Proceedings of the 32nd Conference
on Decision and Control
San Antonio, Texas • December 1993

AN APPROACH TO THE DESIGN OF HIGH SPEED NETWORKS FOR BURSTY TRAFFIC

V. Anantharam

School of Electrical Engineering,
332, Engineering Theory Center,
Cornell University, Ithaca, NY 14853.
ananth@camus.ee.cornell.edu

Abstract

The class of message flows that satisfy the condition that the amount of traffic in an interval is upper bounded by an affine function of the length of the interval has been found to be a useful class of models for traffic on internal links in networks that have to handle bursty traffic. We make the key observation that there is a simple recursively updatable number, which we call the *virtual backlog*, which plays the role of a state for these models, in that it completely summarizes the information gained at any time by having observed the flow and can be recursively updated. This allows the formulation of control and resource allocation problems at nodes within the network as a stochastic game.

1. Introduction

Network designers would soon like to have in place networks that can offer a wide variety of services, such as audio, video, data, etc. using a common protocol suite. The perceived advantages of this concept include the possibility of offering new kinds of services that include several of the individual services one is familiar with as components. Several novel design issues have been thrown up as a consequence of this drive.

In order to facilitate the evolution of the integrated network, standards committees have by and large agreed on a set of standards described by the acronym ATM/SONET. This stands for Asynchronous Transfer Mode/Synchronous Optical Network. SONET is a standard for the physical layer of the network, while ATM describes the data link protocols and some network layer issues. In ATM traffic is packetized into fixed length packets called *cells*. When a connection is established it is assigned a geographical route and the cells associated to this connection are sent along the route. The advantage of establishing standards is that it permits several vendors to enter the market, with the assurance that their products will be compatible with those of other vendors.

A number of design issues at the elements are still left open by the standards. These include issues of time slot allocation and buffer allocation to competing flows at the network elements. This suggests that some intelligence can be used in handling these issues while still remaining compatible with the standards. The methodology developed in this paper is offered as one route to optimize such designs.

To motivate the point of view of this paper, let us briefly describe the leaky bucket flow control scheme used to regulate bursty traffic before admission to the network and for policing. In this scheme, cells associated to a connection are regulated by means of a stream of tokens generated at a constant rate with constant interarrival times. The tokens collect in a token buffer, and the cells collect in a cell buffer. A cell is released only when there is a token available, in which case it consumes one token. Token arriving when the token buffer is full are lost, as are cells arriving when the cell buffer is full (at least this is the case in the simplest version of the protocol, although more involved schemes involving tagging cells are being proposed).

The leaky bucket scheme has been found to be very effective in practice, and some version of it is likely to be eventually adopted in the next generation of networks. We wish to observe that the scheme regulates offered traffic so that it satisfies *burstiness constraints* which take the form that the amount of traffic over a time interval is bounded by an affine function of the length of the interval. The class of traffic flows that satisfy such burstiness constraints seems therefore to be an interesting one to consider. In addition to the fact that flows of this kind can be easily generated by the leaky bucket scheme, such traffic is described by simple parameters that can in principle be negotiated between users and the network.

Motivated by these considerations, several recent works have studied the use of burstiness constrained flows in the analysis and design of networks. As a rule, these works adopt the following point of view : Assuming that the input flows to a network element are burstiness constrained, determine the burstiness constraints on the output flows in terms of those of the input flows and the parameters and policy at the network elements. This then allows an analysis of networks with fixed policies and suggests how to choose parameters such as buffer dimension to achieve the desired performance. For purposes of design, it is suggested that one could carry out the estimates separately for the design alternatives being considered. If one finds that the lower bounds for one design alternative are worse than the upper bounds for the other, this yields a clear cut choice between the alternatives.

In this paper, we make the key observation is that it is possible to characterize the past of a burstiness constrained flow in terms of a simple recursively updatable statistic which we call the *virtual backlog*. This allows us to formulate the problem of designing adaptive control strategies at network elements fed by burstiness constrained flows in a systematic manner using the theory of stochastic games. Our proposal is to integrate this into an overall network design philosophy that involves the network elements exchanging the basic burstiness parameters of the flows they are handling on a slower time scale.

