Nonlinear state-dependent delay modeling and stability analysis of internet congestion control

C. Briat

H. Hjalmarsson, K.H. Johansson, U.T. Jönsson, G. Karlsson, H. Sandberg

Abstract— It is shown that the queuing delay involved in the congestion control algorithm is state-dependent and does not depend on the current time. Then, using an accurate formulation for buffers, networks with arbitrary topologies can be built. At equilibrium, our model reduces to the widely used setup by Paganini et al. Using this model, the delay-derivative is analyzed and it is proved that the delay time-derivative does not exceed one for the considered topologies. It is then shown that the considered congestion control algorithm globally stabilizes a delay-free single buffer network. Finally, using a specific linearization result for systems with state-dependent delays from Cooke and Huang, we show the local stability of the single bottleneck network.

I. INTRODUCTION

Network congestion is an important problem inducing packet loss and large communication delays, which limit the performance of the overall network. The Transfer Control Protocol (TCP) used in most network adapters/servers includes a congestion control mechanism but, despite of this, it is generally recognized that it will be one of the bottlenecks as network capacity and heterogeneity in traffic behavior are scaled up. This has spurred a wide variety of efforts to design improved congestion control algorithms, e.g. [1], [2], [3], [4].

Design and analysis of such algorithms require accurate models. Here there has been a transition from packet-level models [5] towards aggregated fluid flow models [2] which fit better into the control theoretic and optimization frameworks. Recently, there has been significant progress in integrating these two model types. In [6] it is shown that it is important to include the self-clocking property of current window based TCP, i.e. the fact that the rate of transmission is controlled, or 'clocked', by the received acknowledgements, and a new link model is proposed. This is further developed in [7], [8], [9] leading to a model that is able to capture, e.g. sub-RTT burstiness and link interactions with great accuracy. The resulting model is shown to be able to predict previously unknown network behavior, e.g. that inelastic cross-traffic can severely deteriorate the network response and that FAST TCP may become unstable in some situations when there are heterogeneous round-trip-times. See [10], [11] for details.

The first of our contributions is to continue the development of more accurate network models. In Section II we develop a new model for general networks using a delay-based protocol. It is shown that the delay operator acting on the measurement (the queuing delay value) is not the delay at time t but the delay at time before t. This points out a recurrent misconception in the literature and is one of the main contributions of the paper. Then, a more accurate model for FIFO buffers is proved allowing for an explicit expression of all the output flows separately. Such a model can already be found in the literature (see e.g. [12]) for simulating purpose only, in general without any theoretical justification. Finally, it is shown that using the general model it is, in principle, possible to describe any network topologies. When the network is at equilibrium, the model reduces to a standard model identical to the one in [13], suggesting that the proposed model is better disposed to describe the networks transient behavior.

The queuing delays imply that networked systems possess state dependent time-delays. This fact, together with nonlinear properties makes networked systems hard to analyze, restricting available results. We make several contributions in this respect. Firstly, Section III details an analysis of the particular time-delay involved in our new model. Indeed, thanks to the accurate buffer modeling, it is possible to provide an explicit expression for the queuing delays in the multiple bottleneck problem, allowing then for an analysis of the queuing delays in a more general setting than the single bottleneck problem. We show that, under general conditions, the delay derivative is lower than 1, permitting the use of many tools for time-delay systems available in the literature [14], [15].

Further, Section IV details some stability analysis results under various assumptions. The state dependent character makes stability analysis difficult. For example, existing stability results are local, where linearization is performed around the equilibrium time-delay. The latter has not been justified in the literature on stability of networked systems but by using a result in [16], we show that this is correct. Furthermore, we show that the FAST-TCP protocol globally asymptotically stabilizes the nonlinear delay-free single bottleneck problem. We also show that the single bottleneck problem with one user can be made locally asymptotically stable for any network configuration provided that the protocol parameters are chosen accordingly.

II. NETWORK MODEL

This section is devoted to the definition of a network model. The first sections are devoted to the presentation of the models used to described the different elements of the network, namely the buffers and the users protocol. A particular accent is put on the modeling of the queuing delay. Finally, a single-buffer network model with multiple users is given with a discussion on the extension to multiple buffers networks.

A. Buffer Model

A buffer with infinite capacity can be modeled as

$$\dot{\tau}(t) = \mathcal{Q}(\phi(t), \tau(t)) \tag{1}$$

where $\tau(\cdot)$ and $\phi(\cdot)$ are the queuing delay and the input flow respectively. The mapping $\mathcal{Q}$ is given by

$$\mathcal{Q}(\phi(t), \tau(t)) := \frac{1}{c}(\phi(t) - r(\phi(t), \tau(t))) \tag{2a}$$

$$r(\phi(t), \tau(t)) := \begin{cases} c & \text{if } \mathcal{C}(t) \\ \phi(t) & \text{otherwise} \end{cases} \tag{2b}$$

with condition $\mathcal{C}(t) = [\phi(t) \geq c \text{ or } \tau(t) > 0]$ and where $r(\cdot)$ and c denote the output flow and the maximal output capacity respectively. Note that the buffer model (1)-(2) is a differential equation with discontinuous right-hand side [17]. In the sequel, we will assume that the latter model is well-posed and admits solutions.

