Multiscale Analysis and Control of Networks with Fractal Traffic

Warren M. Lam and Gregory W. Wornell

RLE Technical Report No. 625

October 1998

This work was supported in part by DARPA monitored by ONR
under Contract No. N00014-93-1-0686, ONR under Grant No. N00014-96-1-0930,
and AFOSR under Grant No. F49620-96-1-0072.

The Research Laboratory of Electronics
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE, MASSACHUSETTS 02139-4307

Multiscale Analysis and Control of Networks with Fractal Traffic

Warren M. Lam and Gregory W. Wornell

February 1998

Abstract

A recently-introduced multiscale framework is used to develop efficient analysis and design techniques for networks with self-similar traffic. These allow the interarrival density function for fractal point processes under Bernoulli random erasure to be determined, as well as the counting process distribution for superpositions of these processes. The results suggest that fractal characteristics are preserved under traffic branching and merging, which may, in turn, provide insight into the prevalence of self-similarity in aggregate traffic broadly observed on real networks.

Multiscale techniques are also developed for analyzing fractal queueing scenarios. These are used to obtain, as examples, the steady-state customer distribution for a memoryless queue servicing self-similar arrivals, and for Poisson customers serviced with self-similar holding times. The persistent memory inherent in the underlying point processes leads to substantially different behavior than is observed in traditional queueing scenarios, and important implications on resource consumption and quality of service are discussed.

We show how multiscale methods can be used in conjunction with dynamic programming techniques to develop efficient and practical control policies for these fractal queueing scenarios. In particular, optimal server control is developed for a memoryless queueing system with self-similar traffic input, and optimal flow control is formulated for self-similar service of memoryless traffic. By exploiting past history, these controllers achieve substantially better performance—both in terms of quality of service and resource utilization—than traditionally used queueing control strategies.

Index Terms—fractals, point processes, queueing theory, optimal control, teletraffic, networks, self-similar traffic, dynamic programming

1 Introduction

Point processes and queueing systems with fractal properties are increasingly being viewed as important models in a host of communication network applications. In particular, self-similarity in point processes is well-matched to the burstiness observed in many aspects of such networks. A brief listing includes error occurrence on a number of telecommunication

This work has been supported in part by DARPA monitored by ONR under Contract No. N00014-93-1-0686, ONR under Grant No. N00014-96-1-0930, and AFOSR under Grant No. F49620-96-1-0072.

The authors are with the Department of Electrical Engineering and Computer Science, and the Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, MA 02139.

links, arrival patterns of many forms of data such as compressed video, as well as the aggregate traffic on a wide range of networks like local ethernets and the Internet as a whole [1, 2, 3, 4]. At the same time, queueing models with self-similar properties are equally promising, capturing many aspects of packet networks for which traditional memoryless models are overly simplistic. For example, power-law holding times of fractal queues are well-matched with the propagation delay associated with heavy-tailed packet-size distributions.

Motivated by the ubiquity of scale-free event distributions in these and a variety of other applications, a number of fractal point process models have been developed and explored in the literature [3, 5, 6, 7]. In this paper, we develop a multiscale framework for the analysis and design of networks involving self-similar point processes and queues. Our resulting analysis results lead to techniques for predicting the impact that the presence of self-similar distributions has on the performance of telecommunications networks. In turn, our subsequent investigation of network design focuses on methods of optimal control of network activities. Collectively, these results have significant implications in the optimal structuring and management of both existing and future networks.

The rest of this paper is organized as follows. In Section 2, we summarize key aspects of the fractal point process model and an associated multiscale framework developed in [8], which forms the basis for our current development. In Section 3, we introduce a Markovian interpretation for this multiscale framework, which is well suited for determining the counting process statistics for this process. In Section 4, we apply this framework to the analysis of key network activities involving fractal traffic, which include interaction among multiple traffic streams and queueing of fractal processes. In Section 5, we develop extensions of our multiscale framework to allow us to design control policies for a number of fractal queues, including optimal server and flow control. Finally, Section 6 contains some concluding remarks.

2 Fractal Renewal Processes

A point process—i.e., a random distribution of event arrivals in time—is naturally characterized in terms of its interarrival intervals $X[i]$. More precisely, we let $X[1]$ denote the arrival epoch of the first arrival after $t = 0$, and $X[i]$ denote the time interval between the $(i-1)$st arrival and the ith arrival, for $i \geq 2$. Other aspects of a point process are revealed by its characterization in terms of the associated counting process $N(t)$, whose value at time t is defined as the number of arrivals in the interval $(0, t]$. Since the counting process describes the history of the point process, its value at any instant t has dependence on the choice of the reference point $t = 0$. Two scenarios are of special interest. In the arrival-observed case, the reference point coincides with an event. In the context of networking studies, for example, this can represent data traffic as viewed by a user. On the other hand, the random incidence case corresponds to the point of reference being chosen randomly, independent of the point process. As such, random incidence is useful for modeling traffic from the perspective of network management, for example.

The fractal point process model of interest is defined in terms of a particular self-similarity property. Specifically, the associated counting process satisfies $N(t) \stackrel{\mathcal{P}}{=} N(at)$ for all $a > 0$, where the notation $\stackrel{\mathcal{P}}{=}$ denotes statistical equality, in particular in the sense of all finite-dimensional distributions.

Much of the physical network behavior of interest is effectively stationary, exhibiting no preference for a time origin. As such, it is tempting to restrict attention to self-similar point processes that are also renewal processes. However, a true renewal process cannot be self-similar [8]. Nevertheless, it is possible to develop a fractal point process model based on a generalized notion of renewal process. To develop this notion, we first introduce the following convenient terminology: we say that a point process with interarrivals $Y[i]$ is derived from a point process with interarrivals $X[i]$ via *conditioning on the event* $\mathcal{E}$ if $Y[i]$ is the subsequence of $X[i]$ formed by discarding those components such that $X[i] \notin \mathcal{E}$. As developed in [8], a *fractal renewal process* is then defined as a self-similar point process that satisfies the following:

1. When conditioned on the event $\mathcal{E} = \{\underline{x} < X \leq \overline{x}\}$ for some $0 < \underline{x} < \overline{x} < \infty$, the

resulting point process is a renewal process; and

2. When conditioned on each of any number of arbitrary, mutually exclusive events $\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_L$ such that $\mathcal{F}_l = \{\underline{x}_l < X \leq \overline{x}_l\}$ with $0 < \underline{x}_l < \overline{x}_l < \infty$, the resulting point processes are mutually independent.

