Modelling the Performance of In-call Probing for Multi-Level Adaptive Applications

Alan Bain
Statistical Laboratory, Cambridge

Peter Key
Microsoft Research

October 2001

Technical Report

MSR-TR-2002-06

This paper looks at adaptive applications that can switch between a small number of different levels of service, with switching decisions made solely by the originating end-system. Typical of such applications are real time streaming protocols which can use different coding rates. The end-systems probe the network with sample traffic to determine congestion, and decide at what rate to enter according to the fate of their "probe" packets. During the lifetime of a connection, the procedure is repeated to reassess and possibly readjust the rate. We derive analytic models, based on diffusion limits under a natural scaling, to quantify the benefits of in-call probing. We then use simulation to compare the results in a number of scenarios, and show that this simple theory is remarkably accurate in predicting large-scale behaviour. The results also show that a small amount of in-call probing produces significant benefits to the system.

Microsoft Research
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
http://www.research.microsoft.com

1 Introduction

Multimedia applications for the Internet, such as streaming voice or video, or interactive real-time services like voice over IP (VOIP), typically require some minimum bandwidth to function correctly, and are sensitive to congestion. UDP-based protocols are usually insensitive to network conditions, which may not only be counter productive, causing lost data through self-generated congestion, but also behave 'unfairly' compared to adaptive TCP applications.

Reservation, such as the Intserv RSVP proposal [4], avoids congestion by partitioning bandwidth, and gives guarantees to applications, but requires signalling and has scalability problems. An alternative approach uses end-system behaviour to respond to congestion, and normally uses rate control [22], where the end-system uses measurements or probes to determine the state of the network and alter the sending rate. Such schemes [3, 21, 19, 20] typically seek to estimate loss rates and adjust the sending rate of the voice or video rather frequently. Making the rate adaptation follow an additive increase/multiplicative decrease rule gives a form of TCP friendliness [19, 20] where feedback can use the control channel (RTCP) of RTP.

In contrast, we concentrate on typical real time applications where packets can be sent at a small number of distinct rates (corresponding, for example, to differing compression levels and hence differing picture/sound quality from a user's perspective). From the users' viewpoint, overly frequent changes in quality are undesirable; yet adaptation is necessary due to the bandwidth constraint on the channel. Hence we assume infrequent probing where the network state is only probed at those instants when a change of rate is being considered. The probing neither involves applying artificial extra load to the network nor is it used to accurately estimate loss; instead we use a snapshot of the congestion obtained from the data transmission in progress.

In addition, we prefer to use packet marking as an advance warning of incipient network problems, rather than loss. Marking may be implemented as a single bit at the IP level, by building on the proposal in [17] (which focuses solely on TCP), or can be implemented as a multibit signal. In our framework, a probing strategy typically involves looking over a small number of recently sent packets and determining how many of them are marked. According to this number the new data rate for future use of this connection is chosen. Related work in this area has concentrated on admission control, [2, 6, 5, 12, 13], whereas we concentrate on in-call adaptation.

Proposed strategies for marking packets include Random Early Detection (RED) or virtual queue based marking [12]. We shall assume that a timescale separation exists and the marking behaviour can be described as a function of the current load at the queue.

In this paper, we show how to model analytically a large network carrying such adaptive applications. In particular, we derive diffusion limits that allow us to characterise second-order properties of the traffic. Such second-order quantities are essential for performance analysis, since it is the deviations from the mean or tail probabilities which dictate the loss and delay performance which an end-user sees. The models can be used with analytic models of marking behaviour, or to enhance simulations. Specifically, if we use simulations to derive marking behaviour for a small system, we can use analysis to deduce behaviour of large systems which it is infeasible to simulate. We are also able to quantify the benefits of in-call probing, and show that most of the benefit is indeed gained by probing infrequently relative to the call-holding time.

The outline of the paper is as follows: In Section 2, we describe our general framework, and consider some of the architectural issues connected with implementing feedback and reaction of the type proposed. We then introduce a specific motivational example which will be used throughout the paper, in which an application can send at one of two levels, and can switch between them during the call; and the network uses a specific packet-level marking strategy. Section 3 describes the analytic models which allow the benefits of in-call probing to be assessed. We use a large network scaling to derive a fluid limits, give conditions for stability of the control mechanism and then derive diffusion limits which enable us to assess the performance of the system. It turns out that deriving the crucial covariance matrix is equivalent to solving a Lyapunov stability matrix equation, which can be solved using standard techniques. Section 4 presents results of simulations of various scenarios, including those a large star network, and compares these results with the analytic predictions, examining the sensitivity of the results to the assumptions. The results illustrate the accuracy of the analytical results, which are very quick to calculate, in contrast to simulations.

2 Framework and Architecture

In this section we discuss a specific, purposely simple, model of adaptation and end-point admission control, considering some of the associated architectural issues. This simple model is chosen not because it is suggested as a practical implementation; but it will be seen to be stable and highly effective, and this effectiveness will only be bettered by any realistic in-call probing strategy. We shall use packet marking as a means of signalling congestion; packet loss could be used as an alternative, though less informative, means to the same end.

2.1 Connection level adaptation

Suppose there are I different rates which a call or flow can use. When a call (flow) starts, we assume that the end-system sends a small number of *probe packets* into the network, and receives feedback in the form of those packets being marked (or dropped). This information is then used to determine at which of the I rates to enter, where for the moment we assume a call is always accepted. In addition, whilst in progress, each call re-probes the network periodically, and is able to switch between rates. We shall assume that there is a sufficiently great timescale difference between the packet level and the call level that individual acceptance/switching decisions may be considered as independent. Then at the call timescale, the probing and marking behaviour may be taken as defining implicitly for each resource in the network the probability that after probing, a call is accepted at the ith level. This probability is denoted $a^{(i)}(\rho(t))$, where $\rho(t)$ is the load on the resource when the snapshot is taken.

In the network, each route r is represented as the subset of resources which are used by the route. Let $\mathbf{x}_r$ be the vector of calls at levels 1 through I on route r. Then $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_R)$ is the vector of the number of calls on each route at the various rates. Provided sufficient independence exists between resources, we can introduce a product form acceptance function for the

probability of using level i for a call on route r

$$a_r^{(i)}(\mathbf{x}) = \prod_{j \in r} \hat{a}_j^{(i)}(\rho_j(\hat{\mathbf{x}}_j)), \tag{1}$$

where $\hat{a}_j^{(i)}(\cdot)$ is the acceptance function at level i for resource j and $\hat{\mathbf{x}}_j = \sum_{r:r \ni j} \mathbf{x}_r$ is the aggregate number of calls using resource j at the various rates. The load at resource j is given by $\rho_j(\hat{\mathbf{x}})$. We shall assume that the same probing strategy is used during in-call adaptation, hence the function $a_r^{(i)}(\mathbf{x})$ is the probability that a call along route r chooses level i after probing.

In contrast to [5], we do not use probing to estimate the precise level of congestion, but rather use it as a guide. Connections are continually arriving and making decisions, hence the system has a self-regulating property. We do not assume any separate channel or scheduling for the probing packets.