B. Protocol Model

We next derive a protocol model, which defines how the congestion control mechanism interacts with the buffer model above. Consider a user sending data over a single buffer. The data flow at time t experiences a queuing delay $\tau(t)$. At time $t + \tau(t)$, the data has been served by the buffer. We suppose that the data are then instantaneously acknowledged to the sender. Hence, we assume that there is no propagation delay and that the round-trip time (RTT) of the congestion control loop, *as seen by the sender at time t*, is equal to $\tau(t)$.
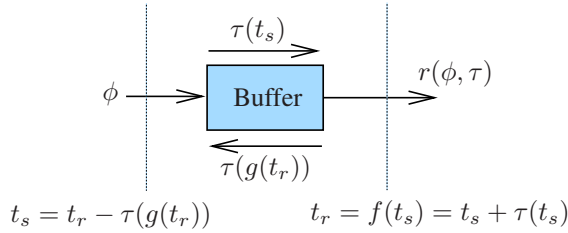

$$t_s = t_r - \tau(g(t_r)) \qquad t_r = f(t_s) = t_s + \tau(t_s)$$

Fig. 1.   Mappings illustration and reference times

Fig. 1 describes the relations between the sending time t_s and the reception time t_r through the map

$$f : t \to t + \tau(t) \tag{3}$$

allowing to compute the reception time t_r from the sending time t_s. This function is a forward operator, predicting the future from the present. Conversely, it will be shown that there also exists a mapping $g : t \to g(t)$ allowing for the computation of the sending time t_s from the reception time t_r. This operator is a backward one, allowing to retrieve the past from the present. Moreover, these operators are related to each other through the equalities

$$\begin{aligned} t_r &= t_s + \tau(t_s) & \text{(4a)} \\ &= g(t_r) + \tau(g(t_r)) & \text{(4b)} \\ t_s &= g(t_r) & \text{(4c)} \end{aligned}$$

as illustrated in Fig. 1. Using these relations, it can be deduced that the RTT satisfies

$$\begin{aligned} RTT &= t_r - t_s \\ &= \tau(t_s) \\ &= t_r - g(t_r) = \tau(g(t_r)) \end{aligned} \tag{5}$$

The interest of the operator $g(\cdot)$ lies in its ability to retrieve the time instant where the data entered the buffer from the knowledge of the acknowledgment reception time only.

The next lemma provides further results on f and g:

Lemma 2.1: The function g exists provided that the input flow $\phi(t) > 0$ for almost all $t \in \mathbb{R}_+$. It coincides with the inverse function of the mapping $f : t \to t + \tau(t)$ where τ evolves according to (1)-(2). Moreover, we have the following relations:

$$D^+[g] = \frac{1}{1 + D^+[\tau](g)} \tag{6a}$$

$$= \begin{cases} c/\phi(g) & \text{if } \mathcal{C}(g) \\ 1 & \text{otherwise} \end{cases} \tag{6b}$$

$$D^+[f] = \begin{cases} \phi/c & \text{if } \mathcal{C} \\ 1 & \text{otherwise} \end{cases} \tag{6c}$$

where $D^+[f](t) = \limsup_{h \downarrow 0} \dfrac{f(t + h) - f(t)}{h}$ is the upper right Dini derivative. $\qquad \square$

Proof: We will prove here Lemma 2.1 as well as relations (4). Since $t_r = f(t_s)$, if we seek a function g satisfying $t_s = g(t_r)$, this means that g corresponds to the inverse function of f. Since by definition t and $\tau(t)$ are continuous, the invertibility of f is equivalent to its monotonicity on $\mathbb{R}^+$. The function τ is differentiable almost everywhere, so the Dini derivative is considered instead of the usual derivative:

$$D^+[f] = \begin{cases} \phi/c & \text{if } \mathcal{C} \\ 1 & \text{otherwise} \end{cases} \tag{7}$$

Since $c > 0$ and $\phi \geq 0$ then $D^+[f] \geq 0$. Hence the only condition for f to be monotonic is $\phi \neq 0$ except, possibly, at a countable number of points. Under the weak assumption $\phi > 0$ almost everywhere (there exists a communication using the buffer) we can conclude on the global existence of the bijective inverse function $g = f^{-1}$ almost everywhere.

Equations (4a)-(4b) and (4c) are direct applications of the relations $f \circ g = g \circ f = \text{Id}$ where Id is the identity operator.

To prove (6b), let us consider equation (4b) evaluated at t. Applying the Dini derivative yields

$$1 = D^+[g] + D^+[g]D^+[\tau](g) \tag{8}$$

or equivalently

$$D^+[g] = \frac{1}{1 + D^+[\tau](g)} \tag{9}$$

Finally substitute (1) inside the latter equation to get (6b). ∎

Remark 2.1: The possibility of retrieving t_s from t_r may seem strange at first sight. Actually, this is an immediate consequence of the behavior of the passive FIFO buffer.
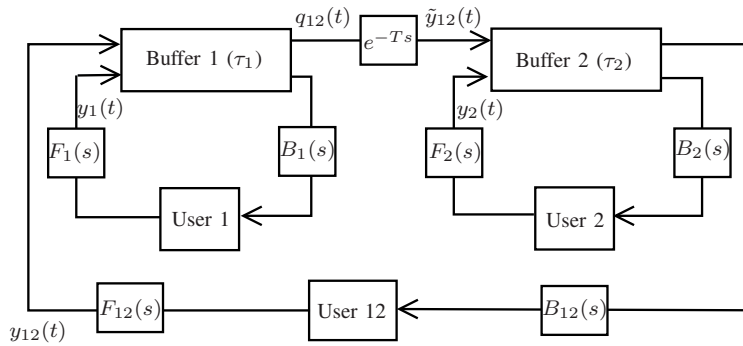
Fig. 2. Example of a multiple bottleneck network model

Each data is stored in order of arrival and this order is preserved when data are served. The preservation of the order is responsible of the monotonicity of f.

Lemma 2.2: The delay measured by the protocol at time t is given by $\tau(g(t))$. Moreover, the delay acting on the state τ is also equal to $\tau(g(t))$. $\square$

Proof: When an acknowledgment is received at time t_r the information on the RTT contained in the acknowledgment is $\tau(t_s)$ and is thus a delayed value of the queuing delay since $t_s \leq t_r$. Using (4c) and (4b) we can conclude that the measured queuing delay at time t_r is equal to $\tau(g(t_r)) = \tau(t_r - \tau(g(t_r)))$, thus normalizing the time as $t = t_r$ concludes the proof. ∎

Using the above results, the following generic model for a delay-based protocol can be given:

$$\begin{array}{rcl} \dot{z}_i(t) & = & \mathcal{P}_i(z_i(t), \tau(g(t))) \\ x_i(t) & = & h_i(z_i(t)) \end{array} \qquad (10)$$

where z_i, x_i are the state of the protocol and the number of outstanding packets for user i respectively. They are related through the output function h_i. The measured delay $\tau(g(t))$ has been defined in Lemma 2.2.