Since observations of physical signals are typically limited both in resolution and duration, Property 1 implies that a fractal renewal process is effectively a renewal process. Furthermore, it can be shown that as a consequence of self-similarity, the interarrivals $Y[i]$ resulting from such observations must be distributed as [8]

$$f_Y(y) \propto \frac{1}{y^\gamma}, \quad \underline{x} < y < \overline{x} \tag{1}$$

where $\underline{x}$ and $\overline{x}$ are determined by the resolution and duration windowing, respectively. The shape parameter γ in (1) is directly related to the fractal dimension D of the process via $\gamma = D + 1$. Fig. 1 illustrates a typical sample function of a fractal renewal process with shape parameter $\gamma = 1.5$, viewed under successive magnification, from which the hallmark scale-independent clustering behavior is apparent.

Analysis of the fractal renewal process is facilitated by a highly efficient Poisson-based multiscale representation developed in [8].[1] The essence of this framework is to model a fractal renewal process as a mixture of a multiscale family of Poisson processes. In terms of interarrival statistics, this is equivalent to decomposing the power-law interarrival distribution of (1) as a weighted sum of dilated and compressed exponential functions. In particular, with a finite number of constituents, the interarrival decomposition takes the form

$$f_X(x) = \sum_{j=0}^{L-1} p_j f_{X_j}(x), \tag{2}$$

[1] In fact, such representations naturally lead to computationally-efficient algorithms for robust estimation of the scaling parameters of such processes, as also developed in [8].

where the X_j are exponential random variables with rate parameters

$$\lambda_j = \frac{\lambda}{\eta^j}, \tag{3}$$

while the p_j form a geometric probability distribution

$$p_j = \sigma^2 \left(\eta^{1-\gamma}\right)^j, \tag{4}$$

with σ^2 as a normalization constant. Spacing between the constituents is governed by the scale increment $\eta > 1$, while the number of scales is equal to L, with scale 0 being the finest scale.

3 The Multiscale Pure-Birth Process

The preceding multiscale model has a natural interpretation in the form of a multiscale pure-birth process. This process, depicted in Fig. 2, forms the basis of our development. Generalizing the well-known pure-birth process (see, e.g., [9]), the state space of this Markov process is naturally partitioned into "superstates," each of which represents a certain number of births. A superstate is, in turn, composed of a set of states which correspond to the scales in our finite-scale framework. Hence, we index the states with a pair of integers (i, j), where the superstate index i is nonnegative, while the scale index j ranges from 0 to $L-1$ for an L-scale representation. We define the probability vector $\mathbf{p}_i(t) \triangleq (P_{i,0}(t), P_{i,1}(t), \ldots, P_{i,L-1}(t))$, with $P_{i,j}(t)$ denoting the probability that the process is in state (i, j) at time t. With this notation, the counting process probability distribution can be expressed as

$$\Pr\{N(t) = i\} = \mathbf{p}_i(t)\mathbf{1}^{\mathrm{T}}, \tag{5}$$

where $\mathbf{1}$ is a row vector with all entries equal to 1. For convenience, we use the notation $\pi_i(t) \triangleq \Pr\{N(t) = i\}$.

Every transition in the multiscale pure-birth process results in an increment in the superstate index, and thus a birth. In our model, the birth rate is assumed independent of the number of births already occurred. Thus, the mean departure rate from each state

(i,j) is only a function of the scale j, taking the specific form $\lambda_j \triangleq \lambda/\eta^j$ in terms of the scale increment η and the reference rate λ. Upon a birth, every state j' of the succeeding superstate can be immediately reached, with probability $p_{j'} \triangleq \sigma^2 q^{j'}$, where $q \triangleq \eta^{1-\gamma}$ governs the relative weighting on the scales, while σ^2 is the normalization factor. Thus, the duration between consecutive births is governed by a probability law of the form (2).

The counting process associated with the fractal renewal process can be readily obtained from the multiscale pure-birth representation. Inspecting Fig. 2, it is straightforward to set up the forward Kolmogorov equations [9] for the multiscale pure-birth process. In vector form, we have that

$$\frac{1}{\lambda}\frac{d}{dt}\mathbf{p}_0(t) = -\mathbf{p}_0(t)\mathbf{B} \tag{6a}$$

$$\frac{1}{\lambda}\frac{d}{dt}\mathbf{p}_i(t) = -\mathbf{p}_i(t)\mathbf{B} + \mathbf{p}_{i-1}(t)\mathbf{b}^{\mathrm{T}}\mathbf{q}, \qquad i \geq 1 \tag{6b}$$

where $\mathbf{q} \triangleq (p_0, p_1, \ldots, p_{L-1})$ contains the choice probabilities, $\mathbf{b} \triangleq (\eta^0, \eta^{-1}, \ldots, \eta^{-(L-1)})$ is a vector of the multiscale dilation factors, and $\mathbf{B} \triangleq \mathrm{diag}(\mathbf{b})$ is a diagonal matrix with the dilation factors along its main diagonal. From (6) we obtain the transform-domain equation

$$\frac{1}{\lambda}\frac{d}{dt}\hat{\mathbf{p}}(z;t) = \hat{\mathbf{p}}(z;t)\left(-\mathbf{B} + z\mathbf{b}^{\mathrm{T}}\mathbf{q}\right), \tag{7}$$

where $\hat{\mathbf{p}}(z;t)$ denotes the z-transform of the sequence $\{\mathbf{p}_i(t); i = 0, 1, \ldots\}$, defined as

$$\hat{\mathbf{p}}(z;t) \triangleq \sum_{i=0}^{\infty} z^i \mathbf{p}_i(t).$$

Eq. (7) can now be readily solved to yield

$$\hat{\mathbf{p}}(z;t) = \hat{\mathbf{p}}(z;0)\exp\left(\left[-\mathbf{B} + z\mathbf{b}^{\mathrm{T}}\mathbf{q}\right]\lambda t\right), \quad t \geq 0, \tag{8}$$

which determines the z-transform up to an initial condition.

