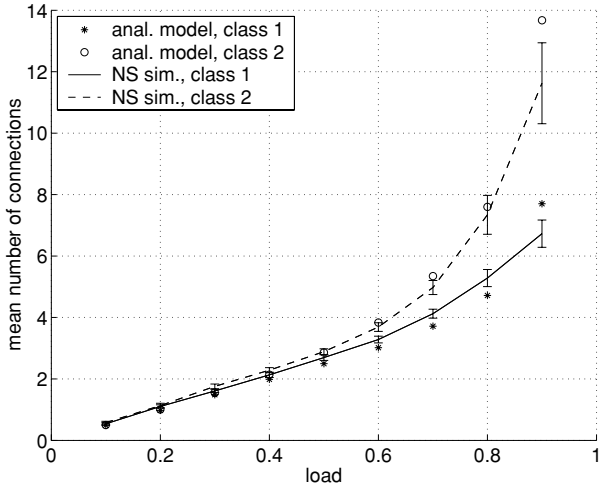

Figure 3: Mean number of connections for two classes, under varying load. Access rates 1 [Mbit/sec.], $\mathrm{RTT}_1^0 = 100$ and $\mathrm{RTT}_2^0 = 200$ [msec.], buffer size 10 [packets].

4.3 Description of the experiments

For different values of the system load ρ, Figure 3 shows the mean number of active connections per class $n_i^\star$; the graph contains both the values obtained by using our method, and the values obtained by using NS2 (including their confidence intervals).
In Figures 4 through 7 we have focused on the perhaps somewhat more appealing performance measure of the mean flow transfer delay (rather than the mean number of flows present); these can be expressed in one another through Little's law $D_i = n_i^\star/\lambda_i$. Bearing in mind the complex TCP dynamics, the figures show that our model results coincide remarkably good with the NS simulation results, and it does so for a wide set of parameter combinations.

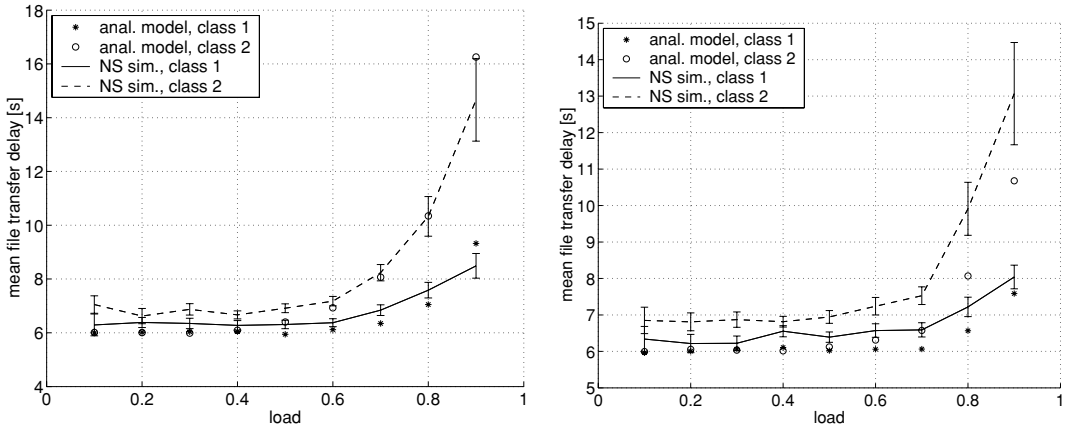

Figure 4: Mean file transfer delay [sec.]; access rates 1 [Mbit/sec.], $\mathrm{RTT}_1^0 = 100$ and $\mathrm{RTT}_2^0 = 200$ [msec.], buffer size left panel: 10 [packets], buffer size right panel: 50 [packets].