Remark 2.2: The model (10) is a backward model of the protocol where the reference of time is the reception time of ACK's. It is also possible to define a forward expression using the sending time as a reference. However, even if the latter is more intuitive, the former allows to analyze properties of the delay more easily.

Remark 2.3: The model (10) belongs to the class of nonlinear systems with state-dependent delays with few results on the stability in the control community [16], [18], [19], [20]. The analysis of such systems is reputed to be difficult due to the strong nonlinear nature of the state-dependent delay.

C. Single Bottleneck Network Model

Gathering the buffer model (1)-(2) and the protocol model (10), the closed-loop system with multiple users is governed by:

$$\begin{array}{rcl} \dot{\tau}(t) & = & \mathcal{Q}\left[\sum_i \phi_i(t), \tau(t)\right] \\ \phi_i(t) & = & \Phi\left[x_i\left(t - T_i^f\right), \tau(t) + T_i\right] \\ \dot{z}_i(t) & = & \mathcal{P}_i\left[z_i(t), \tau\left(t - T_i^b - \tau\left(g\left(t - T_i^b\right)\right)\right)\right] \\ x_i(t) & = & h_i(x_i(t)) \end{array} \qquad (11)$$

where T_i^f, T_i^b are respectively the forward and backward delay for user i and $T_i = T_i^f + T_i^b$. Since the protocol output is a number of outstanding packets to maintain in the network and the buffer input consists of flows, it is necessary to define an operator $\Phi(\cdot)$ converting the number of outstanding packets into flows. It has been shown in [6] that the following operator:

$$x_i(t) = \int_{g(t - T_i^b) - T_i^f}^{t} \phi_i(s) ds \qquad (12)$$

relates the number of outstanding x_i packets from a given flow ϕ_i. However, the integral operator (12) defines the flow for a given number of outstanding packets. Thus $\Phi(\cdot)$ coincides with the inverse operator of (12). It is not possible to describe it explicitly, thus approximations are generally used in the literature. Examples can be found in [6], [10] where it has been shown that the use of (12) provides accurate results in simulations and experiments.

D. Extension to Multiple Bottleneck Network Model

Let us consider the topology considered in Fig. 2 at the top of the page, where three users share two buffers. Users 1 and 2 use buffers 1 and 2 respectively while user 12 uses both of them. The blocks F_i and B_i, $i \in \{1, 2, 12\}$ denote respectively forward and backward constant propagation delays.

Suppose now we would like to derive a network model to describe this topology. It turns out that it is not possible to use the model (1)-(2) since the modeling does not allow for an explicit expression of the output flow q_{12} corresponding to the input flow y_{12} of buffer 1. Indeed, (1)-(2) only defines an aggregate output flow $r(\cdot, \cdot)$ for all input flows.

The problem consisting in finding an explicit formulation for the output flow $r(\phi, \tau)$ is solved in the following proposition:

Proposition 2.1: The output flow corresponding to input flow $\phi_i(t)$ in model (1) is given by

$$r_i(\phi, \tau) = \begin{cases} \dfrac{\phi_i(g(t))c}{\sum_j \phi_j(g(t))} & \text{if } \mathcal{C}(g(t)) \\ \phi_i(t) & \text{otherwise} \end{cases} \quad (13)$$

Moreover, the aggregate output flow is given by $r = \sum_i r_i$ and satisfies (2b). □

Proof: The starting point is equation (12). First, assume w.l.o.g. that propagation delays are neglected and define the initial time to be $g(t)$ with initial condition $\tau(g(t)) > 0$. Since the time needed for the buffer to serve the data initially contained at $g(t)$ is equal to the initial condition $\tau(g(t))$, this implies that the data sent by the user during the interval $[g(t), t]$ are exactly those stored in the buffer at time t. To analyze the output flow corresponding to a given input flow, we will virtually separate the contribution of each input flow to the value of the delay. This is possible using the superposition principle since the flows enter linearly in the buffer model (2a). So, defining the contribution of the flow i by τ_i we have:

$$\tau(t) = \sum_i \tau_i(t) \quad \text{with} \quad \tau_i(t) = \frac{1}{c} \int_{g(t)}^t \phi_i(s)ds \quad (14)$$

and, assuming a continuous flow, differentiation yields

$$\begin{aligned}
\dot{\tau}_i(t) &= \frac{1}{c}[\phi_i(t) - \phi_i(g(t))D^+[g](t)] \\
&= \begin{cases} \dfrac{1}{c}\phi_i(t) - \dfrac{\phi_i(g(t))}{\sum_j \phi_j(g(t))} & \text{if } \mathcal{C}(g(t)) \\ 0 & \text{otherwise} \end{cases}
\end{aligned} \quad (15)$$

We can see that the total output flow $r(\cdot)$ can be decomposed into the sum of atomic $r_i(\cdot)$ defined by (13). ■

Remark 2.4: Not surprisingly, the output flow (13) consists of a delayed and scaled version of the corresponding input flow. Indeed, the proportion of each input flow is preserved and the buffer hence plays the role of a delay operator on the proportion of flows, not on the flows themselves.

Remark 2.5: Using this model, it is in principle possible to build any network topology, such as the one in Fig. 2. Moreover, when the model is at equilibrium, its behavior can be well represented by an LTI system. Propagation delays can then commute with buffers and buffers can be put in parallel using appropriate forward and backward routing matrices. Such a transformation results in the analysis setup studied in [13].

III. DELAY CHARACTERIZATION

This section aims at studying important intrinsic properties of the delay. It will be shown that in the single bottleneck problem, the delay derivative cannot exceed one. However, this nice property may not be preserved when more general topologies are considered.

A. Queuing Delay Analysis

This section is devoted to the analysis of the characteristics of the queuing delay $\tau(g(t))$ in the single bottleneck problem, i.e. model (11). Note that the presence of the term $g(t)$

does not change the delay values range. However, its derivative will be strongly affected by this time-scale. It is well known that fast varying delays (whose derivative exceeds 1) are responsible of sudden loss of stability (and also violate causality [21]) compared to their time-invariant or slowly-varying counterparts [15], [22], [23]. A fast decreasing delay is not penalizing at all, so attention should be paid on the upper bound of the delay derivative only.

