Towards Protocol Equilibrium with Oblivious Routers

Debojyoti Dutta
USC/ISI
4676 Admiralty Way
Marina Del Rey, CA 90292
Email: ddutta@isi.edu

Ashish Goel
Dept. of MS&E
Stanford University
Terman 311, Stanford CA 94305-4026
Email: ashishg@stanford.edu

John Heidemann
USC/ISI
4676 Admiralty Way
Marina Del Rey, CA 90292
Email: johnh@isi.edu

Abstract— **Currently, the Internet is dominated by TCP traffic. TCP is congestion aware, shares bandwidth with other TCP flows, and is stable because most flows are congestion reactive. It has been shown that current AQM schemes may not be resistant to greedy traffic agents. Thus, it is important to study mechanisms which provide incentives to greedy agents to come to an equilibrium state in their own selfish interest. In addition, we want our AQM schemes to be oblivious to the flows' identities which makes them easier to scale and deploy. In this paper, we show that if routers used EWMA to measure the aggregate rate, then the best strategy for a selfish agent to minimize its losses is to arrive at a constant rate. Even though the protocol space is arbitrary, our scheme ensures that the best greedy strategy is simple, i.e. send with CBR. Then, we show how we can use the results of an earlier paper to enforce simple and efficient protocol equilibria on selfish traffic agents.**

I. INTRODUCTION

Congestion control and active queue management (AQM) are two important components that lead to effective performance in the network. An AQM scheme, such as RED [7], queues, schedules, and drops or marks packets according to a specific policy or algorithm. A congestion control protocol, such as TCP [24], operates at the end-points, and uses the drops or marks received from the AQM policies at routers as feedback signals to adaptively modify the sending rate in order to maximize its own goodput. For example, protocols and mechanisms such as TCP [24], [9] (congestion control) and Drop-tail and RED [7] (AQM) respectively have been widely implemented and/or deployed. We classify AQM schemes into two groups: *oblivious* or stateless and stateful. Oblivious AQM policies do not keep per-flow state information, and, hence, cannot perform differential dropping or scheduling for different flows. In contrast, stateful schemes such as fair queueing [2] offer good performance on a variety of metrics. Oblivious schemes such as drop-tail are more scalable and much easier to implement and deploy.

TCP [24], [23] is the dominating transport layer protocol in the Internet and accounts for over 90% of the total traffic [29]. The TCP protocol is characterized by a well defined, feedback based additive increase and multiplicative decrease (AIMD) congestion control algorithm that is well studied, quite robust in practice, and can share bandwidth equally with other TCP flows that have similar round trip times [20]. It is widely believed that if all users deployed TCP, networks will rarely see congestion collapses and the overall utilization of the network will be high. However, there are definite indications that the amount of non-congestion-reactive traffic is on the rise [6]. Most of this misbehaving traffic does not use TCP e.g. Realmedia, network games, and several other real time multimedia applications. If an agent is selfish and its sole aim is to maximize its own benefit, it is not bound to use TCP or any other well known transport protocol for its applications. Its protocol behavior may then be more aggressive, and, hence, severely degrade the performance experienced by the other, well-behaved agents it shares resources with. Even worse, if all users are greedy, they might send at ever-increasing rates to garner a greater share of bandwidth. In fact, in an earlier paper [3], we demonstrate that both Drop-tail and RED do not impose Nash equilibria in the presence of greedy traffic. Thus, it is increasingly important for AQM schemes to be robust against greed.

The problem we study is to design oblivious AQM schemes for which a simple *Protocol Equilibrium* exists, i.e. there is a simple protocol such that if every user follows this protocol, then the network behaves *efficiently* (i.e. with high link utilization and bounded drop probability), and there is no incentive for any user to deviate from the protocol. We would like the equilibrium protocol to be as simple as possible without putting any restrictions on the protocol space of the agents, i.e. a user is allowed to choose arbitrarily complex protocols but we would like to provide incentives to users so that they choose simple protocols that lead to efficient network performance.

We study the case of a single bottleneck link. We use $f_i(t)$ to denote the arrival rate of an agent i. The agents have full knowledge of the network conditions and $f_i(t)$ is only revealed at time t. We will refer to $f_i(t)$ as a protocol [1]. The agent i is allowed to use the entire history of the system till time t to determine $f_i(t)$. A protocol equilibrium is essentially a Nash

---

[1]A congestion control protocol typically refers to the algorithm used to adapt the sending rate in response to a feedback from the system, such as packet losses. But we assume that the users are omniscient, and, hence, providing feedback from the network to the users is redundant. In this idealized setting, a protocol can be thought of as merely the sending rate of a user.

equilibrium in a very large space of all possible protocols of these agents. Recall that a Nash equilibrium is an important solution concept in Game theory [19]. Like any game, our game has rules and players. The rules are set by the AQM strategies and the players are the selfish agents each following a protocol $f_i(t)$.

With stateful AQM schemes such as fair queueing, protocol equilibrium is very easy to obtain. However, such schemes may be hard to scale and deploy. In this paper, we explore the limits of what can be achieved using oblivious AQM schemes that are stateless and easily deployable. Recall that the oblivious routers can only look at aggregate traffic and are not allowed per-flow state. Also, we do not restrict the agents' traffic arrival pattern in any way.

In our game, it would be infeasible for a router to store the entire history since the sending rates $f_i(t)$ may vary arbitrarily with time. Thus, we restrict ourselves to routers which merely maintain an Exponentially Weighted Moving Average (EWMA) of the aggregate arrival rate $f(t) = \sum_i f_i(t)$. The router strategy can be summarized by a single function $p : R^+ \rightarrow [0,1]$ where $p(R(t))$ denotes the fraction of incoming traffic dropped by the AQM at the router, where $R(t) = EWMA(f(t))$. We assume infinite buffers. There is no simple reason to believe that such a simple strategy can lead to a protocol equilibrium. However, in this paper, we show that this is indeed the case. We find functions f that lead to protocol equilibria. We show that $f_i(t)$ is a constant, p is low and bounded, and link utilization is high.