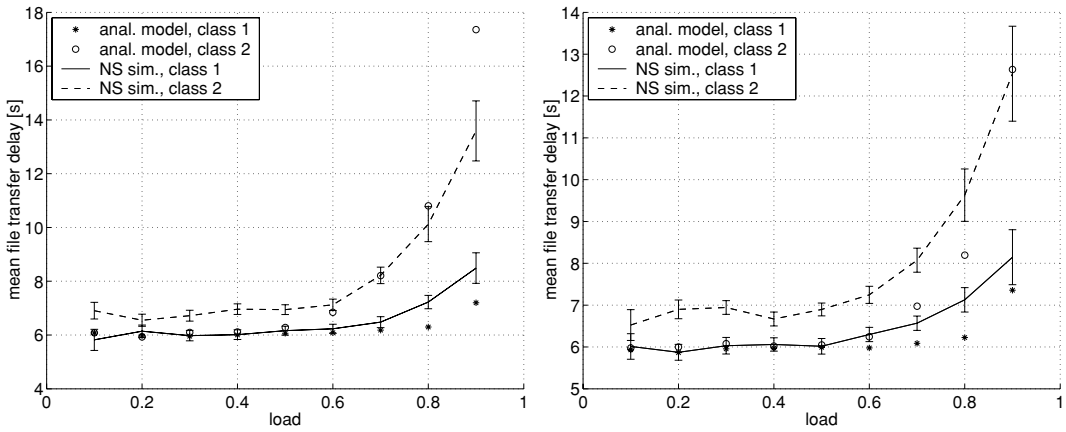

Figure 5: Mean file transfer delay [sec.]; access rates 1 [Mbit/sec.], $\mathrm{RTT}_1^0 = 50$ and $\mathrm{RTT}_2^0 = 200$ [msec.], buffer size left panel: 10 [packets], buffer size right panel: 50 [packets].

4.4 Discussion on the accuracy of the approach

We now discuss in detail the accuracy of our packet/flow level approach. We mention the following observations:

- For low values of the load, our model predicts nearly identical performance for class 1 and class 2. The NS2 simulations, however, show that these results are systematically too optimistic: the real performance is slightly higer. Moreover, the NS2 simulation indicate that the flows with the short round-trip time experience the smaller delay.

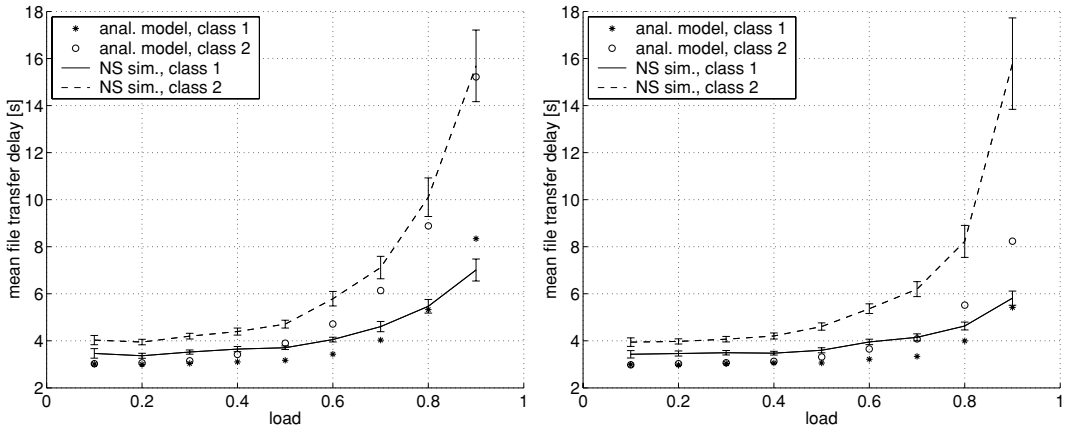

Figure 6: Mean file transfer delay [sec.]; access rates 2 [Mbit/sec.], $\mathrm{RTT}_1^0 = 100$ and $\mathrm{RTT}_2^0 = 200$ [msec.], buffer size left panel: 10 [packets], buffer size right panel: 50 [packets].