Since the queuing delay involved in the protocol (10) is equal to $\tau(g(t))$, the analysis of the delay derivative $d\tau(g(t))/dt$ is less straightforward. Such a derivative analyzes variations of the delay at sending times with respect to variations of the reception time.

Theorem 3.1: The delay derivative obeys

$$D^+[\tau(g)](t) = \begin{cases} 1 - \dfrac{c}{\phi(g(t))} & \text{if } \mathcal{C}(g(t)) \\ 0 & \text{otherwise} \end{cases} \quad (16)$$

and is bounded from above by one. □

Proof: The Dini derivative of $\tau(g(t))$ is given by

$$D^+[\tau(g)](t) = D^+[\tau](g(t)) \cdot D^+[g](t) \quad (17)$$

Using (6b), we get

$$D^+[\tau(g)](t) = \frac{D^+[\tau](g(t))}{1 + D^+[\tau](g)(t)} \quad (18)$$

Substituting the explicit expression of $D^+[\tau](g(t))$ given by (2) evaluated at $g(t)$ yields (16). ■

The delay derivatives $D^+[\tau](t_s)$ and $D^+[\tau(g)](t_r)$ are depicted in Fig. 3 where we can see the large difference between their behavior. Indeed, the delay derivative from the sending viewpoint $d\tau(t)/dt$ exceeds one when the input flow exceeds the double of the buffer maximal output capacity and hence can be made arbitrarily large. Moreover, $\tau(t)$ cannot decrease faster than a unitary slope.

On the other hand, the delay derivative viewed from the reception viewpoint $d\tau(g(t))/dt$ is bounded from above by one and is allowed to decrease arbitrarily fast. This illustrates the large qualitative difference between the delay behavior and shows that considering $d\tau(t)/dt$ instead of $d\tau(g(t))/dt$ is, in fact, very misleading.

B. Delay-derivative for Interconnected Buffers

In this section, we emphasize an important difficulty related to delay derivative in the multiple bottleneck problem. Indeed, in this case, there exist configurations for which the delay derivative can theoretically exceed one. To illustrate this, let us consider the Fig. 2 and focus on the price received by user 12. Users 1 and 2 are considered as unregulated cross-traffic (disturbances) and we ignore propagation delays. Using similar arguments as in Section III-A we get:

$$\begin{aligned}
t_r &= t_s + \tau_1(t_s) + \tau_2(t_s + \tau_1(t_s)) & (20a) \\
t_s &= g_1(g_2(t_r)) & (20b)
\end{aligned}$$

$$D^+[\tau_1(g_1(g_2))] = \begin{cases} \dfrac{(\phi_1(g_1(g_2)) - c_1)c_2}{\phi_1(g_1(g_2))y_2(g_2) + y_{12}(g_1(g_2))c_1} & \text{if } \mathcal{C}_1(g_1(g_2)) \text{ and } \mathcal{C}_2(g_2) \\ 1 - \dfrac{c_1}{\phi_1(g_1(g_2))} & \text{if } \mathcal{C}_1(g_1(g_2)) \text{ and not } \mathcal{C}_2(g_2) \\ 0 & \text{otherwise} \end{cases} \quad (19)$$
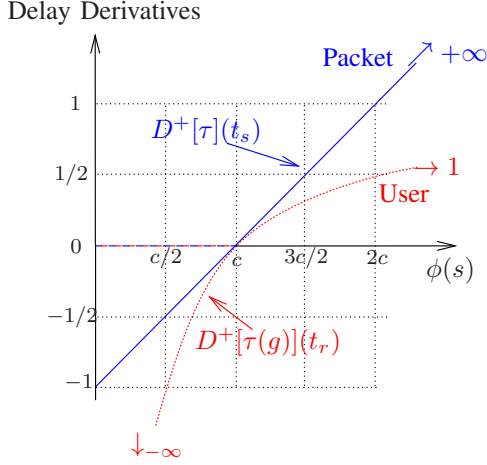

Fig. 3. Delay derivative viewed at sending (plain) and receiving (dashed).

where the g_i's are the inverse functions of $f_i : t \to t + \tau_i(t)$, $i = 1, 2$. Thus the RTT can be expressed as:

$$\begin{aligned} RTT &= \tau_1(t_s) + \tau_2(t_s + \tau_1(t_s)) \\ &= \tau_1(t_r - \tau_1(g_1(g_2(t_r)))) \\ &\quad + \tau_2(t_r - \tau_2(g_2(t_r))). \end{aligned} \quad (21)$$

The obtained closed-loop system will contain the two delayed terms. It has been shown in Section III-A that the derivative of $\tau_2(g_2(t))$ does not exceed 1. So, we focus on the delay $\tau_1(g_1(g_2))$ for which we have the following theorem:

Theorem 3.2: Consider Fig. 2. The derivative of $\tau_1(g_1(g_2))$ is governed by model (19) where $\mathcal{C}_i(\alpha) = [\phi_i(\alpha) > c_i \text{ or } \tau_i(\alpha) > 0]$. The aggregate input flows of buffers 1 and 2 are given by $\phi_1 = y_1 + y_{12}$ and $\phi_2 = \tilde{y}_{12} + y_2$ respectively with

$$\tilde{y}_{12}(g_2) = \begin{cases} \dfrac{y_{12}(g_1(g_2))c_1}{\phi_1(g_1(g_2))} & \text{if } \mathcal{C}_1(g_1(g_2)) \\ y_{12}(g_2) & \text{otherwise} \end{cases} \quad (22)$$

$\square$

Proof: The proof is given in Appendix A. ∎

Proposition 3.1: If $c_1 \geq c_2$, causality is always preserved. However when $c_2 > c_1$, with zero cross-traffic (i.e. $y_1 = y_2 = 0$), the causality is preserved if and only if the input flow y_{12} satisfies

$$y_{12} < c_1 + \frac{c_1^2}{c_2 - c_1}. \quad (23)$$

$\square$

Proof: This is an immediate conclusion drawn from the model (19). ∎

The latter result states that under some mild assumptions (the input flow y_{12} may exceed the output capacity c_1 by $c_1^2/(c_2 -$

c_1), the causality is preserved. However, when the difference $c_2 - c_1$ is large, the assumption becomes stronger and can be more easily violated. In such a case, according to some recent results, the notion of time-delay system collapses and the model might be no longer valid, but the network can still have a correct behavior. Due to space limitation, it is not possible to provide a detailed and general analysis of the delay behavior in presence of cross-traffic. Indeed, a general and relevant analysis is more involved and is left to a future paper.