In the arrival-observed case, where the point of reference $t = 0$ is a renewal, the first interarrival is statistically identical to every other interarrival. Hence, the scales within the zeroth superstate are chosen with probabilities $\{p_j; j = 0, 1, \ldots, L-1\}$, and the initial

condition in (8) is $\hat{\mathbf{p}}(z;0) = \mathbf{q}$. Inverting the resulting transform $\hat{\mathbf{p}}(z;t)\mathbf{1}^{\mathrm{T}}$, we can obtain a time-domain characterization of the arrival-observed counting process distribution, which we denote by $\left\{\pi_i^{(\mathrm{a})}(t); i = 0, 1, \ldots\right\}$. In particular,

$$\pi_0^{(\mathrm{a})}(t) = \mathbf{p}_0(t)\mathbf{1}^{\mathrm{T}} = \hat{\mathbf{p}}(0;t)\mathbf{1}^{\mathrm{T}} = \mathbf{q}\exp(-\mathbf{B}\lambda t)\mathbf{1}^{\mathrm{T}} = \sum_{i=0}^{L-1} p_i \exp\left(-\lambda_i t\right), \tag{9}$$

which decays as $1/t^{\gamma-1}$ for large t when $L \to \infty$. Higher-order terms can be obtained numerically from a Taylor series expansion of the transform $\hat{\mathbf{p}}(z;t)\mathbf{1}^{\mathrm{T}}$, as developed in detail in Appendix A. The main results are that for $i \geq 1$,

$$\pi_i^{(\mathrm{a})}(t) = a_i^{(i)}\frac{(\lambda t)^i}{i!} - a_{i+1}^{(i)}\frac{(\lambda t)^{i+1}}{(i+1)!} + \cdots,$$

with the first-order coefficients given by

$$a_k^{(1)} = \sum_{l=1}^{k} M_l M_{k-l} \tag{10}$$

and the higher-order coefficients obtained via the recurrence relation

$$a_k^{(i)} = \sum_{l=1}^{k-i+1} M_l a_{k-l}^{(i-1)}, \tag{11}$$

where M_l is the lth moment of a random variable R distributed according to

$$\Pr\{R = \eta^{-j}\} = p_j, \quad j = 0, 1, \ldots, L-1. \tag{12}$$

For random incidence, we assume observation begins at a random time, with the point process already in equilibrium. Under this assumption, the scale of the first interarrival is selected with the steady-state marginal distribution over scales, which is $\tilde{\sigma}^2\mathbf{q}\mathbf{B}^{-1}$, where $\tilde{\sigma}^2$ is for normalization; an argument is given in Appendix B. Using this in (8), we can obtain a time-domain characterization of the random incidence counting process distribution which we denote by $\left\{\pi_i^{(\mathrm{r})}(t); i = 0, 1, \ldots\right\}$. For example, it is straightforward to get the closed-

form expression

$$\pi_0^{(r)}(t) = \tilde{\sigma}^2 \sum_{i=0}^{L-1} \frac{p_i}{\lambda_i} \exp(-\lambda_i t), \tag{13}$$

which when $L \to \infty$ approaches 1 if $\gamma < 2$ and decays like $1/t^{\gamma-2}$ for large t if $\gamma \geq 2$. Higher-order terms of the probability distribution take the form

$$\pi_i^{(r)}(t) = r_i^{(i)} \frac{(\lambda t)^i}{i!} - r_{i+1}^{(i)} \frac{(\lambda t)^{i+1}}{(i+1)!} + \cdots,$$

where the coefficients are obtained from those in the arrival-observed distribution via

$$r_k^{(i)} = (a_{k-1}^{(i)} + a_k^{(i-1)})\tilde{\sigma}^2. \tag{14}$$

Details of the derivation are given in Appendix B.

Figs. 3 and 4 illustrate the time evolution of the lower order terms in the arrival-observed and random incidence counting process distributions for the case $\gamma = 1.8$. It is worth remarking that in practice, very few scales are typically needed for a good approximation to the probabilities over a finite time interval, although more scales are generally required in small γ situations due to the more persistent tail.

Several features of the plots are noteworthy. First, in the arrival-observed case of Fig. 3, that the probabilities $\pi_i(t)$ all fall off along the same $1/t^{\gamma-1}$ asymptote implies that $E[N(t)] \sim O(t^{\gamma-1})$. These statistics are consistent with the strong clustering behavior characteristic of these processes, which is increasingly pronounced as γ decreases. By contrast, for the memoryless (Poisson) process, the counting process probabilities fall off exponentially quickly and $E[N(t)] \sim O(t)$. Moreover, the power-law probability decay is also consistent with the results from direct computation of the counting process statistics [10], which involves asymptotic analysis of convolution of multiple power-law functions. In addition, the counting process distribution depicted in Fig. 4 for the random incidence case provides dramatic evidence of the impact of the unusually long quiescent periods between clusters in fractal point processes.

We turn now from the analysis of fractal point processes in isolation to analysis of their

behavior in scenarios representative of those encountered by traffic in large, interconnected networks.

4 Network Analysis for Fractal Traffic

Key activities in a communication network can typically be modeled in terms of transformations on point processes. Interaction among traffic streams, for example, can often be modeled as branching and merging of point processes. Propagation delay and processing latency at sites and links can typically be captured by queueing models. In this section, we apply multiscale techniques to analyze a number of these activities. Branching and erasure of fractal renewal processes will be addressed in Section 4.1, while fractal renewal process superposition will be the theme of Section 4.2. Queueing systems with self-similar properties will be explored in Sections 4.3 and 4.4.