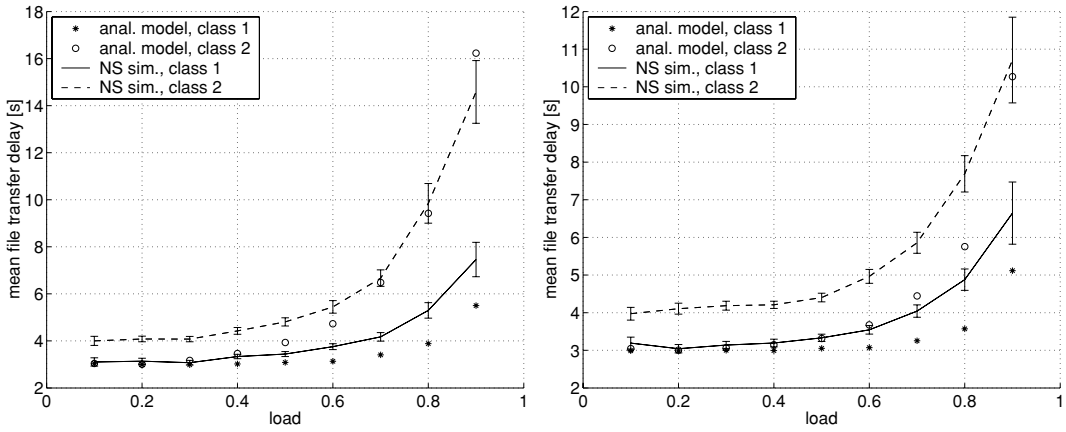

Figure 7: Mean file transfer delay [sec.]; access rates 2 [Mbit/sec.], $\mathrm{RTT}_1^0 = 50$ and $\mathrm{RTT}_2^0 = 200$ [msec.], buffer size left panel: 10 [packets], buffer size right panel: 50 [packets].

It is expected that this behavior is caused by the fact that our method does not incorporate the effects of TCP's slow-start phase: the time-scale decomposition entails that we do as if the flow is, after being initiated, immediately in its 'stationary behavior'. It is noted that, during the slow-start phase, the round-trip times play a crucial role, as they dictate how fast the window size can grow. In [21] the effects of the slow-start phase were successfully compensated, in order to avoid too optimistic estimates; one could pursue the development of such a procedure for the model under study.

As said above, the fact that we did not incorporate the slow-start phase makes our method too optimistic. It is noted that this is the case for both for low and for high values of the load.

- When comparing Figures 4-5 (access rate of 1 Mbit / sec.) with Figures 6-7 (access rate of 2 Mbit / sec.), it could be concluded that for low values of the load the transfer delays are not so much determined by the round-trip times, but rather by the access rate constraints. This is reflected by the fact that the transfer delays in Figs. 4-5 are (roughly) two times as high as in Figs. 6-7.

- The numerical results (also the ones not shown here) indicate that our model (particularly for class 2) is less accurate for large buffers ($B = 50$) than for small buffers ($B = 10$). We suspect that this is due to the fact that the real packet arrival process is often substantially burstier than Poisson, leading to errors in the approximation of the loss probability p and queueing delay δ, which are typically smaller for a small buffer than for a large buffer.

Refinements of our approach, which take into account the issues identified above, are subjects for future research.

4.5 Quantification of the bias; equalizing effects

We finally say some words on the quantification of the bias, in the situation of heterogeneous round-trip times. There are several papers on this issue, see for instance [1, 20]. We here study the claim that per-flow throughputs are inversely proportional to the flow's round-trip time, in line with formula (1), see, e.g., [17, 23, 26]. In case of two classes sharing a bottleneck link, this would mean that

$$\frac{s_1(\bar{n})/n_1}{s_2(\bar{n})/n_2} \approx \frac{\mathrm{RTT}_2}{\mathrm{RTT}_1}. \tag{4}$$

Consider a situation with $\mathrm{RTT}_2^0/\mathrm{RTT}_1^0 = 2$. We plot the left hand side of (4), by using the numbers obtained in our packet-level model, see Figure 8.
Indeed, the ratio is nearly constant for somewhat larger values of n_1, n_2, and has indeed value 2; for 'low' states the ratio is significantly smaller. In fact, close to the origin the ratio is nearly 1, as such a small number of connections (with limited access rate) is not able to 'fill up the link'.
There are circumstances under which the performance asymmetries tend to be 'equalized'. In the first place, this is clearly the case for medium loads and limited access rates. For such loads the most probable states are relatively close to the origin, and Figure 8 then indicates that the 'normalized goodput ratio' will be close to 1.
A second factor that has impact on this 'equalizing effect' is the buffer size. Table 1 shows the limiting value of the 'normalized goodput ratio', i.e., the limit of $(s_1(\bar{n})/n_1)/(s_2(\bar{n})/n_2)$ for large n_1, n_2. We do so for varying (i) ratio of the ('physical') round-trip times $r := \mathrm{RTT}_2^0/\mathrm{RTT}_1^0$, and (ii) buffer size. The link capacity is still 10 [Mbit / sec.].