IV. STABILITY ANALYSIS

This section is devoted to the stability analysis of the congestion control. It will be shown that the congestion control mechanism leads to a globally asymptotically stable network in the delay-free single bottleneck problem (Theorem 4.2). Then, using a linearization result for systems with state-dependent delays [16], local stability of the network is proved and stated through Theorem 4.3, Lemma 4.1 and Theorem 4.4. These latter results are the main results of this section on stability analysis.

A. Linearization of the nonlinear model

The nonlinear model (11) can be difficult to analyze depending on the choice of the flow operator Φ. To the authors' best knowledge no general global stability results exist for delay-based protocol (such as FAST-TCP [24]) taking into account all the involved phenomena. Some results can be found in [13], [25], [26], [6]. The main reason for these difficulties comes from the presence of nonlinearities, state-dependent delays, large number of heterogeneous constant delays, hybrid behavior and an arbitrarily large dimension; making the overall system analysis very intricate.

A usual simplification relies on the linearization of nonlinear model but since the system involves a state-dependent delay, the linearization seems to be a difficult task. Fortunately, the following fundamental result, first provided in [16], restated below, explains how the linearization can be performed.

Theorem 4.1 ([16]): Let us consider a continuous function $f(x, y)$ mapping $\mathbb{R}^n \times \mathbb{R}^n$ to $\mathbb{R}^n$. The nonlinear system

$$\dot{x}(t) = f(x(t), x(t - h(x(t)))) \quad (24)$$

with continuous state-dependent delay $h(x) \geq 0$ and equilibrium point x^* is locally asymptotically stable if the following LTI time-delay system

$$\dot{z}(t) = A_0 z(t) + A_1 z(t - h(x^*)) \quad (25)$$

is asymptotically stable with $A_0 = \dfrac{\partial f}{\partial x}(x^*, x^*)$ and $A_1 = \dfrac{\partial f}{\partial y}(x^*, x^*)$. $\qquad\square$

Surprisingly, no derivative of $h(\cdot)$ is involved. This means that the shape of $h(\cdot)$ around the equilibrium point has no influence on the local stability, only the delay value at equilibrium is important. So, this theorem turns an apparently difficult local stability analysis problem into a very simple one. Indeed, many results are available to study exactly the stability of LTI systems with delays [14], [15], [27]. The latter theorem justifies the commonly considered stability analysis problem [13] where the network is at equilibrium and the queuing delays are set constant.

B. Global Stability Analysis of the Delay-Free Model

In this subsection, we will consider the problem of stability analysis of the model (11) where $\Phi : x \rightarrow x/RTT$ [6]:

$$\dot{\tau}(t) = \frac{1}{c}\sum_i \frac{x_i(t - T_i^f)}{\tau(t) + T_i} + \delta(t) - 1 \qquad (26)$$

where $\delta(t) \in [0, 1)$ is the scaled cross-traffic. The second mode of the buffer is not recalled here but will be considered in the analysis. We will also consider the following model for the FAST-TCP protocol:

$$\dot{x}_i(t) = \gamma\left(-\frac{\tau(g(t - T_i^b))}{T_i + \tau(g(t - T_i^b))}x_i(t) + \alpha_i\right) \qquad (27)$$

where $\gamma, \alpha_i > 0$ are a tuning constant term and the reference of the protocol respectively. The total constant propagation delay for user i is denoted by $T_i = T_i^f + T_i^b$.

Theorem 4.2: The unique equilibrium

$$\tau^* = \frac{\alpha_s}{c(1 - \delta^*)}, \qquad x_i^* = \alpha_i\left(1 + \frac{T_i}{\tau^*}\right) \qquad (28)$$

of the nonlinear delay-free system

$$\begin{aligned}
\dot{\tau}(t) &= \frac{1}{c}\sum_i \frac{x_i(t)}{\tau(t) + T_i} + \delta(t) - 1 \\
\dot{x}_i(t) &= \gamma\left(-\frac{\tau(t)}{T_i + \tau(t)}x_i(t) + \alpha_i\right)
\end{aligned} \qquad (29)$$

is globally asymptotically stable for any $\alpha_i \geq 0$, $\alpha_s = \sum_i \alpha_i$, $c, \gamma > 0$ and equilibrium cross-traffic $\delta^* \in [0, 1)$. $\square$

Proof: The proof is given in Appendix B. $\blacksquare$

C. Local stability analysis in the single-user problem

1) Derivation of the Linear Model: Using Theorem 4.1 we can state that the linearized model of (26)-(27) around the equilibrium point (28) is given by

$$\dot{z}(t) = \mu\underbrace{\begin{bmatrix} -\xi_1 & 0 \\ \xi_2 & -\xi_3 \end{bmatrix}}_{A}z(t) + \mu^2\underbrace{\begin{bmatrix} -\psi_1 \\ 0 \end{bmatrix}}_{A_h}\nu(t - \tau^* - T)$$

$$(30)$$

where $z(t) = \mathrm{col}(y(t), \nu(t))$, $y = x - x^*$, $\nu = \tau - \tau^*$, $\mu = (\tau^* + T)^{-1}$, $\xi_1 = \gamma\tau^*$, $\xi_2 = c^{-1}$, $\xi_3 = \alpha(c\tau^*)^{-1}$ and $\psi_1 = \gamma T x^*$.