We assume that in-call probing is a relatively infrequent process, occurring a few times per call. In the context of an IP telephony call lasting say 200 seconds, this corresponds to probing on the order of tens of seconds. We shall see simulation results which show that we get most of the benefit by probing just once or twice during the call.

2.2 Packet-level Models and Marking

We assume standard FIFO scheduling at the routers, which additionally marks packets according to some specific policy. For example, we may mark packets when some threshold in a buffer is exceeded, or use enhancements based on active queue management such as RED [9].

We assume that this marking is performed at the IP level by the router setting the Congestion Experienced (CE) bit defined by the ECN proposal [16]. This proposal only describes the reaction of TCP to the marks, effected by having TCP receiver return the marks to the source. Similarly we require the streaming source applications to have access to the state of the marked packets: this can either be done at the application level, such as by having the receiver generate return UDP packets, or by using a control channel (corresponding to RTCP for RTP [18]).

The acceptance function $a(\cdot)$ is determined both by the marking function and by the user or end-point policy. A particularly simple user policy for a two level model is the following: send n packets into the network, choose the high rate if no packets are marked (or lost), and otherwise choose at the low rate.

2.3 Example

While our approach handles arbitrary numbers of rates I and applies to complex networks, for ease of exposition we shall concentrate on the simplest case, where there are only two rates (high and low), the network comprises a single resource, and where the resource load is generated by this class of calls alone. The latter is an appropriate model if such applications are segregated for example in a separate DiffServ class.

Suppose that calls originate as a Poisson process of rate ν, and when admitted last for an exponentially distributed holding time with mean one (we can rescale to general mean holding times). At the start of the call, at time t, the network is probed

and the call enters at the high rate, r_H with probability $a(\rho)$ and enters at the low rate, r_L with probability $1-a(\rho)$, where $\rho = \rho(t)$.

Let $X_t^{(1)}$ be the number of high rate calls, $X_t^{(2)}$ the number of low rate calls in progress at time t, each call generating load at mean rates r_H and r_L respectively. The capacity of the resource is C. Then $\rho(X_t^{(1)}, X_t^{(2)}) = (X_t^{(1)} r_H + X_t^{(2)} r_L)/C$. For the moment, we assume that connections are always accepted, and so calls enter at the high level at rate $\nu a(\rho)$, while high-rate calls depart at rate $X_t^{(1)}$, and low rate calls depart at rate $X_t^{(2)}$.

While the connection is in progress, probing occurs as a random (Poisson) process of rate λ, and with probability $a(\rho)$ a call can switch from a low rate to a high rate; this occurs at rate $\lambda X_t^{(2)} a(\rho)$, since there are $X_t^{(2)}$ calls in progress at the low rate. The corresponding rate for switching between high rate and low rate is $\lambda X_t^{(1)}\,(1-a(\rho))$

If we now look at the network on such a large scale that we do not see the random effects of the individual calls, then the network may be viewed as a flow of calls, with the state of the network represented by continuous functions $x_1(t)$ and $x_2(t)$ instead of the stochastic processes $X_t^{(1)}$ and $X_t^{(2)}$. Combining in-call adaptation with the arrival and departure rates, we see that the rates might be expected to satisfy,

$$\begin{aligned}
\frac{\mathrm{d}x_1}{\mathrm{d}t} &= \nu a(\rho(x_1(t), x_2(t))) - x_1(t) + \lambda a(\rho(x_1(t), x_2(t)))x_2(t) - \lambda(1 - a(\rho(x_1(t), x_2(t)))x_1(t), \\
\frac{\mathrm{d}x_2}{\mathrm{d}t} &= \nu \left\{1 - a(\rho(x_1(t), x_2(t)))\right\} - x_2(t) - \lambda a(\rho(x_1(t), x_2(t)))x_2(t) + \lambda \left\{1 - a(\rho(x_1(t), x_2(t)))\right\} x_1(t).
\end{aligned} \tag{2}$$

This is made precise in Section 3, where we prove that these equations are exactly those which arise under a specified limiting regime.

Let K be the threshold for marking in the virtual queue. VQ marking can be implemented with a counter that behaves as a virtual queue, running at a reduced service rate κC compared to the real queue, for some $\kappa \leq 1$. Packets arriving at the real queue are notionally also put into the virtual queue, and marked when the buffer in the virtual queue exceeds a given threshold, K. It is suggested in [12] that one set $\kappa = (K+1)^{-1/K}$. If the packet level queueing behaviour is approximately described by an M/M/1 queue, then for the simplest strategy where just a single probing packet is used, the acceptance function is approximated by

$$a(\rho) = \max\left(0, 1 - (K+1)\rho^K\right). \tag{3}$$

2.4 Rejection

We have deliberately ignored call rejection to avoid introducing another parameter, however the methodology described applies equally well when there is rejection. Roughly speaking, in terms of mean behaviour, rejection has a similar effect to a reduction in the offered traffic. For I possible rates, $a^{(i)}$ is the probability we accept in state i, and $1 - \sum_{i \in I} a^{(i)}$ is the probability we reject a call. Given the the same reaction to in-call probes as to those at the start of call, then in the case of just two levels, the equilibrium values x_1 and x_2 satisfy

$$x_1 = \frac{a^{(1)}\nu}{1 + \lambda\left(1 - a^{(1)} - a^{(2)}\right)}, \qquad x_2 = \frac{a^{(2)}\nu}{1 + \lambda\left(1 - a^{(1)} - a^{(2)}\right)}, \tag{4}$$

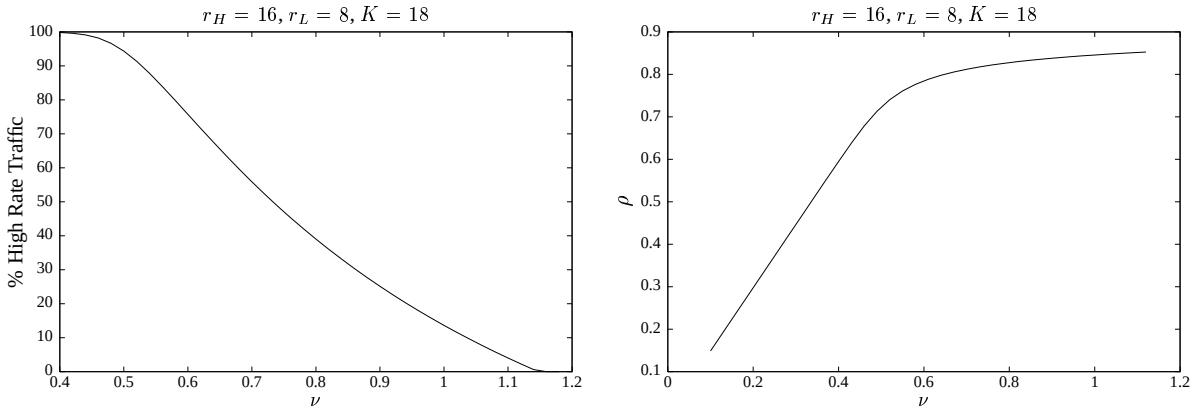

Figure 1: The effect of ρ on the average percentage of high rate calls.

which for small rejection probabilities ($a^{(1)} + a^{(2)}$ close to 1) is similar to the behaviour with no rejection and a reduced arrival rate of $\nu/\left[1 + \lambda(1 - a^{(1)} - a^{(2)})\right]$.