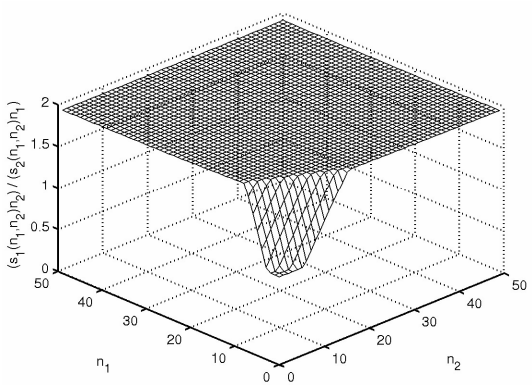

Figure 8: 'Normalized goodput ratio' $(s_1(n_1, n_2)/n_1)/(s_2(n_1, n_2)/n_2)$ plotted as function of the states (n_1, n_2); access rates 1 [Mbit/sec.], $\text{RTT}_1^0 = 100$ and $\text{RTT}_2^0 = 200$ [msec.], buffer size 10 [packets].

Table 1: 'Normalized goodput ratio', buffer sizes are in packets.

	$B = 10$	$B = 50$
$r = 2$	1.90	1.63
$r = 4$	3.45	2.38

From the table we conclude that for small buffers the 'normalized goodput ratio' is close to the ratio of the round-trip times, whereas for larger buffers this match is less good. An explanation lies in the fact that for larger buffers the round-trip times are increasingly determined by the queueing delay (rather than the 'physical' delay components), and this queueing delay is the same for both classes.
We conclude that it is not always accurate to assume goodputs inversely proportional to the round-trip time. It also entails that it could be inaccurate to approximate the model with heterogeneous users with discriminatory processor sharing [11], with weights inversely proportional to the classes' round-trip times, as done in, e.g., [2]. Our model nicely captures the 'equalizing effect' of the access rate limitation and the buffer size on the throughputs, as experienced by flows with different ('physical') round-trip times; it is noted that similar properties were observed in the simulations performed in [5].

5 Extension to networks

In this section we make a first step towards extending our single-link framework to a network setting. The model and analysis are presented in Section 5.1. The approach borrows elements of [14], but we remark again that the crucial difference between our approach and [14] is, that our model determines the distribution of the network popula-

tion 'endogeneously'. In Section 5.2, we consider a few examples, and investigate how these compare to other results on rate allocation.

5.1 Modelling and analysis

Consider a network, consisting of k links, characterized by their link rates C_j and buffer sizes B_j. Flows of class i subsequently pass a number of links. We say that $j \in \mathcal{R}_i$ if class i uses link j. Also, we say that $j_1 \prec_i j_2$ if $j_1, j_2 \in \mathcal{R}_i$, and in addition j_1 lies on this route *before* j_2. We assume that the network is acyclic.

The approach relies again on decoupling the packet level and the flow level. We first determine the throughput of class i, for a given user population $\bar{n}$; then this serves as input for the continuous-time Markov chain model that describes the flow level. The generalization of the throughput formula (1) is, for $i = 1, \ldots, m$,

$$t_i(\bar{n}) = \min \left\{ n_i R_i \,,\ \frac{n_i}{\mathrm{RTT}_i^0 + \sum_{j \in \mathcal{R}_i} \delta_j} \sqrt{\frac{2\left(1 - \sum_{j \in \mathcal{R}_i} p_j\right)}{\sum_{j \in \mathcal{R}_i} p_j}} \right\}. \tag{5}$$

Here p_j is the loss probability at link j, and δ_j the mean delay at link j; we assume that the per link loss probability and mean delay is constant across the user classes.