2) Delay Independent Stability of the Linear Model: This section is devoted to the analysis of the delay-independent stability of the system

$$\dot{z}(t) = Az(t) + A_h\nu(t - h) \qquad (31)$$

where $h \geq 0$ is an unknown constant delay, A and A_h are defined in (30).

The following result states a result on the delay independent stability of (31):

Theorem 4.3 (Delay-Independent Stability): The system (31) is delay independent stable if and only if the condition $T/\tau^* = c(1 - \delta^*)T/\alpha < 1$ holds. $\qquad\square$

Proof: Since the delay acts only on ν, it is thus possible to rewrite the system as a loop interconnection of the SISO transfer function $H(s) = C(sI - A)^{-1}A_h$ with $C = \begin{bmatrix} 0 & 1 \end{bmatrix}$ and the delay operator e^{-sh}. From [15], it is possible to conclude that the system is delay independent stable if and only if both A and $A + A_hC$ are Hurwitz and $||H||_{\mathcal{H}_\infty} < 1^1$. It is straightforward to show using Routh-Hurwitz criterion that both A and $A + A_hC$ are Hurwitz. Moreover, it is also easy to show that $||H||_{\mathcal{H}_\infty} = H(0) = T/\tau^*$. This concludes the proof. $\blacksquare$

From this result, we can conclude that as long as the queuing delay at equilibrium is larger than the propagation delay, the system (26)-(27) will be asymptotically stable. However, if this condition is not satisfied, then the system (31) becomes delay-dependent stable.

3) Delay Dependent Stability of the Linear Model: As discussed in the latter section, the system can be delay-dependent stable in some specific scenarios, i.e. there exists a positive scalar $h_{max} > 0$ such that the system (31) is unstable for all $h \geq h_{max}$. This is stated in the following lemma where a conservative (lower bound) on the critical delay value is provided:

Lemma 4.1 (Delay-Dependent Stability): The system (31) is delay-dependent stable for any constant delay $h < h_{max}$ with $h_{max} < \dfrac{\tau^*}{(1 - \delta^*)T} + \dfrac{1}{T\gamma}$ and for any values $c, \gamma > 0$, $\alpha \geq 0$ and $\delta^* \in [0, 1)$. $\qquad\square$

Proof: Indeed, since we know that the system is stable for $h = 0$ and $h = \infty$, this means that there exists a $h_{max}(x^*, \tau^*) > 0$ (depending on the linearization point) for which the system is stable for all constant $h \in [0, h_{max})$. To prove this, let us consider the polynomial $a(s, w) = \det(sI - A - A_hCw)$ and look for zero-crossings of the quasipolynomial $a(j\omega, e^{-j\omega h}) = 0$ which are clues for sign changes of zeros of $a(\cdot, \cdot)$. This yields

$$\begin{aligned}
-\omega^2 + \mu^2\xi_1\xi_3 + \mu^3\xi_2\psi_1\cos(\omega h) &= 0 \quad (32a) \\
\mu(\xi_1 + \xi_3)\omega - \mu^3\xi_2\psi_1\sin(\omega h) &= 0 \quad (32b)
\end{aligned}$$

after separation of the real and imaginary parts. Let us focus on (32b), which rewrites

$$\frac{\xi_1 + \xi_3}{h\mu^2\xi_2\psi_1} = \frac{\sin(\omega h)}{\omega h}, \quad \omega > 0.$$

---

[1]This is a necessary condition since SISO systems are involved. In general, these conditions are only sufficient.

Since the right-hand side is bounded from above by one, then by choosing a sufficiently small $h > 0$, the left-hand side can be made greater than one. In such a case, equality (32b) cannot be satisfied and hence there is no $\omega > 0$ such that (32) holds. Thus choosing a sufficiently small h prevents any zero-crossing and stability of the system is guaranteed.

It is sufficient to choose $h > 0$ which satisfies

$$\frac{\xi_1 + \xi_3}{\mu^2 h \xi_2 \psi_1} > 1$$

or equivalently:

$$h < \frac{\tau^*}{(1 - \delta^*)T} + \frac{1}{\gamma T}.$$

Thus the system will be stable for all $h \in [0, h_{max})$ with $h_{max} = \frac{\tau^*}{(1 - \delta^*)T} + \frac{1}{T\gamma}$. ∎

4) Local Stability of the Network: We have shown that the system (31) is delay-dependent stable for any values for the linearization point. Moreover, we have derived a conservative lower bound of the exact delay-margin. We can provide then the main result on stability analysis:

Theorem 4.4 (Network Local Stability): Any equilibrium point (x^*, τ^*) of the single-user nonlinear delay-system (26)-(27) is:

1) locally delay-independent stable if and only if $\tau^* > T$.
2) locally delay-dependent stable if $\tau^* < T$ and $\tau^*(T - 1) + T^2 \leq 0$.
3) locally delay-dependent stable if $\tau^* < T$, $\tau^*(T - 1) + T^2 > 0$ and

$$\gamma < \frac{1}{\tau^*(T - 1) + T^2}.$$

□

Proof: Since we know the exact delay value τ^* at each linearization point, it is sufficient to show that $\tau^* + T$ is contained in $[0, h_{max})$. We already know that if $\tau^* > T$ the system is stable, so the case $\tau^* \leq T$ remains to be studied and two cases may occur. The first one is when $\tau^*(T - 1) + T^2 \leq 0$ and in this case the h_{max} is always greater than $\tau^* + T$ independently of the value of γ. Finally, when $\tau^*(T - 1) + T^2 > 1$, then in order to have $h_{max} > \tau^* + T$, we need that $\gamma < \frac{1}{\tau^*(T - 1) + T^2}$. ∎

V. CONCLUSION

In this paper, a precise modeling for networks with multiple buffers has been provided with a particular accent on the queuing delays. It has been shown that the exact measured queuing delay is delayed by a function depending on the buffer state, and thus the network could be represented as a system with state-dependent delays. Moreover, this modeling has allowed for a precise analysis of the derivative of the queuing delay showing that, in the single bottleneck problem, this delay is always bounded from above by one. However, this property might not be satisfied for more complex topologies. A second part has been devoted to the stability analysis of a network using a FAST-TCP-like algorithm. It

has been shown, using a fundamental linearization result for systems with state-dependent delays from Cooke and Huang [16], that a single bottle-neck single-user network is locally asymptotically stable at any equilibrium point provided tuning parameters are chosen accordingly.