3 Analysis and Performance

The previous section defined an adaptive application at the microscopic level of individual calls. We now show that aggregate quantities satisfy certain stochastic differential equations, which leads to a functional weak law of large numbers and a corresponding functional central limit theorem.

3.1 Scaling and a Diffusion Limit

In order to examine the behaviour of the link as a large scale structure, we need to consider the correct limit. This limit is defined over a family of systems indexed by N as $N \to \infty$. In the Nth system, the arrival intensity (arrival rate divided by mean call length) is taken to be νN, whilst the mean call length is held fixed; thus the arrival rate grows linearly in N. The link capacity, that is the service rate of the queue, will be taken as CN packets per second, and it is natural to define $\tilde{X}_t^{(1)} = X_t^{(1)}/N$ and $\tilde{X}_t^{(2)} = X_t^{(2)}/N$. For practical networks, we might typically be interested in N of the order of 10^2 or 10^3.

Figure 1 shows the equilibrium dependence of the nominal utilisation ρ and the percentage of the traffic carried at the high rate, upon the arrival rate ν. For low values of ν, almost all of the traffic is carried at the high rate; whereas for large ν the majority of traffic is carried at the low rate. In practice values of ν this high are avoided by rejecting calls as described in section 2.4.

3.2 Fluid Limit

In the limit as $N \to \infty$, one can show that the scaled network traffic $(\tilde{X}_t^{(1)}, \tilde{X}_t^{(2)})$ converges uniformly on compact sets in t to the fluid limit process, which is the process which solves the system of ordinary differential equations given by (2). Note that

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(x_1(t) + x_2(t)\right) = \nu - \left(x_1(t) + x_2(t)\right),$$

which yields $x_1(t) + x_2(t) = \nu - Ce^{-t}$. Hence as $t \to \infty$, $x_1(t) + x_2(t) \to \nu$; thus the system dynamics are fully described by those on the manifold $x_1 + x_2 = \nu$. An example of the trajectories of this system of fluid equations is shown in Figure 2, for a probing rate of $\lambda = 1$. The equilibrium point is marked by the triangle, and individual curves correspond to different starting values.

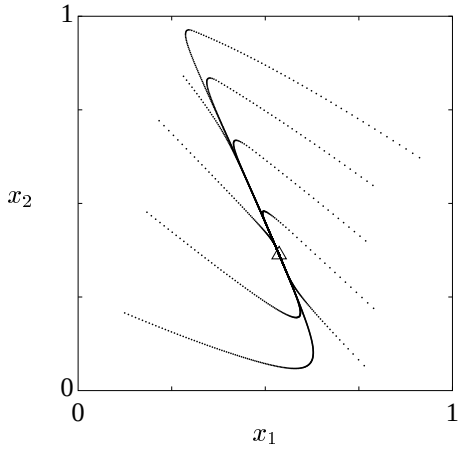

Figure 2: Trajectories of the Fluid Limit.

3.3 Convergence and Stability

The fluid equations have an equilibrium point given by the solution of the equations

$$\overline{x}_1 + \overline{x}_2 = \nu, \qquad \overline{x}_1 = \nu a(\rho(\overline{x}_1, \nu - \overline{x}_1)).$$

Note first that the equilibrium point is independent of λ (the rate of in-call probing), and second that the fixed point is unique. Uniqueness follows from a simple monotonicity argument, since for any reasonable acceptance strategy, transferring calls from the low rate to the high rate should not increase the acceptance probability for a new call! The previous observation that the dynamics are described by those on the manifold $x_1 + x_2 = \nu$ means it suffices to consider the local stability about this point. Uniqueness of the stationary point means that if this point is locally stable, then the dynamics are those on the real line, with a single stable fixed point, which implies that the system is globally stable. Define $u_i = x_i - \overline{x}_i$, for $i = 1,2$ then linearising the fluid equations about their fixed point $(\bar{x}_1, \bar{x}_2)$ gives:

$$\begin{pmatrix} \dot{u}_1 \\ \dot{u}_2 \end{pmatrix} = H \begin{pmatrix} u_1 \\ u_2 \end{pmatrix},$$

where H is given by

$$H = \begin{pmatrix} \nu a_x - 1 + \lambda((\overline{x}_1 + \overline{x}_2)a_x - 1 + a) & \nu a_y + \lambda((\overline{x}_1 + \overline{x}_2)a_y + a) \\ -\nu a_x - \lambda((\overline{x}_1 + \overline{x}_2)a_x - 1 + a) & -\nu a_y - 1 - \lambda((\overline{x}_1 + \overline{x}_2)a_y + a) \end{pmatrix}, \tag{5}$$

and

$$a = a(\rho(\overline{x}_1, \overline{x}_2)), \qquad a_x = \left.\frac{\partial a(\rho(x,y))}{\partial x}\right|_{\overline{x}_1, \overline{x}_2}, \qquad a_y = \left.\frac{\partial a(\rho(x,y))}{\partial y}\right|_{\overline{x}_1, \overline{x}_2}. \tag{6}$$

Direct evaluation of the determinant of H shows that the eigenvalues are -1 and Ψ, where

$$\Psi = (1+\lambda)\left(\nu(a_x - a_y) - 1\right).$$

Thus provided that, $a_x - a_y < 1/\nu$ both eigenvalues will be negative and so the fixed point will be stable. This condition is satisfied by the acceptance models we consider: since the acceptance probability is a function of the nominal intensity, ρ, the condition is $a'(\rho_x - \rho_y) < 1/\nu$. As the acceptance probability is non-increasing in ρ, $a' \leq 0$. As x corresponds to the number of high level calls and y to the number of low level calls we may assume $\rho_x \geq \rho_y$ and so the inequality is trivially satisfied.