In return, the loss probabilities and mean delays depend on the load offered to the links. We can write

$$p_j \equiv p_j(\bar{n}) = p_{\mathrm{M/D/1}} \left(\sum_{i: j \in \mathcal{R}_i} t_i(\bar{n}) \cdot \left(1 - \sum_{k \prec_i j} p_k(\bar{n}) \right), C_j, B_j \right), \tag{6}$$

and

$$\delta_j \equiv \delta_j(\bar{n}) = \delta_{\mathrm{M/D/1}} \left(\sum_{i: j \in \mathcal{R}_i} t_i(\bar{n}) \cdot \left(1 - \sum_{k \prec_i j} p_k(\bar{n}) \right), C_j, B_j \right). \tag{7}$$

Notice that the load imposed on link j is 'thinned', along the various routes i, by the losses in its predecessing links (i.e., k such that $k \prec_i j$); the formula uses, as in [14], the approximation that losses at the various links are independent, cf. also [16], and the well known approximation $\prod_i (1 - x_i) \approx 1 - \sum_i x_i$ for small x_i. Combining (6) and (7) with the throughput formula (5) again yields m equations in m unknowns, which can be solved numerically.

The flow level can again be done by constructing a continuous-time Markov chain; it is obvious that again it can be solved by either solving the balance equations, or by simulating an m-dimensional random process. It is noted that the flow-level procedure remains essentially the same, and that the network topology is just reflected by the packet-level computations.

5.2 Examples; rate allocation in TCP

It has been claimed that the transmission rates allocated in TCP's congestion avoidance are well approximated by the optimizing t_i, $i = 1, \ldots, m$ in

$$\min_{t_1,\ldots,t_m} \sum_{i=1}^{m} \left(\frac{1}{\mathrm{RTT}_0^i}\right)^2 \frac{n_i}{t_i} \quad \text{under} \quad \sum_{i:j\in\mathcal{R}_i} t_i \leq C_j;$$

see for instance [6, Eq. (3)]. For a single link fed by m heterogeneous classes, when performing the optimization, one obtains, with $\kappa_i := (\mathrm{RTT}_i^0)^{-1}$,

$$t_i(\bar{n}) = \frac{C \cdot n_i \kappa_i}{\sum_{j=1}^{m} n_j \kappa_j},$$

i.e., the rate allocation is according to DPS with weights inversely proportional to the round-trip times. We have seen in the previous section that this approximation is particularly accurate when buffers are relatively small.
Now consider the somewhat more involved case of a network with multiple links. For ease we concentrate on the simplest, non-trivial model, as depicted in Figure 9, since this already shows a number of features that are not present in the single-link case. Let there be two links, both of rate C; let type 1 use link 1, type 2 use link 2, and type 3 use both links. It can be verified that the above minimization now yields

$$t_1(\bar{n}) = t_2(\bar{n}) = \frac{C \cdot \sqrt{(n_1\kappa_1)^2 + (n_2\kappa_2)^2}}{n_3\kappa_3 + \sqrt{(n_1\kappa_1)^2 + (n_2\kappa_2)^2}}; \quad t_3(\bar{n}) = \frac{C \cdot n_3\kappa_3}{n_3\kappa_3 + \sqrt{(n_1\kappa_1)^2 + (n_2\kappa_2)^2}}. \quad (8)$$

We see that this allocation rule is not quite DPS, and we wonder whether it coincides with the throughputs realized in our model and in NS2. Interestingly, even when $\mathrm{RTT}_1^0 \neq \mathrm{RTT}_2^0$, the above allocation indicates that the flows of type 1 enjoy the same throughput as the flows of type 2. In practice, however, one expects that this heterogeneity of round-trip times should have impact.

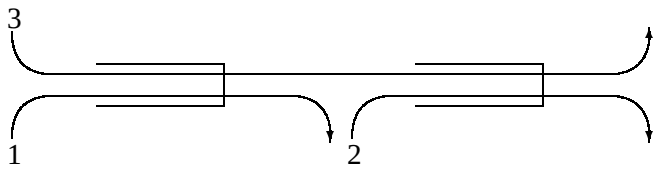

Figure 9: Topology of two-link example.