4.1 Random Erasure of the Fractal Renewal Process

Data loss and branching of data streams are two of the most frequently encountered activities in a network. An often realistic and widely-adopted model for both transformations is Bernoulli point process erasure, whereby each point is independently erased with a common probability p. In what follows, we study the behavior of the fractal renewal process under this mode of erasure. A key result of our analysis is the preservation of self-similar characteristics even under very high erasure probabilities.

To determine interarrival density under Bernoulli erasure, we begin by observing that with probability $1-p$, an arrival of the original process contributes a count of unity to the counting process resulting from erasure. Otherwise, it contributes a count of zero. In the transform domain, this corresponds to the usual replacing of z with $p+(1-p)z$ in the z-transform of the original counting process distribution. In particular, using (8), the counting process distribution of an erased fractal renewal process has the z-transform

$$\hat{\mathbf{p}}(p+(1-p)z;t)\mathbf{1}^{\mathrm{T}} = \hat{\mathbf{p}}(z;0)\exp\left(\left[-\mathbf{B}+(p+(1-p)z)\mathbf{b}^{\mathrm{T}}\mathbf{q}\right]\lambda t\right)\mathbf{1}^{\mathrm{T}}. \tag{15}$$

With the initial condition $\hat{\mathbf{p}}(z;0)=\mathbf{q}$, and with $z=0$ in (15), the arrival-observed proba-

bility of zero arrivals in an interval $(0, w]$ is

$$\tilde{\pi}_0^{(a)}(w) = \mathbf{q} \exp\left(\left[-\mathbf{B} + p\mathbf{b}^{\mathrm{T}}\mathbf{q}\right] \lambda w\right) \mathbf{1}^{\mathrm{T}}. \tag{16}$$

But this event is equivalent to the event $\{W \geq w\}$, where W is the interarrival beginning at 0. Differentiating (16), we have the interarrival density

$$\begin{aligned} f_W(w) = -\frac{d}{dw} \Pr\{W \geq w\} &= \mathbf{q} \exp\left(\left[-\mathbf{B} + p\mathbf{b}^{\mathrm{T}}\mathbf{q}\right] \lambda w\right) \left(\mathbf{B} - p\mathbf{b}^{\mathrm{T}}\mathbf{q}\right) \mathbf{1}^{\mathrm{T}} \\ &= \lambda(1-p)\mathbf{q} \exp\left(\left[-\mathbf{B} + p\mathbf{b}^{\mathrm{T}}\mathbf{q}\right] \lambda w\right) \mathbf{b}^{\mathrm{T}}, \quad w \geq 0. \end{aligned} \tag{17}$$

Using (17), we have plotted in Fig. 5 the interarrival density of a fractal renewal process with shape parameter $\gamma = 1.8$, subject to various erasure probabilities. These plots suggest that the erased interarrival density largely retains the power-law characteristics of the original density (i.e., the $p = 0$ case). In fact, for $p < 1$, empirical studies suggest that as $\lambda \to \infty$, $L \to \infty$,

$$f_W(w) \sim \frac{1}{w^{\gamma}}$$

for every $w > 0$.

4.2 Superposition of Fractal Renewal Processes

Merging of data streams is another point process transformation typical in networks. To investigate the behavior of fractal point processes under merging, we consider the superposition of two independent fractal renewal processes. A key implication of our results is the invariance of fractal point process features under superposition. More importantly, our results also suggest that the family of fractal point processes constitutes a domain of attraction under aggregation, much like the Poisson family. This behavior is consistent with the spectral analysis results obtained in [6]. Together with the random erasure results, these superposition results may prove useful in explaining empirical observations of the ubiquity of self-similarity in aggregate traffic on a broad range of networks.

When two point processes are superimposed, their counting processes add. From an

arrival's viewpoint, the structure of the two counting process distributions are inherently different. On one hand, the process to which it belongs is governed by the arrival-observed distribution; we denote this distribution by $\left\{\pi_i^{(1,\mathrm{a})}(t); i = 0, 1, \ldots\right\}$. On the other hand, since the constituents are independent, the arrival observes the random incidence counting processing distribution, denoted as $\left\{\pi_i^{(2,\mathrm{r})}(t); i = 0, 1, \ldots\right\}$, for the other point process. Thus, the overall counting process distribution will be a discrete convolution of $\pi_i^{(1,\mathrm{a})}(t)$ and $\pi_i^{(2,\mathrm{r})}(t)$. If, in addition, the two constituents have the same fractal dimension, the resulting counting process distribution will be expressed more simply as

$$\tilde{\pi}_i^{(\mathrm{a})}(t) = \sum_{j=0}^{i} \pi_j^{(\mathrm{a})}(t)\pi_{i-j}^{(\mathrm{r})}(t), \quad i = 0, 1, \ldots. \tag{18}$$

By a similar argument, the random incidence counting process distribution of the superposition is the convolution of the two constituent random incidence counting process distributions,

$$\tilde{\pi}_i^{(\mathrm{r})}(t) = \sum_{j=0}^{i} \pi_j^{(\mathrm{r})}(t)\pi_{i-j}^{(\mathrm{r})}(t), \quad i = 0, 1, \ldots. \tag{19}$$

Figs. 6 and 7 show the arrival-observed and random incidence counting process distributions corresponding to the superposition of two independent fractal renewal processes, each with shape parameter $\gamma = 1.8$. The computations were performed according to (18) and (19), respectively, using the counting process distribution results of Section 3. Comparing this set of plots with those for a single process (Figs. 3 and 4), we observe that key features such as the asymptotic power-law decay in the arrival-observed distribution, and dominance of zeroth-order term in the random incidence distribution, are largely preserved under superposition, suggesting invariance of fractal renewal processes under this transformation. These counting process results provide additional verification of the invariance of fractal point proceses under superposition, complementing the spectral evidence given in [6].