3.4 Stability and Delays

There is a RTT delay between sending probe packets and receiving congestion indication signals. We have seen that when there is no such delay, the fixed point is stable for all values of λ. In the case when a delay is present we shall see that there is a maximum value of λ for which the fixed point is stable. Let D be the round trip delay, and assume the acceptance function is a function of ρ, with $\rho(t)$ defined naturally. Hence the fluid limit are a obtained from (2) with $a(\rho(t))$ replaced by $a(\rho(t-D))$. As before as $t \to \infty$, $x_1(t) + x_2(t) \to \nu$. Without loss of generality, put $r_H = 1$, $r_L = \frac{1}{2}$ and define the function $p(y) = 1 - a(y/C)$, then it suffices to consider the equation

$$\frac{\mathrm{d}x_1(t)}{\mathrm{d}t} = (1+\lambda)\left[\nu\left\{1 - p\left(\frac{x_1(t-D)}{2} + \frac{\nu}{2}\right)\right\} - x_1(t)\right].$$

Linearising about the fixed point, setting $\psi(t) = x_1(t) - \overline{x}_1$ with $\bar{y} = \bar{x}_1 + \bar{x}_2/2$ gives

$$\frac{\mathrm{d}\psi(t)}{\mathrm{d}t} = (1+\lambda)\left[-\frac{\nu}{2}p'(\bar{y})\psi(t-D) - \psi(t)\right].$$

Taking Laplace transforms gives the characteristic equation in s, whence setting $\gamma = sD$ we obtain

$$-D(1+\lambda)e^{\gamma} - \gamma e^{\gamma} - D(1+\lambda)\frac{\nu}{2}p'(\bar{y}) = 0.$$

This fixed point is stable if the roots of the characteristic equation both have negative real parts. By the theorem of Hayes [1, Theorem 13.8] this holds if and only if

$$-D(1+\lambda) < D(1+\lambda)\frac{\nu}{2}p'(\bar{y}) < \sqrt{\phi^2 + (D(1+\lambda))^2},$$

where ϕ is the root of $\phi = -D(1+\lambda)\tan\phi$ such that $0 < \phi < \pi$. (Note that in contrast to the no delay solution, increasing λ has a negative impact on convergence). Since $D > 0$ and $\lambda > 0$ then $\phi > \pi/2$. Hence for virtual queue marking, using the

notation of Section 2.3 and equation (3) for $a = 1 - p$ we have the sufficient condition

$$D(1+\lambda)\frac{\nu}{2c}(K+1)K\left(\frac{\bar{y}}{C}\right)^{K-1} < \sqrt{\left(\frac{\pi}{2}\right)^2 + (D(1+\lambda))^2}.$$

We are interested in the case $\rho < 1$, implying $\nu < 2C$, giving the sufficient condition

$$D(1+\lambda)(K+1) < \pi/2.$$

This gives us an upper bound on the reprobing rate λ in order that the equilibrium point remain stable. For example with $K = 18$, we require $D(1+\lambda) < 0.08$ (a more exact calculation gives the right hand-side as 0.2). Recall that we have rescaled units so that the holding time is 1, and effectively D is measured in holding time units, which in practice means there should be no stability problems. We are typically interested in flows lasting tens of seconds: for a mean holding time of 200s, and a round-trip time of 200ms we can allow λ to increase to 80 and still have a stable system.

3.5 Variance and the Diffusion Limit

There is no significant dynamic behaviour described by the fluid limit – the network traffic simply converges to the unique fixed point; for the network to be stable this must correspond to a load on all the links of less than one. In the case of a real network, the fluid limit is useful to approximate the behaviour of a finite size system. In this system fluctuations from the fluid limit are not negligible, since with an equilibrium point chosen so that all of the links are subcritically loaded, it is the fluctuations which cause queues to overflow. Thus from a performance point of view we are interested in the probability of these events; they may be calculated from knowledge of the equilibrium point and the traffic covariance matrix. The fluctuation behaviour is described by the diffusion limit.

Let $\mathbf{U}(t)$ be the vector of differences of the traffic vector from the equilibrium, so $\mathbf{U}_t = (\tilde{X}_t^{(1)} - \overline{x}_1, \tilde{X}_t^{(2)} - \overline{x}_2)$. Then the following central limit theorem holds, the proof of which may be found in the Appendix.

Theorem 1. *If in the limit as $N \to \infty$, $\sqrt{N}\mathbf{U}(0) \stackrel{\mathcal{D}}{=} A$, for some random variable A, then as $N \to \infty$ the deviations from the equilibrium fluid limit satisfy*

$$\lim_{N\to\infty} \sqrt{N}\mathbf{U}(t) \stackrel{\mathcal{D}}{=} R_t,$$

where $R(t)$ is the unique solution with $R_0 = A$ of the stochastic differential equation

$$\mathrm{d}\mathbf{R}_t = H\mathbf{R}_t\mathrm{d}t + F\mathrm{d}\mathbf{B}_t, \tag{7}$$

where $\mathbf{B}_t$ is a six dimensional Brownian motion. This equation describes an Ornstein-Uhlenbeck process. The convergence described is weak convergence in $D_{\mathbb{R}^2}$, the space of càdlàg paths, endowed with the Skorohod J_1 topology. The matrix H is

the linearisation of the fluid limit about the equilibrium point given by (5), and

$$F = \begin{pmatrix} \sqrt{a\nu} & -\sqrt{\overline{x}_1} & \sqrt{a\lambda\overline{x}_2} & -\sqrt{(1-a)\lambda\overline{x}_1} & 0 & 0 \\ 0 & 0 & -\sqrt{a\lambda\overline{x}_2} & \sqrt{(1-a)\lambda\overline{x}_1} & \sqrt{\nu(1-a)} & -\sqrt{\overline{x}_2} \end{pmatrix} \tag{8}$$

where a, a_x and a_y are given by (6).

This theorem has the following corollary.

Corollary 2. *Let the initial condition A be such that the stationary solution to the SDE is obtained, then at any fixed time time t, the scaled difference vector $\sqrt{N}\mathbf{U}_t$ is distributed according to a multivariate normal distribution with mean zero and covariance matrix Σ given by*

$$\Sigma = \int_{-\infty}^{0} e^{-uH} F F^T (e^{-uH})^T \mathrm{d}u,$$

with F and H are as defined in the previous theorem.

The above was derived for a single link, it readily generalises to the context of a network by writing equations for the traffic along each route r. We have already seen the product form of the acceptance function (1). In a similar fashion fluid equations can be derived for this traffic, and the above theorem and corollary extend to the network case; with H defined as the matrix which linearises the network fluid equations, and F a block matrix with a block corresponding to each route of the form (8).

If the matrix H is diagonalisable then we may write $H = \Gamma\Phi\Gamma^{-1}$, where Φ is a diagonal matrix, with the eigenvalues of H, namely ϕ_1 and ϕ_2 along the diagonal. Following simple manipulation

$$\Sigma = \left(\Gamma \left[\int_{-\infty}^{0} e^{-u\Phi}\Gamma^{-1} F F^T \left(\Gamma^{-1}\right)^T e^{-u\Phi} \mathrm{d}u \right] \Gamma^T \right).$$

The integral inside the square brackets may be expressed directly terms of the eigenvalues of H, so

$$\left[\int_{-\infty}^{0} e^{-u\Phi}\Gamma^{-1} F F^T \left(\Gamma^{-1}\right)^T e^{-u\Phi} \mathrm{d}u \right]_{r,s} = \frac{\left(\Gamma^{-1} F F^T \left(\Gamma^{-1}\right)^T\right)_{r,s}}{\phi_r + \phi_s}.$$

Through this technique the explicit form of the variance for a special case of the single link is found 3.6. This is a poor technique for solving the problem in the case of a large H matrix, since the diagonalisation process is prone to numerical instability. Instead the following method is used. Applying integration by parts, it is readily seen that Σ satisfies a Lyapunov stability equation $\Sigma H^T + H\Sigma = -FF^T$. Efficient and stable numerical technique exists to solve this system, [11]. Hence the steps needed in a numerical computation of the covariance matrix are:

(i) Find the equilibrium point. This is easily performed by the use of a modification of the Newton-Raphson method. In many cases this equilibrium point is independent of λ, so this step does not need to be repeated when considering the effect of varying λ.