To study these effects we have performed two experiments. In the first we vary the heterogeneity between the round-trip times. This is done by choosing $\mathrm{RTT}_1^0 = 1$ and $\mathrm{RTT}_3^0 = 3$, and $\mathrm{RTT}_2^0 \in \{0.5, 1, 2, 5, 10\}$. We have fixed the numbers of users $n_1 = n_2 = n_3 = 10$, the buffer size $B_1 = B_2 = 10$ packets, and the link rates $C_1 = C_2 = 100$. For ease

we assume that no access rate limitations are imposed. The throughputs as derived by our model, as well as those based on (8), are tabulated in Table 2.A. We conclude that the heterogeneity does have a significant impact on throughputs; in particular, there could be a substantial difference between t_1 and t_2, which was not predicted by the rate allocation models of, e.g., [6]. Interestingly, class 3 considerably benefits when RTT_2^0 increases.

The second experiment studies the impact of the buffer size on the bias. We have varied $B = B_1 = B_2$, and fixed $\mathrm{RTT}_0^2 = 5$; the other parameters are the same as in the first experiment. Table 2.B shows the same 'equalizing effect' as we have seen before: for small buffers the bias is very pronounced, whereas for larger buffers, class 1 and class 2 see essentially the same round-trip delay (mainly queueing delay), and as a consequence the bias disappears.

Table 2: A. (left) Throughputs $t_i(\bar{n})$ as a function of RTT_2^0; the allocation according to (8) is given between parentheses. B. (right) Throughputs $t_i(\bar{n})$ as a function of B; the allocation according to (8) would be $t_1 = 75.4$, $t_2 = 75.4$, and $t_3 = 24.6$.

RTT_2^0	t_1		t_2		t_3	
0.5	72.4	(87.0)	77.1	(87.0)	12.2	(13.0)
1	68.3	(80.9)	68.3	(80.9)	16.7	(19.1)
2	65.6	(77.0)	60.7	(77.0)	19.8	(23.0)
5	64.3	(75.4)	53.0	(75.4)	21.3	(24.6)
10	64.0	(75.1)	48.1	(75.1)	21.6	(24.9)

B	t_1	t_2	t_3
2	43.9	22.3	13.7
5	56.9	39.9	18.5
10	64.2	52.9	21.3
20	68.4	62.0	24.6
50	69.9	67.2	26.5

6 Concluding remarks

We have developed a versatile TCP performance model that explicitly captures user heterogeneity. This multiple time-scale model integrates the dynamics at the packet level with those at the flow level. It is relatively simple, and allows for straightforward numerical evaluation. The accuracy of the model was validated by using TCP simulator NS2. In extensive numerical experiments, we have studied the impact of heterogeneity in the round-trip times on user-level characteristics such as throughputs and flow transmission times. Interestingly, we have seen that the asymmetry caused by heterogeneity in round-trip times is somewhat mitigated if the access rates are small (and, in addition, the load is relatively low), or the buffer is large.

We have pointed out how the single-link model can be extended in a straightforward way to the general network setting. It is noted that settings with heterogeneous users (in particular networks) pose interesting questions related to rate allocation. Like in the single-link case, we have seen that the size of the buffers does affect the rates allocated by TCP: again for large buffers the performance bias disappears. The network model opens up the possibility of a careful assessment of those phenomena; it is remarked that most

existing rate allocation results, see, e.g. [6], do not take into account buffering.

References

[1] E. Altman, C. Barakat, E. Laborde, P. Brown, and D. Collange. Fairness analysis of TCP/IP, Proceedings IEEE Conference on Decision and Control, pp. I: 61-66, 2000.

[2] E. Altman, T. Jiménez, and D. Kofman. DPS queues with stationary ergodic service times and the performance of TCP in overload, Proceedings INFOCOM 2004, 2004.

[3] K. Avrachenkov, U. Ayesta, E. Altman, P. Nain, and C. Barakat. The effect of the router buffer size on the TCP performance, Proceedings of the LONIIS International Seminar on Telecommunication Networks and Teletraffic Theory, pp. 116-121, 2002.

[4] F. Baccelli and D. Hong. Flow level simulation of large IP networks, Proceedings INFOCOM 2003, 2003.