4.3 Queueing of the Fractal Renewal Process

While random erasure and superposition capture key interaction among multiple data streams, queueing models are appropriate for activities at individual sites and channels. More generally, queueing analysis is important in many applications involving resource sharing. The multiscale pure-birth model can be readily extended to support queueing scenarios. For example, addition of backward transitions, or deaths, results in the multiscale birth-and-death process of Fig. 8, which models a fractal renewal process being serviced by a single-server, memoryless queue. The added transitions model service completion and customer departure and therefore all occur at the mean service rate μ. Because the service being modeled is memoryless, the multiscale birth-and-death model is applicable to work-conservative queues with arbitrary service discipline.

As an illustration of the types of results that can be obtained with this Markov model, we develop systematic and efficient methods to compute the steady-state probability distribution of the number of customers in the system $\{\pi_i; i = 0, 1, \dots\}$, which is particularly insightful for buffer allocation and predicting quality of service, for example. The dynamics of the process in Fig. 8 are governed by the system of forward Kolmogorov equations:

$$\frac{1}{\lambda}\frac{d}{dt}\mathbf{p}_0(t) = -\mathbf{p}_0(t)\mathbf{B} + \frac{1}{\rho}\mathbf{p}_1(t)\mathbf{I} \tag{20a}$$

$$\frac{1}{\lambda}\frac{d}{dt}\mathbf{p}_i(t) = -\mathbf{p}_i(t)\left(\mathbf{B} + \frac{1}{\rho}\mathbf{I}\right) + \mathbf{p}_{i-1}(t)\mathbf{b}^{\mathrm{T}}\mathbf{q} + \frac{1}{\rho}\mathbf{p}_{i+1}(t)\mathbf{I}, \quad i \geq 1, \tag{20b}$$

where $\rho \triangleq \lambda/\mu$. To obtain the equilibrium state distribution $\{\mathbf{p}_i; i = 0, 1, \dots\}$, we employ the matrix-geometric methods of [11]. By this, we first obtain a positive semi-definite solution $\mathbf{R}$ to the quadratic equation

$$\mathbf{0} = \rho\mathbf{b}^{\mathrm{T}}\mathbf{q} - \mathbf{R}(\rho\mathbf{B} + \mathbf{I}) + \mathbf{R}^2. \tag{21}$$

This can be accomplished via successive approximation (see [11]). The steady-state probabilities are then given by

$$\mathbf{p}_i = \mathbf{x}\mathbf{R}^i, \quad i = 0, 1, \dots, \tag{22}$$

where $\mathbf{x}$ is the (unique) left null vector of $\rho\mathbf{B} - \mathbf{R}$, normalized by the constraint

$$\mathbf{x}(\mathbf{I} - \mathbf{R})^{-1}\mathbf{1}^{\mathrm{T}} = 1. \tag{23}$$

Finally, the steady-state customer distribution can be obtained via $\pi_i = \mathbf{p}_i\mathbf{1}^{\mathrm{T}}$.

These methods are a much more computationally practical means for predicting queue performance than are simulation-based approaches. In Fig. 9, the analytical method for determining the steady-state distribution for the number of customers in the system with service rate $1/\rho = \mu/\lambda = 0.15$ and shape parameters $\gamma = 1.5, 1.8$ is compared with that obtained by simulations using the next-event time advance methods of [12]. Despite large sample sizes, discrepancy between the theoretical and simulation results remains largely as a consequence of substantial outliers in the simulations. Such outliers are characteristic of the heavy-tailed distributions involved.

Both the theoretical and simulation results suggest that conditioned on one or more customers, the distribution approaches a geometric function. This is indeed the case, and can be attributed to the fact that the solution to (21) is of rank 1. In particular, it can be shown that the solution must be of the form

$$\mathbf{R} \triangleq \mathbf{b}^{\mathrm{T}}\mathbf{w}, \tag{24}$$

which implies that the steady-state customer distribution satisfies

$$\frac{\pi_{i+1}}{\pi_i} = \mathbf{w}\mathbf{b}^{\mathrm{T}} \triangleq k, \quad i = 1, 2, \ldots. \tag{25}$$

Thus, k is the decay rate of the geometric portion of the distribution. From (21), we get that $\mathbf{w}$ in (24) must satisfy

$$\rho\mathbf{q} = \mathbf{w}\left(\rho\mathbf{B} + \mathbf{I}\right) - \mathbf{w}k = \mathbf{w}\left[\rho\mathbf{B} + (1 - k)\,\mathbf{I}\right].$$

Right multiplication of both sides by $[\rho\mathbf{B} + (1-k)\mathbf{I}]^{-1}\,\mathbf{b}^{\mathrm{T}}$ leads to

$$\sigma^2\left[\frac{\rho}{\rho+(1-k)} + \frac{q\rho/\eta}{\rho/\eta+(1-k)} + \cdots + \frac{q^{L-1}\rho/\eta^{L-1}}{\rho/\eta^{L-1}+(1-k)}\right] = k, \tag{26}$$

which can be solved iteratively using bisection search techniques, for instance.

A useful bound for k can be obtained via Jensen's inequality. Specifically, the left hand side of (26) is less than or equal to

$$\frac{\rho E[R]}{\rho E[R]+(1-k)}, \tag{27}$$

where R is the random variable defined in (12). Simplifying, we get that

$$k(1-k) \leq \rho E[R](1-k),$$

which, since $k < 1$ for an ergodic queue, simplifies to

$$k \leq \rho E[R] = \frac{\lambda}{\mu} E[R]. \tag{28}$$

Fig. 10 shows values of k obtained via bisection-search solution of (26), together with the bounds as prescribed by (28). As expected, higher service efficiency (μ/λ) leads to less congested system, as reflected by the lower k (or, sharper decay in the customer distribution). These plots also suggest that while the bound in (28) is somewhat loose for slow-service scenarios, it is a potentially useful closed-form approximation for k in the fast-service regime.

Another important feature of the distributions in Fig. 9 is the dominance of the probability of zero customers (i.e., idle system). This can again be attributed to the unusually long gaps between clusters of customer arrivals. This behavior suggests that service rate may be lowered for large time intervals, with no noticeable degradation in the quality of service. In Section 5.1, we show that this can indeed be accomplished via dynamic server control.