We will use the term *message flow* to denote a sequence of nonnegative real numbers $(a_n, n \geq 0)$. We model traffic on the links of a network by a message flow.

Definition 1

A message flow $(a_n, n \geq 0)$ is said to be (σ, ρ) constrained, if for all $0 \leq n_0 \leq n_1 < \infty$, we have

$$\sum_{n_0}^{n_1} a_k \leq \sigma + \rho(n_1 - n_0 + 1) . \quad (1)$$

2. Virtual Backlog

In this section we discuss the issue of how to represent the information gained by having observed the past of a (σ, ρ) constrained flow. It turns out that this question has a rather simple answer.

Definition 2

Let $(a_n, n \geq 0)$ be a (σ, ρ) constrained flow. We say that it has initial ***virtual backlog*** σ_0, if in addition to (1) the flow obeys the constraints

$$\sum_{0}^{n} a_k \leq \sigma_0 + \rho(n+1)$$

for all $n \geq 0$.

The key observation we make is the following :

Lemma 1

Let $(a_n, n \geq 0)$ be a (σ, ρ) constrained flow with initial virtual backlog σ_0. Suppose that a_0 is revealed. Then the information gained about $(a_n, n \geq 1)$ is exactly summarized by the statement that $(a_n, n \geq 1)$ is a (σ, ρ) constrained flow with initial virtual backlog σ_1, where

$$\sigma_1 = \min(\sigma_0 + \rho - a_0, \sigma)$$

There are two parts to this statement. The interesting part is that *every* (σ, ρ) flow with initial virtual backlog σ_1 is consistent with being the portion $(a_n, n \geq 1)$ of a flow $(a_n, n \geq 0)$ that is known to be (σ, ρ) constrained with initial virtual backlog σ_0 and has initial flow a_0.

3. Network control and stochastic games

We visualize a network element as being fed by a number of burstiness constrained flows and as implementing actions based on past information. We assume that the ***state*** of the network element at time n, $n = 0, 1, \ldots$ is given by an element $\xi_n \in \Xi$, where Ξ is a set called the ***element state space***. At time n the network element is to choose a ***control action*** $u_n \in U$ where U is a set called the set of ***control actions***. For simplicity we have assumed that the set of possible control actions at each time is the same. This can easily be generalized. The evolution of the state of the network element is assumed to occur in response to the incoming flows at the current time and the choice of control action. In general one has the abstract evolution equation

$$\xi_{n+1} = f(\xi_n, u_n, \underline{a}_n)$$

where we have used the underlined notation for the message flows to indicate that there may be more than one driving message flow. Note that when there are K driving message flows with the the i th flow, being (σ^i, ρ^i) constrained, $1 \leq i \leq K$, the domain of f is $\Xi \times U \times \prod_{i=1}^{K} [0, \sigma^i + \rho^i]$.

Let $0 < \beta < 1$ be a discount factor. We formulate the problem of the controller as one of minimizing the total infinite horizon discounted cost

$$\sum_{n=0}^{\infty} \beta^n c(\xi_n, u_n, \underline{a}_n) .$$

We interpret the cost $c(\xi_n, u_n, \underline{a}_n)$ incurred at time n as an amount ***paid by the controller to the driving sources.*** In other words we adopt the formulation of the problem as a ***zero-sum stochastic game***, played between the controller and the driving sources. Even though the sources are treated as one player having an active control role in this formulation it is easily seen that they are actually passive and the formulation is equivalent to a worst case design. Under simple and widely applicable technical conditions, we can prove the existence a value and of a stationary optimal strategy for the controller given by a Shapley recursion. This optimal control strategy can ***in principle be implemented in real time.*** Indeed, the entire history of each burstiness constrained flow can be kept track of by its ***recursively updatable*** virtual backlog, which is a simple finite dimensional statistic. In practice of course, one would expect that the basic burstiness parameters should be updated on a slower time scale, but this is a topic for future work.