A. Our Results

In this section, we summarize our main results.

- In a single player game, where a player has a finite number of packets to send in a finite time window, routers measure rate using EWMA, and the drop probability is linear in the EWMA, the best strategy for the agent to minimize its loss is to equispace its packet arrivals.
- In a two player game where each player declares its entire arrival sequence apriori, the routers measure rate using EWMA, and the drop probability is linear in the EWMA, if one player arrives at regular intervals with an initial random shift, the other player essentially plays a game with itself. It is now easy to show that there exists a rate such that each player arriving at that rate (with an initial random shift) is a mixed strategy equilibrium for the game. Since the entire arrival sequence is defined in the beginning, we call the resultant equilibrium *Apriori Equilibrium*. It is easy show that this game does not have a deterministic (pure) strategy equilibrium.
- **Main Result:** We show using a fluid approximation of traffic and a large class of arrival functions, that arriving with a CBR is optimal for a single user. The proof uses a fluid model of traffic and calculus of variations. Then, we show that if all $N - 1$ agents arrive with CBR, the best response for the Nth agent is also to arrive with a CBR.

- Now, the protocol space has been reduced to a choice of rates. We can now invoke results from our earlier work [3] and show that there exists an oblivious AQM scheme which combined with EWMA leads to an efficient equilibrium. At this equilibrium, each agent sends at the same constant bit rate.

Observe, that CBR is a consequence of the EWMA and our AQM, and not a requirement. While our proofs are quite involved, the overall motivation is quite easy to grasp. Given an EWMA, the best strategy for an agent is to arrive at uniform intervals in the discrete case and with CBR in the continuous case. As stated earlier, this is desirable since we want the protocol space to be arbitrary but the equilibrium strategy to be simple. Note that the agents must arrive with a CBR in their own interest. They cannot improve their performance by making arbitrarily fine-grained decisions. Also note that the router is doing a minuscule amount of work.

B. Related Work

The work related to protocol equilibria can be categorized into two major groups. The systems community has investigated different AQM schemes that impose some degree of fairness on misbehaving flows. The theoretical work has addressed this question by constructing models for selfish user behavior and analyzing them with tools such as Game theory. In this section, we first present a short survey of the AQM literature followed by a slightly longer survey of the game theoretic approaches.

One can group AQM policies into stateless, stateful, and those in between the above two. The best examples of stateless policies that have been widely implemented and/or deployed are Drop-tail and RED [7]. At the other end of the spectrum is fair queueing [2]. In between the two are various schemes that use some amount of state. For example, FRED [16] keeps track of active flows. CSFQ [28] ensures that the bandwidth allocation in core routers is done in a stateless and an approximately max-min fair manner while the edge routers need to measure the rate of flows. RED-PD [17] detects the top few misbehaving flows, keeps track of them using minimal state and punishes those flows. In [18], [4], the authors propose scalable algorithms for identifying large flows using techniques such as sample and hold, and multistage filters.

Game theory [19] is a very mature topic. The current challenges of game theory applied to computer networks are summarized by Papadimitrou [21]. Several papers (for example [22], [14], [13], [27], [1], [26], [25], [15], [10], [5], [15]) have applied tools from microeconomics and game theory to computer networks over the last fifteen years. A thorough literature survey is beyond the scope of the paper. We now consider some of the most directly relevant related work and compare them with our approach.

In [26], Roughgarden et. al. study the effect of degradation of service due to selfish routing. However, they do not study the mechanisms to reach equilibria. We, on the other hand,

assume fixed routes and look at the strategies where the user is free to choose its arrivals. Also, our utility function is different from theirs. Korilis et. al. [13] study the existence of equilibria. However, they do not show how to get to equilibrium. In [14], the authors study equilibria of a routing game over parallel links. Our work is different from the above papers because, unlike them, we present simple AQM schemes to impose protocol equilibrium on selfish agents.

Shenker [27] defines the Internet game from a switch scheduling perspective and proves that with Markovian arrival rates, the *fair share* allocation scheme is the only scheme within a class of buffer allocation functions (called MAC) that can guarantee a Nash equilibrium (on selfish agents), and is also Pareto efficient. We, on the other hand, study equilibria imposed by oblivious AQM schemes with very general drop functions and arrival processes. In fact, our mechanisms do not fall into the class MAC. Several other papers have investigated related problems. For example, Park et. al. [22] study Nash equilibrium properties of the QoS game. However their utility functions are much different from ours due to their multilevel service model and threshold based step functions for modeling utility. They claim that for a single service level, their model boils down to that in [27]. Gibbens et. al [8] have studied the effect of selfish users in the context of the user optimization problem defined in [12]. Their model assumes that the selfishness arises due to the user's disregard for the effect of its own action on prices. They model the Internet game differently using a AIMD like protocol with the congestion marks from the routers. Thus, our assumptions and our problem is different from theirs. Note that we have not looked at feedback signals from the network and pricing issues in our work.

Very recently, Johari et.al. [11] have shown the existence and the uniqueness of Nash equilibria for a network game with a single congested link defined as follows: each player sends its bid to a network manager and gets bandwidth in return. It maximizes its own utility which is the difference of a concave function of its own bid as well the aggregate bid and its own bid. They show that the *price of anarchy* for the above game is at most $\frac{1}{4}$, i.e. the normalized utility lost because of selfish users trying to game the system is at most 25%. Their results require a stateful *manager* or the router while we assume that routers do not keep per-flow state. Unlike them, we do not assume any pricing mechanism, and do not use a virtual currency. In our game, the utility is the goodput.