4.4 Power-Law Service of the Poisson Process

While the system of Section 4.3 addresses self-similarity in the arrivals, fractal characteristics can often be found in other aspects of a queueing scenario. For example, the distribution of packet sizes in many networks is often well-modeled as a power law, with frequent short packets and occasional extraordinarily long packets. The power-law holding time arising for this packet distribution can be modeled with the Markov process of Fig. 11, obtained by transposing the multiscale birth-and-death process of Fig. 8. Thus, the input process is now Poisson with arrival rate λ, while the service is captured by an L-scale representation with finest-scale service rate μ. Since the service is no longer memoryless, this model is restricted to queueing disciplines with no preemption. As such, various forms of time-slicing are precluded, for example. Nevertheless, many important disciplines are still captured, such as first-in-first-out, last-in-first-out, and service in random order.

To obtain the steady-state customer distribution of this queueing system, we again use the matrix-geometric method of [11]. In particular, we solve the matrix quadratic equation

$$\mathbf{0} = \rho\mathbf{I} - \mathbf{R}(\rho\mathbf{I} + \mathbf{B}) + \mathbf{R}^2\mathbf{b}^{\mathrm{T}}\mathbf{q} \tag{29}$$

for R via successive approximation, where as before, $\rho \stackrel{\Delta}{=} \lambda/\mu$. Once $\mathbf{R}$ is obtained, the steady-state customer distribution can be generated via (22) and (23). However, unlike (21) in the system of Section 4.3, the solution to (29) is not rank 1, precluding further simplication.

Two sets of results are plotted in Fig. 12, based on the above computaton and next-event time advance simulation. For the top curves, the input arrival rate is $\rho = 2 \times 10^{-7}$, while the shape parameter of the service duration is $\gamma = 1.5$. For the bottom curves, the arrival rate is set at $\rho = 1 \times 10^{-5}$, while the shape parameter of the service time is $\gamma = 1.8$. Agreement between our prediction and simulation results is close.

The figure also suggests that a key feature of this queueing scenario is the heavy-tailed customer distribution, which asymptotically approaches a power law. An implication of this is the requirement of large buffers for its implementation. In addition, long delay and hence poor quality of service can be expected for the customers if a first-in-first-out (FIFO)

discipline is adopted. In Section 5.2.1, we develop an optimal dynamic queueing control strategy to improve performance of this system.

5 Network Control for Fractal Traffic

The analysis in Sections 4.3 and 4.4 focusses on the behavior of queues with fixed system parameters. However, constant-rate memoryless service of input processes with self-similar arrivals can result in tremendously inefficient utilization of resources, as reflected by very high idle probability. Similarly, poor quality of service can result for service of a constant-rate Poisson input when the holding-time statistics are self-similar, which manifests itself in the form of a heavy-tailed queue-length distribution.

In many realistic scenarios, however, various aspects of a queueing system are controllable, often in real time. For example, controllability of service rate is quite feasible. In communications engineering, a number of schemes have been proposed for versatile allocation of bandwidth, ranging from fast packet switching networks [13] to flexible assignment of multilevel trunks and trunk groups [14]. Likewise, control of input arrival rate, or flow control, is also possible in many situations. Specifically, input throttling, or arrival rate reduction, can be implemented via admission toll or traffic re-routing.

Dynamic programming [15, 16] is a natural tool for developing queueing policies. Indeed, dynamic programming has been used to develop a number of optimal controllers for the M/M/m family of queues, i.e., queueing systems with memoryless input and finitely many (m) memoryless servers. For example, it is straightforward to develop server control for the M/M/1 queue that optimizes a long-term cost of two components, corresponding to the costs of service and buffer occupancy, respectively [17]. In this scenario, it is well-known that the optimal service rate at any time depends only on the number of customers in system. Further, under rather broad conditions, the optimal service rate increases monotonically with the number of customers present.

Similar techniques can be used to design optimal flow control for the M/M/1 queue [17]. In this case, the admission rate is designed to minimize a cost function composed of a cost on buffer occupancy and a penalty for input throttling. As in the server control problem,

optimal strategy for this problem depends only on the number of customers present, and the results agree closely with intuition: under rather broad conditions, customer arrivals are increasingly deterred as the queue becomes more crowded.

That the optimal server and flow control policies for M/M/m systems do not exploit history of the system and are completely determined by the current state is a direct consequence of the memoryless nature of such systems. In contrast, we show in this section that optimal control strategies for fractal queues depend heavily on past history to maintain efficient operation due to the long-term dependence in self-similar point processes. These controllers, which we develop by applying dynamic programming techniques within our multiscale modeling framework, significantly improve performance in terms of both resource usage and quality of service over controllers that do not exploit past history.

5.1 Server Control for the Fractal Renewal Process

We first consider the optimal control of the queueing system of Section 4.3, where a single memoryless server processes a fractal renewal process input. As is common in other state-space control problems, we develop optimal server control policies for this system based on a two-step approach: we first develop ideal controllers that rely on complete state information, then develop practical controllers by replacing the state information with suitably defined state estimates. The first of these subproblems is addressed in Section 5.1.1; the second is treated in Section 5.1.2.

5.1.1 State-Based Multiscale Server Control

To model the behavior of this system, we again employ the multiscale birth-and-death model of Fig. 8. However, the service rate μ is now assumed controllable and can be varied over an achievable range $0 \leq \mu \leq \overline{\mu}$ based on the state of the queue. To reflect its state dependence, we use the notation $\mu_{i,j}$ to denote the service rate when i customers are present and the scale of the next interarrival is j [i.e., when the state is (i, j)].

A fundamental tradeoff exists in the operation of this queue. On one hand, as pointed out in Section 4.3, high service rates will result in inefficiently used server, as reflected by the substantial idle probability. On the other hand, if the service rate is excessively reduced,

the quality of service can severely decline. In the extreme case, the system will become non-ergodic and the queue will grow without bound.