(ii) Calculate the values a, a_x and a_y at this equilibrium point, and hence evaluate the matrices H and F.

(iii) Solve the Lyapunov equation for Σ.

3.6 Example

We let the acceptance probability be given by (3) and consider the total traffic variance, which may be expressed in terms of the covariance matrix Σ as

$$v(\lambda) = r_H^2 \Sigma_{11} + r_H r_L \left(\Sigma_{12} + \Sigma_{21}\right) + r_L^2 \Sigma_{22}.$$

Recall that the acceptance functions give the probability of accepting at the high rate, which decreases as the load increases. The variance decreases sharply with λ: a value of $\lambda = 1$, corresponding to just one in-call probe per call on average, reduces the variance of the carried traffic by about 30%; while for $\lambda = 7$ the reduction is about 50%. Little is gained by having much larger values of λ.

For a single link case where we do not reject any traffic, when $r_H = 2r_L$, the variance of the total traffic has the explicit form

$$v(\lambda) = \frac{2{r_L}^2 \left(\nu + 3x_1\right)}{2 - \nu a_x} + \frac{2\lambda a_x {r_L}^2 (\nu + x_1)^2}{\left(2 - \nu a_x\right)\left(2\left(2 + \lambda\right) - \left(1 + \lambda\right)\nu a_x\right)} \tag{9}$$

where we have made use of the relations $a_y = a_x/2$, $x_2 = \nu - x_1$ and $\nu a = x_1$. The first term in equation (9) is the variance when there is no probing ($\lambda = 0$), and shows the impact of the derivative a_x on the variance. Recall that a_x is a negative, hence the larger the modulus of a_x, the smaller the variance. The second term in (9) has negative sign (giving a reduction in variance for non-zero λ) whose rate of decrease decays to zero as λ increases, and the variance tends to a limit. In fact the ratio of the decrease

$$\frac{v(1) - v(0)}{v(\infty) - v(0)} = \frac{2 - \nu a_x}{6 - 2\nu a_x} \tag{10}$$

shows that at least a third of the possible variance reduction is obtained by using a λ of just 1. For $\lambda = 5$, the ratio decreases to $(10 - 5\nu a_x)/(14 - 6\nu a_x)$, hence $\lambda = 5$ gives at least 70% of the maximum benefit.

4 Results and Performance

We used simulations of a single link to determine the average marking rate as a function of the load, giving a form for the function $a(\rho)$. This "empirically determined" acceptance function was then used to evaluate analytically the performance of a larger and more complex network made up of many links.

The analysis has used some simplifications in the interests of tractability. Significant assumptions are that the calls have exponentially distributed lengths, and that the round-trip delay for packets has a negligible effect on the limiting process. To evaluate the effects of these simplifications, we compare the analytical results to simulations of the whole complex network.

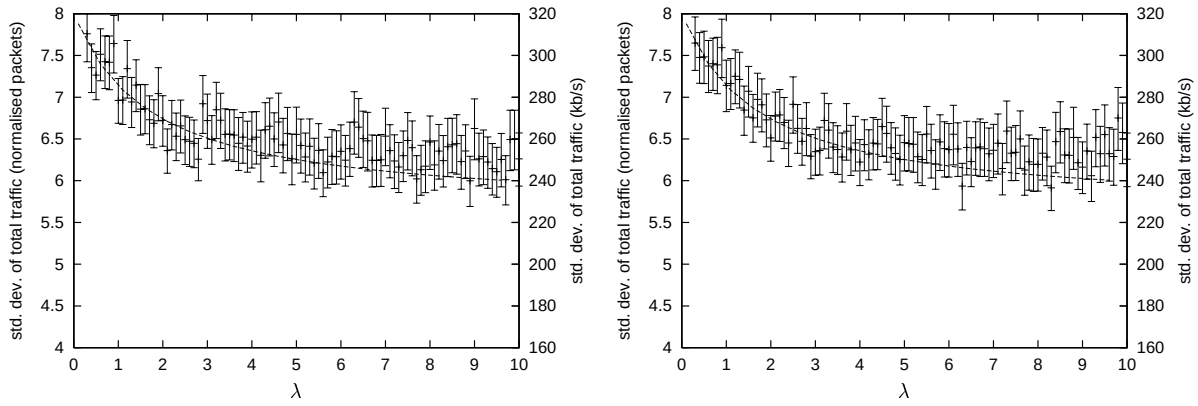

Figure 3: Traffic standard deviation against probing rate λ for a single link, with link delay 10ms (left) and 100ms (right). The theoretical prediction is shown by the dashed line.

4.1 Simulation Methodology

The *ns*[8] network simulator was used to model various scenarios, such as a single-link and a star-network. In the first phase a single-link was subjected to a variety of loads and the proportion of packets marked recorded. In the second phase an adaptive system of the form described in Section 2 was used. Enhancements to *ns* were created to allow in-call probing and adaptation based on ECN marks.

The 'calls' in the simulations resemble typical voice calls carried over an IP network. The 'high' data rate has been chosen as 64kbit/s (the value used in the ITU G711 PCM encoding of voice), and the 'low' rate chosen as 32kbit/s.

Calls arrive as a Poisson process of rate ν per holding time; each call lasts for an exponential period of time, with mean 200s. Each call also reprobes at rate $\lambda/200$ (as a Poisson process, with the scaling such that a value of $\lambda = 1$ corresponds on average to one "in-call" probe per call). Each call generates a stream of regularly spaced 500 byte UDP packets.

In the simulations the probing strategy is implemented through the following mechanism. The sender marks a packet as a probe by setting a bit in its header. If the queue marks the packet it sets the CE bit. When the destination receives a packet marked as a probe without the CE bit set, it immediately send a small *no mark* UDP datagram back to the source, which waits for a suitable period of time. If the probe packet was marked, or dropped no such *no mark* packet will be received and the source infers that the probe was marked (drops are treated as equivalent to marks). Each source uses a single packet to probe the network and enters at the high rate if a no mark packet is not received back by the source; otherwise it enters at the low rate. The packet arrival streams are periodic, and thus for large marking threshold values the $M/M/1$ model of (3) is inappropriate, a better model being that in [10], which also defines the values of threshold which are considered as large.

Congestion detection was done using the virtual queue marking algorithm in routers, as described in section 2, which has one free parameter K, and marks packets (sets the CE bit) if the virtual buffer exceeds K, where $K = 18$. The router itself has a buffer capacity for 20 packets, and the virtual queue had the same maximum capacity.