Akkela et. al. [1] have modeled greedy agents using TCP-like AIMD algorithms. They show that RED does not have a Nash equilibria using empirical models and through simulation. Also they have a restricted notion of selfish traffic. Unlike them, we consider *oblivious AQM techniques*, consider a wider class of traffic arrivals and drop functions, and show how one can reach the equilibrium state.

C. Practical Considerations

We believe that a practical solution to greed is important. However, before our ideas can be used to build real systems, two issues need to be resolved.

1) **Feedback:** In our version of the protocol equilibrium, agents are omniscient. But for the scheme to be practical, the state of the network needs to be be sent to end-point agents in form of feedback. Note that we wish to send this feedback without maintaining per-flow state at the routers.

2) **Tragedy of the commons:** It is easy to show that a malicious agent can disrupt our scheme by sending at a modestly higher rate than the equilibrium rate. The malicious agent would suffer but so would everyone else. Thus, we need to weed out the malicious flows out of the system. We would like to design a low-state architecture that can filter or dampen the non responsive malicious flows.

The format of this paper is as follows. Section II defines our Internet game, and the Nash condition from [3]. Then, Section III shows that in a discrete packet arrival model, if an agent has a finite number of packets to send in a given time window, the best strategy for the agent is to equispace the packets. Then, Section IV-A shows that CBR is the best strategy under a fluid model. Then, we show how to determine the rate of the CBR in order for the agents to reach equilibria. Finally we discuss our future directions and conclude in Section V.

II. BACKGROUND

This section describes our assumptions, the models we use, and the definition of the game we study.

A. The Internet Game

In our Internet game, we assume that selfish traffic agents can have arbitrary arrival processes, as long as they have a well defined average rate. This is in contrast to earlier work [3], [27]. The rules of the game are enforced by routers, and more specifically, the queue management algorithms at the input/output queues. The router measures the aggregate arrival rate using an exponentially weighted moving average (EWMA). Then, it calculates a drop probability that is a function of this measured rate, and drops packets with a probability given by the function.

Each player i has a simple utility function U_i equal to its goodput μ_i. As mentioned earlier, the AQM schemes in routers enforce the rules of the game on the selfish agents. In this work, we only consider oblivious AQM schemes. An oblivious router has a drop probability p due to the *measured* aggregate average load of λ, and an average service time of unity. However, for convenience, assume for now that each agent i uses a constant bit rate (CBR) with λ_i chosen upfront. Note that if traffic is a CBR with rate λ_i, then the EWMA is also λ_i. Now, oblivious routers may or may not impose *symmetric Nash equilibria* on selfish agents. A symmetric Nash equilibrium is one where every agent has the same goodput at equilibrium. We only consider *symmetric Nash equilibria* in this paper and we drop the symmetric adjective throughout the paper.

For a Nash equilibrium to hold, we have the following conditions:

- No agent can increase its goodput unilaterally, at Nash equilibrium, by either increasing or decreasing their throughput. This can be written down as

$$\forall\ i,\ \frac{\partial U_i}{\partial \lambda_i} = 0. \tag{1}$$

Since $U_i = \mu_i$, $\forall i, \frac{\partial \mu_i}{\partial \lambda_i} = 0$.

- At Nash equilibrium, all flows have the same utility or goodput. That is, $\forall i, j\ \{\mu_i = \mu_j\ \text{ and }\ \lambda_i = \lambda_j\}$.
- For oblivious AQM strategies and functions of router states like drop probability and queue length,

$$\forall i,\ \frac{\partial}{\partial \lambda_i} = \frac{d}{d\lambda}.$$

The above conditions can be used to derive an interesting condition that must be true at Nash equilibrium for an oblivious AQM. Assume there are n users. The utility function for each agent can be written down as

$$U_i\ =\ \mu_i\ =\ \lambda_i(1-p).$$

Taking partial derivatives we get

$$\frac{\partial \mu_i}{\partial \lambda_i}\ =\ 1 - p - \lambda_i \frac{\partial p}{\partial \lambda_i}\ =\ 0.$$

Since we consider only oblivious AQM schemes, we have

$$\frac{\partial p}{\partial \lambda_i} = \frac{dp}{d\lambda}.$$

Since we consider symmetric Nash equilibria, we must also have.

$$\lambda_i\ =\ \frac{\lambda}{n}.$$

It has been shown in our earlier work [3] that it is possible to derive what we call the **Nash condition** which must be satisfied at Nash equilibrium:

$$\frac{dp}{1-p}\ =\ \frac{n d\lambda}{\lambda}. \tag{2}$$

In order to apply the above to protocol equilibrium, we present an AQM that will ensure that selfish agents use CBR. Observe that the above condition is applicable to both CBR as well as Poisson traffic agents.

To evaluate whether a Nash equilibrium imposed by an AQM scheme is good, let us define a term, *efficiency*. Let the aggregate throughput, goodput, drop probability at Nash equilibrium of a system with i users be denoted by $\tilde{\lambda}_i$, $\tilde{\mu}_i$ and $\tilde{p}_i$ respectively. The Nash equilibria imposed by an AQM is efficient, if the aggregate goodput of any selfish agent is bounded below when the throughput (offered load) of that same agent is bounded above. The conditions for efficiency are:

1) $\tilde{\lambda}_i(1 - \tilde{p}_i) \geq c_1$.
2) $\tilde{\lambda}_i \leq c_2$.

where c_1, c_2 are some constants. Thus it is easy to see that even the drop probability at equilibrium is also bounded. Since

goodput is lower bounded by μ_l and upper bounded by μ_u, and $\mu = \lambda(1-p)$, we can write the Nash condition as

$$\frac{n\mu_l}{\lambda^2}\ \leq\ \frac{dp}{d\lambda}\ \leq\ \frac{n\mu_u}{\lambda^2}.$$

We call this the efficient Nash condition.