To facilitate quantitative analysis, we assign costs to these two criteria. As a measure of resource consumption, we define a service cost $c(\mu)$, which is a continuous, nondecreasing function of the service rate. For convenience, an idle server is assumed to inflict zero cost, i.e., $c(0) = 0$. To capture quality of service, we define a holding cost $h(i)$, which is nondecreasing in i, the number of customers in the system. This cost is directly related to the waiting time experienced by each customer under the first-in-first-out (FIFO) discipline, for example. For convenience, an empty system is assumed cost-free, i.e., $h(0) = 0$. The overall objective function is then the expected value of the combined costs, accumulated over time:

$$J = E\left[\int_0^\infty e^{-\beta t}\left[h(i(t)) + c(\mu(t))\right] dt\right], \tag{30}$$

where $i(t)$ is the number of customers in the queueing system at time t, and $\mu(t)$ is the service rate at time t. The discount rate β is included to allow weighting of future costs with respect to the present.

To exploit results from discrete-time dynamic programming, we recast our continuous-time Markov decision problem into an equivalent discrete-time one. We begin by adding self-transitions in the multiscale birth-and-death process so that the total departure rate from each state is $\Omega \triangleq \lambda + \overline{\mu}$, i.e., the maximum transition rate in the original process. This yields the equivalent Markov process shown in Fig. 13. Since the departure rates are now uniform across all states, this step is sometimes referred to as rate uniformization [17].

Next, we rewrite the cost in (30) as

$$J = E\left[\sum_{n=0}^{\infty} \int_{t_n}^{t_{n+1}} e^{-\beta t} dt \left(h(i[n]) + c(\mu[n])\right)\right], \tag{31}$$

where $\{t_n; n = 0, 1, \dots\}$ are the transition epochs of the uniformized process, and $i[n]$ and $\mu[n]$ are respectively the number of customers and the service rate during the nth interval, $(t_n, t_{n+1}]$. Carrying out the integration in (31) and taking expectation, we get that the cost

is

$$J = \sum_{n=0}^{\infty} E\left[\frac{e^{-\beta t_n} - e^{-\beta t_{n+1}}}{\beta}\right] E\left[h(i[n]) + c(\mu[n])\right]$$
$$= \frac{1}{\beta+\Omega} \sum_{n=0}^{\infty} \frac{\Omega^n}{(\beta+\Omega)^n} E\left[h(i[n]) + c(\mu[n])\right], \tag{32}$$

where the second equality follows from the fact that t_n is an nth-order Erlang random variable with mean n/Ω. The final expression in (32) is the objective function of a discrete-time dynamic programming problem for a Markov chain with the same topology as the process in Fig. 13, where for any pair of states x and y, the transition rate $t_{x,y}$ is replaced by transition probability $P_{x,y} = t_{x,y}/\Omega$. The holding and service rate costs are respectively $h(\cdot)/(\beta+\Omega)$ and $c(\cdot)/(\beta+\Omega)$, while the discount rate is $\Omega/(\beta+\Omega)$.

Optimal stationary policy for this discrete-time dynamic programming problem is governed by the Bellman equations [17],

$$V_{0,j} = \frac{1}{\beta+\Omega}\left\{\frac{\lambda}{\eta^{j-1}} \sum_{j'=1}^{L} \sigma^2 q^{j'-1} V_{1,j'} + \left(\Omega - \frac{\lambda}{\eta^{j-1}}\right) V_{0,j}\right\}, \quad j = 0, 1, \ldots, L-1 \tag{33}$$

$$V_{i,j} = \min_{\mu \in [0,\overline{\mu}]} \frac{1}{\beta+\Omega}\left\{h(i) + c(\mu) + \frac{\lambda}{\eta^{j-1}} \sum_{j'=1}^{L} \sigma^2 q^{j'-1} V_{i+1,j'} + \mu V_{i-1,j}\right.$$
$$\left. + \left(\Omega - \frac{\lambda}{\eta^{j-1}} - \mu\right) V_{i,j}\right\}, \quad i = 1, 2, \ldots, \quad j = 0, 1, \ldots, L-1 \tag{34}$$

where $V_{i,j}$ is the total accumulated cost if the system commences with i customers and with the interarrival scale being j [i.e., in state (i,j)]. The optimal stationary service rate $\mu^*_{i,j}$ for state (i,j) is then the minimizing rate for the corresponding equation of (34). If the minimum is achieved at multiple values of μ, the lowest rate is picked. This system of equations can be solved numerically via the value iteration method [17]. Specifically,

beginning with $V_{i,j}(0) = 0$ for all (i, j), we iterate the system

$$V_{0,j}(k+1) = \frac{1}{\beta+\Omega}\left\{\frac{\lambda}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}V_{1,j'}(k) + \left(\Omega - \frac{\lambda}{\eta^{j-1}}\right)V_{0,j}(k)\right\} \tag{35}$$

$$V_{i,j}(k+1) = \min_{\mu\in[0,\bar{\mu}]}\frac{1}{\beta+\Omega}\Big\{h(i) + c(\mu) + \frac{\lambda}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}V_{i+1,j'}(k) + \mu V_{i-1,j}(k) + \left(\Omega - \frac{\lambda}{\eta^{j-1}} - \mu\right)V_{i,j}(k)\Big\}, \quad i = 1,2,\ldots, \quad j = 0,1,\ldots,L-1 \tag{36}$$

until convergence. The optimizing rate $\mu_{i,j}^{(k)}$ at each iteration k then converges to $\mu_{i,j}^*$.

Based on this approach, optimal server control is designed for several systems with fractal renewal process input, and the resulting policy is shown in Fig. 14. To allow closed-form minimization of (36) in each step, a quadratic service rate cost $c(\mu) = c_0\mu^2$ is adopted throughout. Moreover, a linear holding cost $h(i) = h_0 i$ is used, which is a measure of expected delay in a first-in-first-out queue. In the cases depicted, $c_0 = 1$ and $h_0 = 0.01$. To reflect equal importance of the present and the future in our decision making, a discount rate $\beta = 0$ is used.