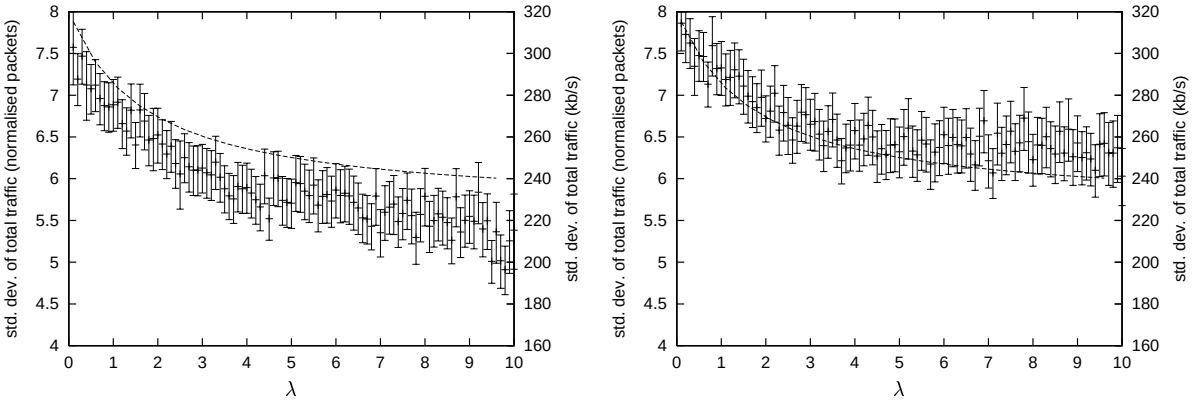

Figure 4: Pareto Call Holding times, shape parameters 1.2 (left) and 1.5 (right).

4.2 Single Link Results

In this section we use $N = 100$, and $\nu = 0.9$ in scaled units, so the real arrival rate is 90 per holding time. Link capacity was set to 4.3Mb/s. Figure 3 compares simulation with theoretical results, as the probing rate λ (plotted normalised to a unit call holding time) is altered. The theoretical results were computed using an acceptance function which was determined from simulations using the same marking strategy. The error bars show the 95% confidence intervals for the computed variance of the traffic on the link during the simulations, where each simulation represents 10^5 seconds. Notice first the excellent agreement with theory. Increasing λ from zero to four reduces the variability ($\pm 2\sigma$) by about 10% of the link capacity. For λ significantly greater than 10, further simulations showed that the variance of the total traffic increased with λ, indicating that the effect of the round trip time was no longer negligible at such high probing rates.

4.3 Sensitivity

In the theoretical analysis, the call holding times were assumed to be exponential with mean one. Figure 4 shows the effect on the standard deviation of the traffic as a function of λ, when the call holding times are heavy tailed, distributed according to a Pareto distribution with mean 1 and shape parameter $\beta = 1.2$, or 1.5. Although the initial decay in traffic variance as λ increases is slightly less rapid than in the case of exponential call holding times, for $\lambda = 4$, the variability has decreased by approximately 30%, indicating that the "in-call" probing strategy is just as useful as when the holding times were distributed according to an exponential distribution.

4.4 Star Network

Here we consider a final example of a more complex network, the precise details of which are relatively unimportant. This example mainly serves to demonstrate that the results discussed earlier apply also in more complex situations. The good comparison with simulation results for this complex system shows that the analytic model covers the essential features. The example has a simple star topology, illustrated in Figure 5. Calls originate from the end nodes (which are numbered from one to ten), and each call connects with a randomly chosen distinct edge node. Thus there are 45 possible routes. This may be

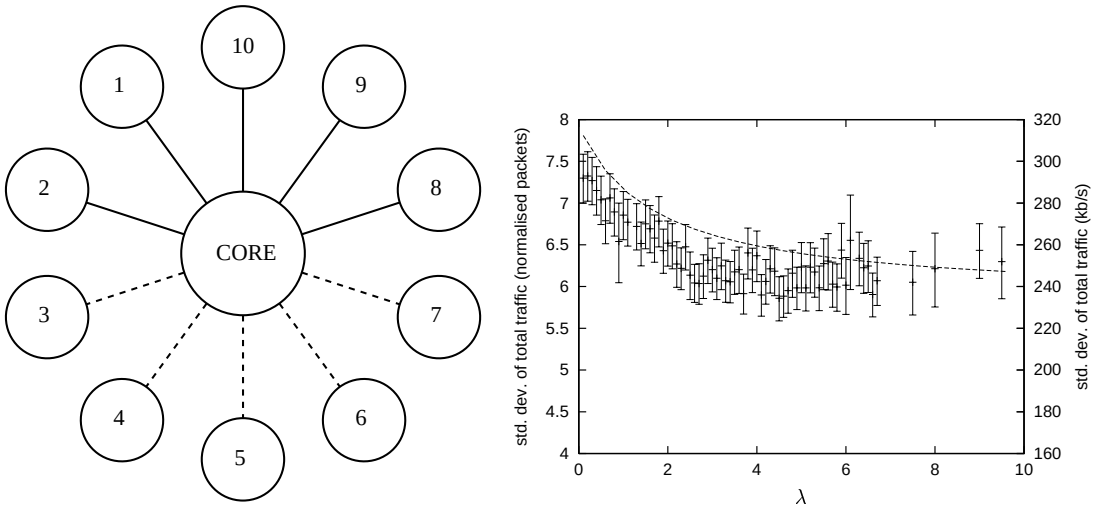

Figure 5: (Left) Star network topology. Solid lines denote duplex links with a delay of 10ms, and dashed lines those with a 100ms delay. (Right) Traffic standard deviation against probing rate λ for link between edge 1 and core. The theoretical prediction is shown by the dashed line. Further work is being done to identify the cause of the remaining discrepancy between these results.

motivated by an Internet architecture with a transparent core switch connected to routers by a variety of links. These routers are represented in the model by the edge nodes since they aggregate calls arriving from a number of individual sources.

The result of simulating this network and measuring the variance of the aggregate traffic along one of 100ms delay links is shown in Figure 5. These simulations were very time consuming (approximately 10 days CPU time); the theoretical model may readily be solved in approximately five minutes. The simulation results show a similar reduction in variance to those for the single link, however the majority of the effect has been achieved by $\lambda = 1$, which is a quarter of the value observed for a single link.

5 Mixed traffic and bandwidth sharing

We have concentrated on the case where the adaptive traffic has sole use of the resource. We now comment briefly on what happens when other traffic shares the resource. If extra traffic of load ρ (rate normalised by capacity) shares the link, then the acceptance functions a are now affected by the load ρ as well as the number of calls in progress at the different rates. In effect, with a slight abuse of notation, writing ρ_x for the load generated by x, instead of $a(\rho_x)$ we have a revised acceptance function $\tilde{a}(\rho_x) = a(\rho_x + \rho)$. More generally, the other traffic's load ρ may itself depend upon x, for example it may comprise flow-controlled streams.