B. Router Mechanisms

As mentioned in the preceeding section, rate is measured using an exponentially weighted moving average (EWMA) which is defined as follows. Let $f(t)$ denote the aggregate arrival rate at time t. The exponentially weighted moving average of f is defined as

$$\text{ewma}(t) = \int_{x=-\infty}^{t} f(t) e^{-\eta(t-x)} dx$$

where η is a parameter called the decay constant. There is also a natural discrete analog: if t_i denotes the time of the ith packet arrival, then the expected moving average at time t_k, denoted ewma_k is given by $\sum_{i=1}^{k} e^{-\eta(t_k - t_i)}$. Without the loss of generality, we assume $\eta = 1$ in our proofs.

Each router penalizes flows at that router with a drop probability that is a function of the EWMA of the aggregate arrival rate. Hence, we use the terms penalty and drop probability interchangeably.

III. APRIORI PROTOCOL EQUILIBRIUM FOR DISCRETE PACKET SIZES

In this section, we will study the simple case of apriori equilibrium for discrete packet arrivals. While our main results are contained in the next section, this section illustrates the intuition quite well. We will restrict ourselves to linear penalty functions of the type $g(r) = \alpha r + \beta$ for suitably chosen α and β. Again, a more general protocol equilibrium is discussed in the next section. We will first study a simple single player-game.

A. Finite single player game

Now, we will show that for an agent in a single player game has only a finite number of packets to send, and the drop probability of the router is linear in the measured rate, then equispacing the packets is the best strategy.

Theorem 3.1: Consider a single-player game where the router penalty is a linear function of the EWMA of the arrival rate. Suppose the user has to send n packets in a time window Δ. Then, the solution which minimizes the total penalty is one where the packets are equispaced.

Proof:

Since the penalty is linear in the measured rate, and the total number of arrivals is n, we need to minimize the total penalty imposed by the router. Suppose the arrival sequence of the packets is given by

$$t_1, t_2, \ldots, t_n;$$

such that $t_1 \geq 0$, $t_n \leq \Delta$, $t_{i+1} \geq t_i\ \forall i, 1 < i < n$.

For each ith arrival, the router sees a rate of $r_i = \sum_{j=1}^{i} e^{-(t_i - t_j)}$. Note that each arrival consists of 1 packet

only. Now, the router penalty p_i is linear in r_i i.e. $p_i = \alpha r_i + \beta$. Without loss of generality, we can assume α, β to be unity since they will not affect our results. Thus, we want to minimize the following function:

$$U = \sum_{i=1}^{n} \sum_{j=1}^{i} e^{-(t_i - t_j)}.$$

Partially differentiating U with respect to t_i, we have

$$\frac{\partial U}{\partial t_i} = \sum_{j=1}^{i-1} -e^{t_j - t_i} + \sum_{j=i+1}^{n} e^{t_i - t_j}. \tag{3}$$

Let us define two quantities S_1 and S_2 as follows:

$$S_1(i) = e^{-t_i} \sum_{j=1}^{i-1} e^{t_j}. \tag{4}$$

$$S_2(i) = e^{t_i} \sum_{j=i+1}^{n} e^{-t_j}. \tag{5}$$

Note that decreasing t_1 and increasing t_n can only help. So we will assume that $t_1 = 0$ and $t_n = \Delta$. For all other i, we must have $\frac{\partial U}{\partial t_i} = 0$. We have by Equation 3, $\forall\ i, 1 < i < n,\ S_1(i) = S_2(i)$. By substituting $i+1$ for i in Equation 4, we get

$$
\begin{aligned}
S_1(i+1) &= e^{-t_{i+1}} \sum_{j=1}^{i} e^{t_j} \\
&= e^{-t_{i+1}} \left(\sum_{j=1}^{i-1} e^{t_j} + e^{t_i} \right) \\
&= e^{-t_{i+1}} \left(e^{t_i} S_1(i) + e^{t_i} \right).
\end{aligned}
$$

Thus, we have

$$\forall i, 1 \le i < n, S_1(i+1) = e^{t_i - t_{i+1}} \left(S_1(i) + 1 \right). \tag{6}$$

By substituting $i+1$ for i in Equation 5, we get

$$
\begin{aligned}
S_2(i+1) &= e^{t_{i+1}} \sum_{j=i+1}^{n} e^{-t_j} \\
&= e^{t_{i+1}} \left(\sum_{j=i+1}^{n} e^{-t_j} - e^{-t_{i+1}} \right) \\
&= e^{t_{i+1}} \left(e^{-t_i} S_2(i) - e^{-t_{i+1}} \right).
\end{aligned}
$$

Thus, we have

$$\forall i, 1 \le i < n,\ S_2(i) = e^{t_i - t_{i+1}} \left(1 + S_2(i+1) \right). \tag{7}$$

Dividing Equation 6 by Equation 7, we get

$$\frac{S_1(i+1)}{S_2(i)} = \frac{1 + S_1(i)}{1 + S_2(i+1)}. \tag{8}$$

Since $\forall\ i, 1 < i < n,\ S_1(i) = S_2(i)$, from Equation 8 we get

$$\forall i, 1 \le i < n-1,\ \frac{S_2(i+1)}{S_2(i)} = \frac{1 + S_2(i)}{1 + S_2(i+1)}. \tag{9}$$