Future works will be devoted to a more detailed analysis of the queuing delays and on nonlinear stability analysis of the network in order to obtain global stability results or estimate the stability region.

APPENDIX

A. Proof of Theorem 3.2

The RTT equation (21) shows that two different delays are involved in the expression of the RTT, namely $\tau_2(g_2)$ and $\tau_1(g_1(g_2))$. The former has the same structure as the one analyzed in Section III-A, and we know that its derivative is bounded from above by 1, so let us focus on the second one. Its Dini derivative is given by

$$D^+[\tau_1(g_1(g_2))] = D^+[\tau_1(g_1)](g_2) \cdot D^+[g_2]. \tag{33}$$

Similarly as in Section III-A, the derivative of $D^+[\tau_1(g_1)](g_2)$ can be computed and is equal to $D^+[\tau_1](g_1(g_2)) \cdot D^+[g_1](g_2)$ or equivalently

$$D^+[\tau_1(g_1)](g_2) = \begin{cases} 1 - \dfrac{c_1}{\phi_1(g_1(g_2))} & \text{if } \mathcal{C}_1(g_1(g_2)) \\ 0 & \text{otherwise} \end{cases} \tag{34}$$

where $\phi_1(\cdot) = y_1(\cdot) + y_{12}(\cdot)$ is the total input flow of buffer 1. On the other hand, we have

$$D^+[g_2] = \begin{cases} \dfrac{c_2}{\phi_2(g_2)} & \text{if } \mathcal{C}_2(g_2) \\ 1 & \text{otherwise} \end{cases} \tag{35}$$

where $\phi_2(\cdot) = y_2(\cdot) + \tilde{y}_{12}(\cdot)$ is the total input flow of buffer 2. However, in order to take explicitly into account the flow circulating between the buffers, we need to provide an exact expression for $\tilde{y}_{12}$. Such an exact formulation has been provided in the Proposition 2.1 in Section II-D. Using this result, we get (22). Finally evaluating (33) using (34), (35) and (22) yields (19). This concludes the proof.

B. Proof of Theorem 4.2

For given $\alpha_i, c > 0$ and $\delta^* \in [0, 1)$, the unique equilibrium point can be determined by solving for the stationary solutions of (29). Shifting the model (29) around the equilibrium with the change of variable $y_i = x_i - x_i^*$, $\varepsilon = \delta - \delta^*$, $\nu = \tau - \tau^*$ yields

$$\begin{aligned} \dot{y}_i(t) &= -\gamma R_i(t) \left[(\nu(t) + \tau^*) y_i(t) + T_i x_i^* \nu(t) \right] \\ \dot{\nu}(t) &= \frac{1}{c} \sum_j R_j(t) y_j(t) - \frac{1}{c\tau^*} \sum_j R_j \alpha_j \nu(t) \\ R_i(t) &= (\nu(t) + \tau^* + T_i)^{-1} \end{aligned} \tag{36}$$

where we have set $\varepsilon = 0$. Note that this model is valid only when the buffer is non empty or when the input flow exceeds the effective maximal output capacity of the buffer. In the other case, the equation $\dot{\nu}(t) = 0$ must be used instead. We

will study the stability of the origin of the latter system. To this aim define the Lyapunov function

$$V(y, \nu) = \sum_i \frac{1}{2\gamma T_i x_i^*} p_\nu y_i^2 + \frac{c}{2} p_\nu \nu^2$$

where $p_\nu > 0$ is scalar. Computing the derivative of V along trajectories solution of (36) we get the following expression

$$\dot{V} = - \begin{bmatrix} y \\ \nu \end{bmatrix}^T \begin{bmatrix} p_\nu \, \mathrm{diag}[M_i(t)] & 0 \\ 0 & p_\nu \sum_i \alpha_i R_i / \tau^* \end{bmatrix} \begin{bmatrix} y \\ \nu \end{bmatrix} \tag{37}$$

with $M_i = (\nu(t) + \tau^*) R_i(t) / (x_i^* T_i)$. Thus when $\nu > -\tau^*$ the Lyapunov function is exponentially decreasing. However, this does not say anything when $\nu = -\tau^*$. Indeed, in such a case, we have $\dot{V}(t) = -p_\nu \sum_i \alpha_i \tau^* / T_i < 0$ which only says that V is strictly decreasing, but we have no exact information on how the states evolve. Moreover, when the input flow $\sum_i y_i$ is lower than $(1 - \delta^*)c$, the system is in the noncongested mode which is not taken into account in the inequality (37). In order to characterize the stability of the hybrid system over the whole state-space $(y, \nu) \in [-x^*, +\infty) \times [-\tau^*, +\infty)$, we will differentiate three cases:

1) $\sum_i y_i < (1-\delta^*)c - \sum_i x_i^*$ and $\nu = -\tau^*$ (empty buffer with input flow lower than the effective maximal output capacity $(1 - \delta^*)c$)
2) $\sum_i y_i \geq (1-\delta^*)c - \sum_i x_i^*$ and $\nu = -\tau^*$ (empty buffer with input flow greater than the effective maximal output capacity $(1 - \delta^*)c$)
3) $\nu > -\tau^*$ (non empty buffer); the exponential stability has been proved above.