When the other traffic comprises a number of TCP connections, some argue that the traffic competing with TCP should be 'TCP-friendly', receiving a throughput similar to TCP. This approach is not without its difficulties, nevertheless we indicate how the choice of acceptance functions can approximately achieve this. There are a number of approximations to TCP throughput when the packet marking rate is p (or equivalently the marking rate for ECN-aware TCP). In the absence of

time-outs the simplest relation takes the form

$$r_{TCP} \approx \frac{\texttt{MSS}}{\texttt{RTT}}\sqrt{c\frac{1-p}{p}}, \tag{11}$$

where c is a constant which we can approximate by 2, MSS is the maximum segment size, and RTT the round trip time. If the streaming traffic has high rate r_H (and possibly other lower rates), and is rejected with probability $1-a_H$, then the throughput is bounded above by $2\texttt{RTT}r_H a_L$; hence we will not prejudice TCP traffic provided

$$a_L \leq \frac{1}{2\texttt{RTT}r_L}\sqrt{2\frac{1-p}{p}}. \tag{12}$$

There are a number of ways we could choose acceptance strategies to achieve this, however a particularly simple way is to send packets at the high rate for one round trip time, generating $N = 2\texttt{RTT}a_L$ packets, and choosing to enter at the low rate provided less than m are marked in such a way that the above equation holds.

6 Concluding Remarks

We have illustrated the benefits of in-call probing for applications that can adjust their rate. The approach is light-weight – applications look at whether a small number of probing packets are marked, and use this information to decide whether or not to alter their rate. No extra signalling channel is required, and the connection set-up is very fast. We looked at the simplest possible scheme, in which all calls are accepted, and an application sends just one probe packet into the network, if this is marked it chooses the low-rate, otherwise the high rate. with the same policy used at the start of the connection and during the connection. More complex schemes involving more complex probing strategies (e.g. distinct initial acceptance and reprobing strategies) could doubtless achieve better performance. We have concentrated on the case where the application has just two admissible transmission rates; the theory applies equally well to more levels. We constructed a diffusion limit to quantify the benefits of probing, which agrees well with simulation, is tractable and can be applied to general networks. Results show that whilst the mean traffic flow on the network is unaffected by the probing, just a small amount of in-call probing significantly improves the behaviour of the system: allowing on average just one in-call change decreases the variance of the system significantly, and allowing between 5 and 10 in-call changes gives almost all of the possible gains, reducing the traffic variance by 30% in some cases. This benefits the system, and means that users also do better in the long run. So, while further benefits are clearly possible through the use of a more complex scheme, it can be seen that the simple scheme described here realises most of the potential (when compared to the $\lambda \to \infty$ limit representing continuous adaptation) without the complexity.

To implement such in-call probing, marks have to be fed back to applications, and we have discussed ways in which the ECN proposal could be suitably adapted. Alternatively, loss could be used as the feedback signal.

We have not dwelt on how adaptive traffic should be integrated with other traffic. There are a number of possibilities: such traffic could be segregated into a separate DiffServ class. If end-system behaviour is enforced in some way then soft-guarantees

could be given on packet loss and rejection. If such traffic is not segregated, then guarantees are necessarily weakened. The theory can be adapted to the integrated case by suitably altering the acceptance strategy with altered probing behaviour, such as sending more probe packets to allow for marking caused by high unresponsive load.

References

[1] R. Bellman and K. L. Cooke. *Differential-Difference Equations*. Academic Press, New York, 1963.

[2] G. Bianchi, A. Capone, and C. Petrioli. Throughput analysis of end-to-end measurement based admission control in IP. In *INFOCOM 2000*. IEEE, 2000.

[3] J.-C. Bolot and T. Turletti. Experience with control mechanisms for packet video in the Internet. *Computer Communication Review*, 28(1):4–15, 1998.

[4] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin. Resource ReSerVation Protocol (RSVP) version 1 functional specification. RFC 2205, Internet Engineering Task Force, September 1997.

[5] L. Breslau, E. Knightly, S. Shenker, I. Stoica, and H. Zhang. Endpoint admission control: Architectural issues and performance. In *Sigcomm 2000*, August 2000.

[6] V. Elek, G. Karlsson, and R. Rönngren. Admission control based on end-to-end measurements. In *INFOCOM 2000*. IEEE, 2000.

[7] S. Ethier and T. Kurtz. *Markov Processes – Characterization and Convergence*. Wiley, 1986.

[8] K. Fall and K. Varadhan. NS documentation, 2001. `http://www.isi.edu/nsnam/ns`.

[9] S. Floyd and V. Jacobson. Random Early Detection gateways for congestion avoidance. *IEEE/ACM Transactions on Networking*, 1(4):397–413, 1993.

[10] B. Hajek. A queue with periodic arrivals and constant service rate. In F. P. Kelly, editor, *In Probability, Statistics and Optimisation A tribute to Peter Whittle*, pages 147–157. Wiley, 1994.

[11] S. Hammarling and B. Wichmann. Numerical Solution of the stable, non-negative definite Lyapunov equation. *IMA J. of Num. Anal.*, 2:303–323, 1982.

[12] F. P. Kelly, P. B. Key, and S. Zachary. Distributed admission control. *IEEE Journal on Selected Areas in Communications*, 18(12):2617–2628, December 2000.

[13] T. Kelly. An ECN probe-based connection acceptance control. *Computer Communication Review*, 31(3), July 2001.

[14] T. G. Kurtz. Strong approximation theorems for density dependent markov chains. *Stochastic Process. Appl.*, 6:223–240, 1978.

[15] A. Mandelbaum, W. A. Massey, and M. I. Reiman. Strong approximations for markovian service networks. *Queueing Systems – Theory and Applications*, 30:149–201, 1998.

[16] K. Ramakrishnan and S. Floyd. A proposal to add explicit congestion notification (ECN) to IP. RFC 2481, IETF, January 1999.

[17] K. Ramakrishnan, S. Floyd, and D. Black. RFC 3168: The addition of explicit congestion notification (ECN) to IP, 2001.

[18] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RFC 1889: RTP: A transport protocol for real-time applications, 1996.

[19] D. Sisalem and H. Schulzrinne. The loss-delay based adjustment algorithm: A TCP-friendly adaptation scheme. In *Proceedings of NOSSDAV*, Cambridge, UK., 1998.

[20] D. Sisalem, H. Schulzrinne, and F. Emanuel. The direct adjustment algorithm: A TCP friendly adaptation scheme, 1997.

[21] D. Wu, Y. Hou, and Y. Zhang. Transporting real-time video over the internet. In *Proceedings of the IEEE*, volume 88, pages 1855–1875, December 2000.

[22] D. Wu, Y. T. Hou, W. Zhu, Y.-Q. Zhang, and J. Peha. Streaming video over the internet: Approaches and directions. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(3):282–300, March 2001.

Appendix

Proof of Central Limit Theorem and Weak Limit Theorem

This result will be proved in the case that the function a is a continuous Lipshitz function. The function a is, of course a not a real element of the system, but a modelling construct which describes the packet timescale behaviour of the system in a fashion which is useful at the call timescale. A simple sufficient condition for a to be Lipshitz is that $|\nabla a|$ be bounded. This result can be extended to locally Lipshitz a, but the simpler Lipshitz case illustrates the mechanism of the proof, and avoids many distracting technicalities. Let the load in the Nth system, when there are x high and y low rate calls be given by $\rho^{\,N}(x,y)$.