Now, we solve Equation 9. Let $S_2(i) = x$ and $S_2(i+1) = y$. Then, we can simplify Equation 9 as

$$y + y^2 = x + x^2.$$

Solving for x, and taking the positive root (since $S_2 \ge 0$), we have:

$$x = -\frac{1}{2} + \frac{1}{2}\sqrt{1 + 4y + 4y^2} = y.$$

Hence, $S_2(i) = S_2(i+1), \forall i, 1 < i < n-1$. Similarly, we rewrite Equation 9 for S_1 and get $S_1(i) = S_1(i+1)$. Thus, substituting in Equation 6, we have

$$\forall i, 1 \le i < n-1,\ S_1(i+1) = e^{t_i - t_{i+1}} \left(S_1(i+1) + 1 \right). \tag{10}$$

Also, from Equation 4 we get

$$\forall i, 1 \le i < n-2,\ S_1(i+2) = e^{t_{i+1} - t_{i+2}} \left(S_1(i+1) + 1 \right). \tag{11}$$

Comparing the Equations 10, 11 and noting that $\forall i, 1 \le i < n-2,\ S_1(i+1) = S_1(i+2)$, we have

$$\forall i, 1 < i < n-2, t_{i+2} - t_{i+1} = t_{i+1} - t_i.$$

$\blacksquare$

B. Multi-player games

We now move on to multi-player games. The strategy for user i is to specify the exact arrival times beforehand in the following form: $\sigma_i = <t_{i1}, t_{i2}, \ldots, t_{ik}, \ldots>$ where $t_{i1} \ge 0$, $t_{i,j+1} \ge t_{ij}$. Call it the game G. We have the following theorem:

Theorem 3.2: If routers use EWMA to measure rate and use a linear drop function, the game G does not have a deterministic Nash equilibrium.

We omit the proof of the above theorem. The basic intuition is that each user wants its own packets to arrive infinitesimally before those of another fellow to obtain the benefit of a lower EWMA. Hence, we need to study mixed strategies to obtain a Nash equilibrium in this setting. Next we show that in the 2-player game G, we can have a mixed strategy equilibrium.
Definition: A Δ-uniform strategy send the first packet at a randomly chosen time x and from then on, each packet is sent at increments of Δ.

Now, we will assume that the router penalty function is of the form $\alpha \tilde{R}$ where $\tilde{R}$ is the EWMA. First we have the following lemma.

Theorem 3.3: In a 2-player game G, if player 1 chooses a Δ_1-uniform strategy, then the expected penalty of player 2 depends only on its own arrivals and $\frac{1}{\Delta_1}$, for a suitably chosen value of the initial rate R_0.

Proof: Let $\tilde{R}_i(t) = \sum_{j:t_{ij} \le t} e^{t_{ij} - t}$. Intuitively, $\tilde{R}_i(t)$ is the contribution to the EWMA at time t due to player i. Also assume that time $t = 0$, there is some initial rate due to EWMA. Call it R_0. Then, for a k player game, we have the following equation:

$$\tilde{R}(t) = R_0 e^{-t} + \sum_{i=1}^{k} \tilde{R}_i(t). \tag{12}$$

We will assume that the player 1 has a Δ_1-uniform strategy. Now consider a sequence of arrivals of player 2. We will now calculate the expected value of the penalty at time t. Since player 1's strategy is Δ_1-uniform, the expected number of packet arrivals due to player 1 during an infinitesimal time

interval dt is $\frac{dt}{\Delta_1}$. Hence, using linearity of expectations, we have

$$E[\tilde{R}_1(t)] \;=\; \int_0^t \frac{1}{\Delta_1} e^{x-t} dx.$$

and

$$E[\alpha R(t)] \;=\; \alpha \left(E[R_0 e^{-t}] \;+\; E[\tilde{R}_1(t)] \;+\; E[\tilde{R}_2(t)] \right)$$

$$= \; \alpha \left(R_0 e^{-t} \;+\; \int_0^t \frac{1}{\Delta_1} e^{x-t} dx \;+\; \tilde{R}_2(t) \right)$$

$$= \; \alpha \left(R_0 e^{-t} \;+\; \frac{1}{\Delta_1}(1 - e^{-t}) \;+\; \tilde{R}_2(t) \right)$$

$$= \; \alpha \left(e^{-t}(R_0 - \frac{1}{\Delta_1}) + \frac{1}{\Delta_1} + \tilde{R}_2(t) \right).$$

Choose $R_0 = \frac{1}{\Delta_1}$. Hence, the expected penalty is

$$E[\alpha R(t)] \;=\; \alpha \left(\frac{1}{\Delta_1} + \tilde{R}_2(t) \right).$$

Thus, the expected penalty of player 2 depends only on its own arrival pattern and player 1's average rate. Informally, player 2 seems to be playing a game against itself. ∎

Corollary 3.1: In a 2-player game G, if player 1 chooses a Δ_1-uniform strategy, then the best strategy for player 2 is to equispace its own packets.

The proof of the corollary follows from the the above two theorems. From Theorem 4.1, it is clear that player 2 must equispace its packets with some interval, say Δ_2.

Theorem 3.4: The two-player game G, as described above, has a mixed strategy Nash equilibrium when each player uses a Δ-uniform strategy.