Case 1: This case describes the case when the aggregate flow is small (lower than the maximal output capacity of the buffer) and the buffer is empty. This configuration typically could arise when a network is 'starting'. Considering the network equations with initial conditions $\nu(t_0) = -\tau^*$ and $\sum_i (y_i(t_0) + x_i^*) < (1 - \delta^*)c$, we get

$$\begin{array}{rcl} \dot{y}(t) & = & \gamma x^* \tau^* > 0 \\ \dot{\nu}(t) & = & 0 \end{array} \tag{38}$$

for $t \geq t_0$. Thus after some time $t_1 > t_0$, we have $\sum_i [y_i(t_1) + x_i^*] = (1 - \delta^*)c$ and we switch to case 2.
Case 2: The model is given by (36) with initial conditions $(y(t_1), \nu(t_1)) = (y(t_1), -\tau^*)$ for $t \geq t_1$. Note that $\nu(t_1) = -\tau^*$, so it cannot decrease more and note also that $y_i(t) > y_i(t_1)$ for all $t > t_1$ by virtue of the first equation $\dot{y}_i(t) = \gamma x_i^* \tau^*$. Since the contribution of the y_i to the ν dynamics is positive (i.e. $\dot{\nu}$ increases when y_i increases), then it will exist a time $t_2 \geq t_1$ for which $\dot{\nu}(t_2) > 0$ and we switch to case 3 for which stability has been proved. This concludes the proof.

References

[1] L. Brakmo and L. Peterson, "TCP Vegas: end-to-end congestion avoidance on a global Internet," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 8, pp. 1465–1480, 1995.
[2] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, pp. 237–252, 1998.
[3] C. Jin, D. X. Wei, and S. H. Low, "FAST TCP: motivation, architecture, algorithms, performance," in *Proceedings of IEEE Infocom*. IEEE, March 2004.
[4] F. Paganini, Z. Wang, J. C. Doyle, and S. Low, "Congestion control for high performance, stability, and fairness in general networks," *IEEE/ACM Transactions on Networking*, vol. 13, no. 1, pp. 43–56, 2005.
[5] V. Jacobson, "Congestion avoidance and control," *SIGCOMM Comput. Commun. Rev.*, vol. 18, no. 4, pp. 314–329, 1988.
[6] K. Jacobsson, L. Andrew, A. Tang, S. Low, and H. Hjalmarsson, "An improved link model for window flow control and its application to FAST TCP," *IEEE Transactions on Automatic Control*, vol. 54(3), pp. 551–564, 2009.
[7] A. Tang, L. Andrew, K. Jacobsson, K. Johansson, S. Low, and H. Hjalmarsson, "Window flow control: Macroscopic properties from microscopic factors," in *27th IEEE Conference on Computer Communications (INFOCOM 2008)*, Phoenix, Ar, 2008, pp. 538–546.
[8] K. Jacobsson, L. Andrew, A. Tang, K. Johansson, H. Hjalmarsson, and S. Low, "ACK-clock dynamics: Modeling the interaction between ack-clock and network," in *27th IEEE Conference on Computer Communications (INFOCOM 2008)*, Phoenix, Ar, 2008, pp. 181–185.
[9] L. Andrew, A. Tang, K. Jacobsson, K. Johansson, H. Hjalmarsson, and S. Low, "Queue dynamics with window flow control," *IEEE/ACM Transaction on Networking*, 2010, to appear.
[10] K. Jacobsson, "Dynamic modeling of internet congestion control," Ph.D. dissertation, KTH School of Electrical Engineering, 2008.
[11] N. Möller, "Window-based congestion control," Doctoral thesis, KTH, Stockholm, Sweden, 2008.
[12] D. Ros and R. Marie, "Loss characterization in high-speed networks through simulation of fluid models," *Telecommunication Systems*, vol. 16(1-2), pp. 73–101, 2001.
[13] F. Pagaganini, J. Doyle, and S. Low, "A control theoretic look at internet congestion control," *in Multidisciplinary Research in Control: The Mohammed Dahleh Symposium 2002. , Eds. L. Giarre' and B. Bamieh, Lecture Notes in Control and Information Sciences*, vol. 289, 2003.
[14] S.-I. Niculescu, *Delay effects on stability. A robust control approach*. Springer-Verlag: Heidelbeg, 2001, vol. 269.
[15] K. Gu, V. Kharitonov, and J. Chen, *Stability of Time-Delay Systems*. Birkhäuser, 2003.
[16] K. L. Cooke and W. Huang, "On the problem of linearization for state-dependent delay differential equations," *Proceedings on the American Mathematical Society*, vol. 124(5), pp. 1417–1426, 1996.
[17] A. F. Filippov, *Differential Equations with Discontinuous Righthand Sides*. Kluwer Academic Publishers, 1988.
[18] E. Verriest, "Stability of systems with state-dependent and random delays," *IMA Journal of Mathematical Control and Information*, vol. 19(1-2), pp. 103–114, 2002.
[19] H.-O. Walther, "A periodic solution of a differential equation with state-dependent delay," *Journal of Differentials Equations*, vol. 244, 2008.
[20] I. Györi and F. Hartung, "Exponential stability of a state-dependent delay system," *Disrete and Continuous Dynamical Systems*, vol. 18(4), 2007.
[21] E. Verriest, "Causal behavior of switched delay systems as multi-mode multi-dimensional systems," in *8th IFAC International Symposium on Time-Delay Systems*, Sinaia, Romania, 2009.
[22] V. Kharitonov and S. Niculescu, "On the stability of systems with uncertain delay," *IEEE Transactions on Automatic Control*, vol. 48, pp. 127–132, 2003.
[23] A. Papachristodoulou, M. M. Peet, and S.-I.Niculescu, "Stability analysis of linear systems with time-varying delays: Delay uncertainty and quenching," in *46th Conference on Decision and Control*, New Orleans, LA, USA, 2007, 2007.
[24] D. Wei, C. Jin, S. Low, and S. Hedge, "FAST TCP: Motivation, algorithms, performance," *IEEE/ACM Transactions on Networking*, vol. 14(6), pp. 1246–1259, 2006.
[25] J.-Y. Choi, K. Koo, D. Xei, and S. Low, "Global exponential stability of FAST TCP," in *INFOCOM 2005*, Miami, Florida, USA, 2005.
[26] I. Lestas and G. Vinnicombe, "Scalable robust stability for nonsymmetric heterogeneous networks," *Automatica*, vol. 43, pp. 714–723, 2007.
[27] W. Michiels and S. Niculescu, *Stability and stabilization of time-delay systems. An eigenvalue based approach*. Philadelphia, USA: SIAM Publication, 2007.