Proof. Let $X_t^{(1)}$ and $X_t^{(2)}$ be stochastic processes describing the number of calls in progress at the low and high data rates at time t respectively, then these satisfy the following Poisson counter driver stochastic differential equations:

$$\begin{aligned}\mathrm{d}X_t^{(1)} = \mathrm{d}A\left(\nu_N\int_0^t a(\rho_N(X_\tau^{(1)},X_\tau^{(2)}))\mathrm{d}\tau\right) - dB\left(\int_0^t X_\tau^{(1)}\mathrm{d}\tau\right)\\ + \mathrm{d}C\left(\lambda\int_0^t a(\rho_N(X_\tau^{(1)},X_\tau^{(2)}))X_\tau^{(2)}\mathrm{d}\tau\right) - \mathrm{d}D\left(\lambda\int_0^t\left(1-a(\rho_N(X_\tau^{(1)},X_\tau^{(2)}))\right)X_\tau^{(1)}\mathrm{d}\tau\right), \quad (13)\end{aligned}$$

and

$$\mathrm{d}X_t^{(2)} = \mathrm{d}\tilde{A}\left(\nu_N \int_0^t \left(1 - a(\rho_N(X_\tau^{(1)}, X_\tau^{(2)}))\right) \mathrm{d}\tau\right) - \mathrm{d}\tilde{B}\left(\int_0^t X_\tau^{(2)} \mathrm{d}\tau\right) - \mathrm{d}C\left(\lambda \int_0^t a(\rho_N(X_\tau^{(1)}, X_\tau^{(2)})) X_\tau^{(2)} \mathrm{d}\tau\right) + \mathrm{d}D\left(\lambda \int_0^t \left(1 - a(\rho_N(X_\tau^{(1)}, X_\tau^{(2)}))\right) X_\tau^{(1)} \mathrm{d}\tau\right),$$

where A, B, $\tilde{A}$, $\tilde{B}$, C, and D are mutually independent unity rate Poisson processes. We now introduce scaled versions of the processes, namely $\tilde{X}_t^{(1)} = X_t^{(1)}/N$, and $\tilde{X}_t^{(2)} = X_t^{(2)}/N$. Note that $\rho_N(xN, yN) = \rho(x, y)$, since we assumed the link capacities scaled as N. The following is then a scaled version of the SDEs:

$$\mathrm{d}\tilde{X}_t^{(1)} = \frac{1}{N}\mathrm{d}A\left(N\nu \int_0^t a(\rho(\tilde{X}_\tau^{(1)}, \tilde{X}_\tau^{(2)}))\mathrm{d}\tau\right) - \frac{1}{N}\mathrm{d}B\left(N \int_0^t \tilde{X}_\tau^{(1)} \mathrm{d}\tau\right) + \frac{1}{N}\mathrm{d}C\left(N\lambda \int_0^t a(\rho(\tilde{X}_\tau^{(1)}, \tilde{X}_\tau^{(2)}))\tilde{X}_\tau^{(2)} \mathrm{d}\tau\right) - \frac{1}{N}\mathrm{d}D\left(\lambda N \int_0^t \left(1 - a(\tilde{X}_\tau^{(1)}, \tilde{X}_\tau^{(2)})\right) \tilde{X}_\tau^{(2)} \mathrm{d}\tau\right),$$

and

$$\mathrm{d}\tilde{X}_t^{(2)} = \frac{1}{N}\mathrm{d}\tilde{A}\left(\nu \int_0^t \left(1 - a(\tilde{X}_\tau^{(1)}, \tilde{X}_\tau^{(2)})\right) \mathrm{d}\tau\right) - \frac{1}{N}\mathrm{d}\tilde{B}\left(N \int_0^t \tilde{X}_\tau^{(2)} \mathrm{d}\tau\right) - \frac{1}{N}\mathrm{d}C\left(\lambda N \int_0^t a(\tilde{X}_\tau^{(1)}, \tilde{X}_\tau^{(2)})\tilde{X}_\tau^{(2)} \mathrm{d}\tau\right) + \frac{1}{N}\mathrm{d}D\left(\lambda N \int_0^t \left(1 - a(\tilde{X}_\tau^{(1)}, \tilde{X}_\tau^{(2)})\right) \tilde{X}_\tau^{(1)} \mathrm{d}\tau\right).$$

Let $\tilde{\mathbf{X}}$ be the vector $(\tilde{X}_t^{(1)}, \tilde{X}_t^{(2)})$, this SDE can then be expressed in vector form as

$$\tilde{\mathbf{X}}^N(t) = \tilde{\mathbf{X}}^N(0) + \sum_{i \in I} A_i \left(N \int_0^t \alpha_s \left(\frac{1}{N}\tilde{\mathbf{X}}^N(s), i\right) \mathrm{d}s\right) \mathbf{v}_i.$$

where the vectors $\mathbf{v}_i$ and the rate functions α_i for $i = 1, \ldots, 6$ are defined as follows:

$$\mathbf{v}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \alpha(\mathbf{x}, 1) = \nu a(x, y) \qquad \mathbf{v}_2 = \begin{pmatrix} -1 \\ 0 \end{pmatrix} \quad \alpha(\mathbf{x}, 2) = x$$

$$\mathbf{v}_3 = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad \alpha(\mathbf{x}, 3) = \lambda a(x, y) y \qquad \mathbf{v}_4 = \begin{pmatrix} -1 \\ 1 \end{pmatrix} \quad \alpha(\mathbf{x}, 4) = \lambda(1 - a(x, y))x$$

$$\mathbf{v}_5 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \alpha(\mathbf{x}, 5) = \nu\left(1 - a(x, y)\right) \qquad \mathbf{v}_6 = \begin{pmatrix} 0 \\ -1 \end{pmatrix} \quad \alpha(\mathbf{x}, 6) = y$$

Under our assumption that a is Lipshitz, it follows that all of the other relevant rate functions (α_1 through α_6) are also Lipshitz. They are also all continuous. It is now possible through the introduction of suitable initial conditions to apply Kurtz's theorems (see [7, 14, 15]) to establish the weak convergence to the fluid limit

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \sum_{i=1}^{6} \alpha(\mathbf{x}(t), i)\mathbf{v}_i,$$

and also the CLT about the fluid limit, namely that

$$\lim_{N\to\infty} \sqrt{N}\left[\tilde{\mathbf{X}}_t^{(N)} - \mathbf{x}(t)\right] \stackrel{\mathcal{D}}{=} \mathbf{R}_t,$$

where $\mathbf{R}_t$ solves the stochastic differential equation

$$\mathrm{d}\mathbf{R}_t = \sum_{i=1}^{6} \left(\nabla\alpha(\mathbf{x}(t), i) \cdot \mathbf{R}(t)\right) \mathbf{v}_i \,\mathrm{d}t + \sum_{i=1}^{6} \sqrt{\alpha(\mathbf{x}(t), i)}\,\mathrm{d}B_i(t)\mathbf{v}_i,$$

where $B_i(t)$ is the ith component of a six dimensional standard Brownian Motion process. The most useful application of this is when the fluid limit is the time-independent equilibrium distribution, that is $\mathbf{x}(t) = (\overline{x}_1, \overline{x}_2)$, when the result simplifies to that which was stated. □