Proof: In the game G, assume that player 1 has a mixed strategy. We show that the corresponding pure strategy of player 2 results in player 2 maximizing her utility. Consider the player 1 having a Δ_1 mixed strategy. By the above theorem, we know that if player 1 has a Δ_1 strategy, the best strategy for player 2 is to arrive in an equispaced fashion. Let the spacing of player 2 be Δ_2. Hence, the rates of the two players are given by λ_1, λ_2 respectively with $\lambda_1 = \frac{1}{\Delta_1}$ and $\lambda_2 = \frac{1}{\Delta_2}$. Thus, the utility of player 2, when t is sufficiently large, is given by

$$U_2 \;=\; \lambda_2 \left(1 - \alpha \left(\lambda_1 + \frac{\lambda_2}{1 - e^{-\frac{1}{\lambda_2}}} \right) \right). \tag{13}$$

Partially differentiating the above with respect to λ_2, for the above expression to have an extrema, we must have

$$\frac{\partial U_2}{\partial \lambda_2} \;=\; 1 - \alpha \lambda_1 - \frac{2\alpha \lambda_2}{1 - e^{-\frac{1}{\lambda_2}}} - \frac{\alpha e^{-\frac{1}{\lambda_2}}}{(1 - e^{-\frac{1}{\lambda_2}})^2} \;=\; 0.$$

It is possible to show that the above equation is satisfied for some $\lambda_1 = \lambda_2$ when $\alpha > 0$. This can also be verified using Mathematica, for example. Thus, the pure strategy of player 2 corresponding to the mixed strategy of the player 1 maximizes player 2's utility when both the players have the same rate, or the same inter-packet spacing. Thus, there exists a mixed strategy Nash equilibria for the game G. ∎

IV. PROTOCOL EQUILIBRIUM WITH FLUID MODELS

This section generalizes the results from the previous section using a fluid model for the traffic.

A. CBR Maximizes Utility

In this section, we first show that irrespective of the router drop function, if EWMA is used by the router to measure aggregate arrival rates, the best strategy for a single user is to send traffic as a constant bit rate (CBR). We model the arrivals using a fluid approximation to prove the above statement. Suppose the arrival rate of traffic at the router at time t is given by $f(t)$. Then, the rate estimate at the router, denoted by $R(x)$ at time x, is given by

$$R(x) \;=\; \int_0^x f(t) \cdot e^{t-x} dt. \tag{14}$$

Let $p(R(x))$ denote the drop probability for the arrival rate $R(x)$ and let $g(R(x))$ be $1 - p(R(x))$. We refer to g as the goodput function. Then, the total goodput $G(x)$ till time x is given by

$$G(x) \;=\; \int_0^x f(y)g(R(y))dy. \tag{15}$$

We also define a class of functions Ψ to contain those functions g such that $g(x) = \frac{\gamma}{x}$ has countable roots for all γ. It can be shown that most realistic goodput functions belong to Ψ; certainly the ones that we use belong to this class. For example, if $g(x)$ is any polynomial which is not identical to $\frac{\gamma}{x}$, $g(x)$ is in Ψ. Also we will assume the function f to be smooth. Now, we use the above fluid model to prove the following theorem:

Theorem 4.1: In the above fluid model and a single-player game, CBR is the best strategy for the player provided the the goodput function belongs to Ψ, and the arrival function f and its integral is smooth.

Proof: We want to maximize the following quantity

$$U \;=\; \lim_{a \to \infty} \frac{1}{a} \int_0^a f(y)g(R(y))dy.$$

Let us define a *functional* $J_a(f)$ to be

$$J_a(f) \;=\; \int_0^a f(y)g(R(y))dy.$$

Our goal is to maximize $J_a(f)$. From the fundamentals of calculus of variations, we need to obtain a function $f(x)$ that makes the *differential* of the J_a vanish. This would also mean $\frac{\delta J_a}{\delta f} = 0$, which is the necessary condition for the functional J_a to have an extremum. Now, let us divide the interval from 0 to a into n equal parts of the length δx. At $x = x_k$, the corresponding value of the function is given by $y_k = f(x_k)$. Now, J_a can be written as

$$J_a(f) = \lim_{\delta x \to 0} \Big(\int_0^{x_k - \delta x} f(y)g(R(y))dy$$
$$+ \; f(x_k)g(R(x_k))\delta x$$
$$+ \; \int_{x_k}^a f(y)g(R(y))dy \Big). \tag{16}$$

Now,
$$\frac{\delta J_a}{\delta f} = 0 \iff \lim_{\delta x \to 0} \frac{\partial J_a}{\partial (y_k \delta x)} = 0.$$

When we apply the above condition on Equation 17, we see that the first term of Equation 17 vanishes. But the third term does not because the change in y_k affects all the y_i $\forall i$, $i \geq k$ due to the computation of R. Also, since y_k depends on $f(x_k)$ for any k, the above equation implies that what we really need to find out is $\frac{\partial J_a}{\partial f(x)\delta x}$. Hence, the condition that J_a has an extremum is $\frac{\partial J_a}{\partial f(x)\delta x} = 0$. Thus, we have the following condition

$$\frac{\partial}{\partial (f(x)\delta x)} \left(f(x)g(R(x))\delta x + \int_x^a f(y)g(R(y))dy \right) = 0. \quad (17)$$

Partially differentiating Equation 17 with respect to $f(x)\delta x$ and using the Leibniz rule for differentiation under the integral sign, we obtain

$$\frac{\partial J_a}{\partial (f(x)\delta x)} = g(R(x)) + \int_x^a f(y) \cdot \frac{\partial R(y)}{\partial (f(x)\delta x)} \cdot g'(R(y)) \cdot dy.$$

Now,
$$\frac{\partial R(y)}{\partial (f(x)\delta x)} = e^{x-y}.$$

For J_a to have an extrema, we must have

$$\frac{\partial J_a}{\partial f(x)\delta x} = g(R(x)) + \int_x^a f(y) \cdot e^{x-y} \cdot g'(R(y)) \cdot dy = 0. \quad (18)$$

Differentiating the above equation with respect to x, we obtain

$$g'(R(x))R'(x) - f(x)g'(R(x))$$
$$+ \int_x^a f(y) \cdot e^{x-y} \cdot g'(R(y)) \cdot dy = 0. \quad (19)$$