[5] S. Ben Fredj, T. Bonald, A. Proutière, G. Régnié, and J.W. Roberts. Statistical bandwidth sharing: a study of congestion at flow level, Proceedings ACM SIGCOMM 2001, pp. 111-122, 2001.

[6] T. Bonald and L. Massoulié. Impact of fairness on Internet performance, Proceedings ACM SIGMETRICS 2001, pp. 82-91, 2001.

[7] P. Brown. Resource sharing of TCP connections with different round trip times, Proceedings INFOCOM 2000, pp. 1734-1741, 2000.

[8] T. Bu and D. Towsley. Fixed point approximation for TCP behavior in an AQM network, Proceedings ACM SIGMETRICS 2001, pp. 216-225, 2001.

[9] H.K. Choi and J.O. Limb. A behavioral model of web traffic, Proceedings of the Seventh Annual International Conference on Network Protocols, pp. 327-334, 1999.

[10] J.W. Cohen. The multitype phase service network with generalized processor sharing, Acta Informatica, Vol. 12, pp. 245-284, 1979.

[11] G. Fayolle, I. Mitrani, and R. Iasnogorodski. Sharing a processor among many classes, Journal of the ACM, Vol. 27, pp. 519-532, 1980.

[12] N. van Foreest, M. Mandjes, and W. Scheinhardt. A versatile model for asymmetric TCP sources, Proceedings ITC 18, pp. 631-640, 2003.

[13] N. van Foreest, B. Haverkort, M. Mandjes, and W. Scheinhardt. Versatile Markovian models for networks of asymmetric TCP sources. Submitted, 2004.

[14] R. Gibbens, S. Sargood, C. van Eijl, F. Kelly, H. Azmoodeh, R. Macfadyen, N. Macfadyen. Fixed-point models for the end-to-end performance analysis of IP networks, Proceedings 13th ITC Specialist Seminar: IP Traffic Measurement, Modeling, and Management, 2000.

[15] Homepage, Network Simulator II, `http://www.isi.edu/nsnam/ns/`

[16] F. Kelly. Blocking probabilities in large circuit-switched networks. Advances in Applied Probability, Vol. 18, pp. 473-505, 1986.

[17] F. Kelly. Mathematical modeling of the Internet, Mathematics Unlimited - 2001 and Beyond (Editors B. Engquist and W. Schmid), pp. 685-702, Springer-Verlag, Berlin, 2001.

[18] A. Kherani and A. Kumar. Stochastic models for throughput analysis of randomly arriving elastic flows in the Internet. Proceedings INFOCOM 2002, pp. 1014-1023, 2002.

[19] P. Kuusela, P. Lassila, J. Virtamo, and P. Key. Modeling RED with idealized TCP sources. Proceedings IFIP ATM & IP 2001, 2001.

[20] T.V. Lakshman and U. Madhow. The performance of TCP/IP for networks with high bandwidth-delay products and random loss. IEEE/ACM Transactions on Networking, Vol. 5, pp. 336-350, 1997.

[21] P. Lassila, H. van den Berg, M. Mandjes, and R. Kooij. An integrated packet/flow model for TCP performance analysis, Proceedings ITC 18, pp. 651-660, 2003.

[22] L. Massoulié and J.W. Roberts. Arguments in favour of admission control for TCP flows, Proceedings ITC 16, pp. 33-44, 1999.

[23] M. Mathis, J. Semke, J. Mahdavi, and T. Ott. The macroscopic behavior of the TCP congestion avoidance algorithm, Computer Communication Review, Vol. 27, pp. 67-82, 1997.

[24] V. Misra, W. Gong, and D. Towsley. Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED, Proceedings ACM SIGCOMM 2000, pp. 151-160, 2000.

[25] R. Núñez-Queija, H. van den Berg, and M. Mandjes. Performance evaluation of strategies for integration of elastic and stream traffic, Proceedings ITC 16, pp. 1039-1050, 1999.

[26] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose. Modeling TCP throughput: a simple model and its empirical validation, Proceedings ACM SIGCOMM 1998, pp. 303-314, 1998.

[27] J. Roberts, U. Mocci, and J. Virtamo. *Broadband network teletraffic; Final report of action COST 242*, Springer, Berlin, 1996.