Subtracting the Equation 19 from Equation 18 and simplifying, we get

$$g(R(x)) - g'(R(x))R'(x) + f(x)g'(R(x)) = 0. \quad (20)$$

Observe that the dependence on a has vanished. Now, on differentiating $R(x)$ using the Leibniz rule, we obtain

$$R(x) + R'(x) = f(x). \quad (21)$$

Thus, by applying Equation 21 to Equation 20, we have

$$g(R(x)) = g'(R(x))R'(x) - (R(x) + R'(x))g'(R(x)).$$

Simplifying, we obtain

$$g(R(x)) = -R(x)g'(R(x)).$$

On rearranging the above equation, we obtain

$$\frac{g'(R(x))}{g(R(x))}R'(x) = -\frac{R'(x)}{R(x)}.$$

On integrating and adjusting the constants, we obtain

$$R(x)g(R(x)) = \gamma, \text{ where } \gamma \text{ is a constant.}$$

Let $z = R(x)$. This implies, $z.g(z) = \gamma$. We assumed that the integral of f is smooth, and $g \in \Psi$; which implies that z can have a countable set of values it can take. Now, $z = R(x)$ is a smooth function, and does not have jumps. Hence, $z = R(x)$

must be a constant. Using Equation 21, we get $f(x)$ to be a constant. ∎

The above theorem shows that if routers use EWMA, then a single player must arrive at a constant rate. Now, the following corollary generalizes this result to many players.

Theorem 4.2: In an n-player game, if routers measure aggregate rate using EWMA, the best strategy for all agents is to arrive as CBR.

Proof: At time x, let $f(x)$ denote the aggregate rate and let the ith user arrive with a rate of $f_i(x)$. Now, by problem, $f(x) - f_i(x)$ is a constant for some i. The proof of this corollary is very similar to that of Theorem 4.1. As in the above theorem, we need to maximize

$$U = \lim_{a \to \infty} \frac{1}{a} \int_0^a f_i(x)g(R(x))dy.$$

Note that $R(x)$ depends on the aggregate rate f. Thus, the functional we need to maximize is given by

$$J_a(f) = \int_0^{x_k - \delta x} f_i(y)g(R(y))dy + f_i(x_k)g(R(x_k))\delta x$$
$$+ \int_{x_k}^a f_i(y)g(R(y))dy.$$

instead of Equation 17. Now, observe that if everyone except user i arrives at a constant rate, $\frac{\partial}{\partial f_i} = \frac{\partial}{\partial f}$. Hence, $\frac{\partial R(y)}{\partial (f_i(x)\delta x)} = e^{x-y}$. Thus, Equation 18 is unchanged, and the rest of the proof of Theorem applies in this case too.[2] Then, we get $f(x)$ to be a constant as in the above theorem. Now, $f(x) = f_i(x) +$ some constant. Hence, $f_i(x)$ is also a constant. ∎

B. Protocol Equilibria with CBR Agents

The previous section showed that agents should arrive with CBR in their own interest. Since the agents must send traffic with CBR, we can now invoke the results of our previous paper [3], and determine the rate of the CBR. This section tells us to compute the rates at which agents arriving with CBR will lead to efficient equilibria.

Assume user i arrives at a constant rate λ. Let $\lambda = \sum_i \lambda_i$. Assume R_0 to be the initial rate estimate of the EWMA at the router. We can write down the EWMA rate estimate at time x as

$$R(x) = \int_0^x \lambda \cdot e^{t-x}dt + R_0 e^{-x} = \lambda + e^{-x}(R_0 - \lambda).$$

Setting R_0 appropriately, we get $R(x) = \lambda$. Thus, the estimated rate is also λ at all times. Hence, we can use the arguments similar to [3] to design a stateless AQM to impose equilibria on selfish agents. In [3], we showed that at an efficient Nash equilibrium, the following relationship holds:

$$\tilde{\lambda}_n = 1 - \frac{1}{4n^2}.$$

---

[2] It is easy to see that if $g(x) \in \Psi$, then $g(x + \lambda) \in \Psi$, where λ is a constant.

where $\tilde{\lambda}_n$ is the estimated offered load at equilibrium by n selfish agents. Then, using the Nash condition (Equation 2), we have

$$\frac{dp}{1-p} = \frac{d\lambda}{2\lambda\sqrt{1-\lambda}}.$$

As in [3], we can solve the above equation and get

$$p = 1 - \frac{1}{\sqrt{3}}\sqrt{\frac{1+\sqrt{1-\lambda}}{1-\sqrt{1-\lambda}}}.$$

Note that the above analysis is exactly as in [3] because the estimated rate of an equispaced arrival at the router using EWMA is same as the rate itself.

V. Conclusions and Future Work

In this paper, we presented a simple mechanism to ensure that selfish agents send at CBR. We showed that under reasonable assumptions of router drop functions, if EWMA is used to estimate the aggregate arrival rate, the greedy agents must arrive with CBR to minimize losses. For a discrete arrival process and a router drop probability function that is linear in the EWMA, we showed that if a single agent wants to send a finite number of packets within a time window, it must equispace the packets arrivals. Then, we showed that for a two player game where the players choose their arrival sequences apriori, if one agent arrives in an equispaced fashion with a random shift, the other player essentially plays a game against itself, and, hence, must arrive in an equispaced fashion. We then presented our main result which is to show that for a single player game, under the assumptions of a fluid model, EWMA for aggregate rate estimation, and for a large class of drop functions, the best strategy for the agent is to arrive with a CBR. Then, we show that if in an n-player game, if $n-1$ agents arrive with CBR, then the best strategy for the nth player is also to do the same. Finally, we apply the results from our earlier paper [3] to determine the rates of the agents that will lead to efficient equilibria.

There are several directions that deserve further exploration. We feel that this paper is the first attempt to achieve protocol equilibria. The first interesting problem is to study the effects of buffering. In this paper, we have assumed the goodput to be our utility. It would be interesting to revisit the problem with respect to different utility functions such as average delay and jitter. As we have pointed out in Section 1, a malicious user can easily bring down a scheme that takes of care of greed alone. We need to design low state filters to weed out the unresponsive malicious flows from the system. Finally, an open problem is to design a scheme in practice that is insulated against both greed and malice, and requires very little state to be maintained while providing close to maximum network utilization.

Acknowledgment

References

[1] A. Akella, R. Karp, C. Papadimitrou, S. Seshan, and S. Shenker, "Selfish behavior and stability of the internet: A game-theoretic analysis of TCP," *ACM SIGCOMM*, 2002.

[2] A. Demers, S. Keshav, and S. Shenker, "Analysis and simulation of a fair queueing algorithm," *ACM SIGCOMM*, 1989.

[3] D. Dutta, A. Goel, and J. Heidemann, "Oblivious AQM and Nash Equilibria," *IEEE INFOCOM*, April 2003.

[4] C. Estan and G. Varghese, "New directions in traffic measurement and accounting," *ACM SIGCOMM*, August 2002.

[5] J. Feigenbaum, C. H. Papadimitriou, and S. Shenker, "Sharing the cost of multicast transmissions," *Journal of Computer and System Sciences*, vol. 63, no. 1, pp. 21–41, 2001.

[6] S. Floyd and K. Fall, "Promoting the use of end-to-end congestion control in the Internet," *IEEE/ACM Transactions on Networking*, vol. 7, no. 4, pp. 458–472, 1999.

[7] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance." *IEEE/ACM Transactions on Networking, V.1 N.4*, 1993.

[8] R. Gibbens and F. Kelly, "Resource pricing and the evolution of congestion control," *Automatica*, no. 5, pp. 1969–1985, 1999.

[9] V. Jacobson, "Congestion avoidance and control," *ACM Computer Communication Review; Proceedings of the SIGCOMM '88 Symposium in Stanford, CA, August, 1988*, vol. 18, 4, pp. 314–329, 1988.

[10] K. Jain and V. V. Vazirani, "Applications of approximation algorithms to cooperative games," in *ACM Symposium on Theory of Computing*, 2001, pp. 364–372.

[11] R. Johari and J. Tsitsiklis, "Network resource allocation and a congestion game," in *Mathematics of Operations Research (to appear)*.

[12] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," in *Journal of the Operational Research Society*, vol. 49, 1998.

[13] Y. A. Korilis and A. A. Lazar, "On the existence of equilibria in noncooperative optimal flow control," *Journal of the ACM*, vol. 42, no. 3, pp. 584–613, 1995.

[14] Y. A. Korilis, A. A. Lazar, and A. Orda, "Architecting noncooperative networks," *IEEE Journal of Selected Areas in Communications*, vol. 13, no. 7, pp. 1241–1251, 1995.

[15] E. Koutsoupias and C. Papadimitriou, "Worst-case equilibria," *Lecture Notes in Computer Science*, vol. 1563, pp. 404–413, 1999.

[16] D. Lin and R. Morris, "Dynamics of random early detection," *ACM SIGCOMM*, pp. 127–137, September 1997.

[17] R. Mahajan, S. Floyd, and D. Wetherall, "Controlling high-bandwidth flows at the congested router," *ICNP*, november 2001.

[18] G. Manku and R. Motwani, "Approximate frequency counts over data streams," in *In Proceedings of the 28th International Conference on Very Large Data Bases*, July 2002.

[19] M. J. Osborne and A. Rubenstein, *A Course in Game Theory*. Cambridge, Massachusetts: The MIT Press, 1994.

[20] J. Padhye, V. Firoiu, D. Towsley, and J. Krusoe, "Modeling TCP throughput: A simple model and its empirical validation," in *ACM SIGCOMM '98 conference on Applications, technologies, architectures, and protocols for computer communication*, Vancouver, CA, 1998, pp. 303–314.

[21] C. H. Papadimitrou, "Algorithms, games, and the internet," in *ACM Symposium on Theory of Computing (STOC)*, 2001.

[22] K. Park, M. Sitharam, and S. Chen, "Quality of Service provision in noncooperative networks: heterogeneous preferences, multi-dimensional qos vectors, and burstiness," *Proc. 1st International Conference on Information and Computation Economies*, 1998.

[23] V. Paxson, "End-to-end Internet packet dynamics," *IEEE/ACM Transactions on Networking*, vol. 7, no. 3, pp. 277–292, 1999.

[24] J. Postel, "Transmission Control Protocol," September 1981, internet Request for Comments RFC 793.

[25] T. Roughgarden, "Designing networks for selfish users is Hard," in *IEEE Symposium on Foundations of Computer Science*, 2001, pp. 472–481.

[26] T. Roughgarden and E. Tardos, "How bad is Selfish Routing?" in *IEEE Symposium on Foundations of Computer Science*, 2000, pp. 93–102.

[27] S. Shenker, "Making greed work in networks: A game-theoretic analysis of switch service disciplines," *IEEE/ACM Transactions on Networking*, vol. 3, 1995.

[28] I. Stoica, S. Shenker, and H. Zhang, "Core-stateless Fair Queueing: Achieving approximately fair bandwidth allocations in high speed networks," *ACM SIGCOMM*, 1998.

[29] K. Thompson, G. J. Miller, and R. Wilder, "Wide-area internet traffic patterns and characteristics," *IEEE Network*, April 2003.