Each curve in Fig. 14 represents the optimal service rate $\mu_{i,j}^*$ as a function of the inter-arrival scale j, with the number of customers i held fixed. For a given queue length, the optimal service rate is seen to decrease monotonically for coarser (i.e., larger) scales. Thus, if the next arrival is expected to be distant, the service can be relaxed without burdening the future. The optimal service rate is also monotonic in the number of customers when the scale is fixed: the more crowded the system, the busier the server. In Appendix C, we show that this actually holds more broadly for any monotonic convex holding cost $h(i)$. Finally, we note that small γ cases are less demanding on the server. In effect, heavier tails of the corresponding interarrival densities imply that prospective customers are in general more distant, and hence, have less impact on the present decision making.

5.1.2 State Estimation and Realizable Server Control

In this section, we develop scale estimates that can be used together with queue length information in the state-based controller structure of the previous section to obtain practical control policies. In particular, we consider estimation of the interarrival scale from the time t that has elapsed since the last arrival. With a multiscale representation of the interarrival density as in (2), the minimum probability-of-error—i.e., maximum-a-posteriori (MAP)—scale estimate is the scale j that maximizes

$$\mathcal{L}(j) = p_j \Pr\{X_j > t\} = p_j \exp(-\lambda_j t).$$

Using (4), we get that

$$\frac{\mathcal{L}(j+1)}{\mathcal{L}(j)} = q \exp(-(\lambda_{j+1} - \lambda_j)t),$$

which is monotonically increasing in t. Thus, $\mathcal{L}(j+1) > \mathcal{L}(j)$ if and only if

$$t > \ln\left(\frac{1}{q}\right) \frac{1}{\lambda_j - \lambda_{j+1}} = \eta^j \ln\left(\frac{1}{q}\right) \frac{\eta}{(\eta-1)\lambda},$$

where the equality follows from (3). Thus, our scale estimate $\hat{\jmath}(t)$ is the unique integer j such that

$$\eta^{j-1} \ln\left(\frac{1}{q}\right) \frac{\eta}{(\eta-1)\lambda} < t \leq \eta^j \ln\left(\frac{1}{q}\right) \frac{\eta}{(\eta-1)\lambda},$$

i.e.,

$$\hat{\jmath}(t) = \left\lceil \log_\eta \left(\frac{t(\eta-1)\lambda}{\eta \ln(1/q)} \right) \right\rceil. \tag{37}$$

Fig. 15 depicts the optimal server policy of Fig. 14(a), modified to operate without knowledge of the interarrival scale. These plots are obtained by warping the horizontal axis to yield $\mu^*_{i,\hat{\jmath}(t)}$ as a function of number of customers present i, and the time elapsed since the last customer arrival t. To apply the prescription of Fig. 15, we follow a particular solid curve as time progresses, until a customer enters or leaves the system. For a customer

arrival, we move to the beginning of the next higher curve. Upon a customer departure, we drop vertically to the next lower curve. Thus, past history of the system, specifically the arrival epoch of the last customer, is crucial in our decision making.

5.1.3 Simulations

Through simulations, we examine the benefits of our multiscale controller over more conventional designs which ignore past history, such as the M/M/1 controller and its variations. In contrast to our systematic multiscale controller design, we will see that delicate hand-tuning is generally required for these memoryless controllers to achieve even reasonable performance when servicing fractal input.

Our simulations are based on the next-event time advance method of [12]. For fair comparison, each controller is given the same customer input, which is a sample function of a fractal renewal process with shape parameter $\gamma = 1.8$, consisting of $N = 500\,000$ arrivals. The cost parameters are set at $c_0 = 10$, $h_0 = 0.01$ and $\beta = 0$ throughout. To derive the M/M/1 controller of [17], the input is treated as if it were a Poisson stream, with arrival rate estimated according to $N/S_X[N]$, where $S_X[N]$ denotes the arrival epoch of the last customer. The resulting service policies are shown as the dashed plots in Fig. 16, together with the multiscale policies which are represented by the solid curves.

Based on the simulated systems, we have obtained the total cost J (defined in (30)) and the average delay under a first-in-first-out discipline, and have tabulated these quantities in Table 1. From the table, it is apparent that the M/M/1 controller is substantially inferior in terms of either measure. As we can see in Fig. 16, this controller attempts to lower overall cost by reducing the service cost, at the expense of a higher holding cost. Consequently, it is more susceptible to congestion. The estimated steady-state customer distributions in Fig. 17 provide further evidence for a longer queue expected for the M/M/1 controller.

One way to lower the holding cost is by raising service rates. This can be achieved simply by scaling up the service rates of the optimal M/M/1 controller. By trial and error, we have tuned the M/M/1 controller to yield the least overall cost for the given customer input. The resulting controller is depicted in Fig. 18, alongside the optimal multiscale controller, and its performance is summarized in the third column of Table 1. Although it

is the best controller obtainable by scaling the M/M/1 controller, this design is nevertheless still suboptimal, as can be seen from Table 1. Also, as is apparent from Fig. 18, the modified M/M/1 controller is more demanding on resources, requiring a substantially higher peak service rate than the multiscale controller.

That a large proportion of interarrivals in a fractal renewal process occur on short time scales suggests that an alternative controller design could, in principle, be based on mimicking fine-scale behavior of the optimal multiscale controller. We next design a memoryless controller of this type based on the fine-scale service rates in the optimal multiscale policy. As before, we hand-tune these service rates to obtain the lowest total cost for the given customer input, using trial and error. The resulting controller is plotted in Fig. 19, while its performance is summarized in the last column of Table 1. As is apparent from the table, this controller is still inferior to the multiscale scheme; note, for example, the considerably higher average delay.

The preceding results collectively reflect that while memoryless service policies are simple to implement, in general they yield rather poor results for fractal customer input. Even when carefully hand-tuned according to the actual customer arrivals, these policies are inherently inferior to the multiscale controller. Because the fractal model is a process with strong memory, intelligent allocation of service based on its history is ultimately key to successful control of this class of queues.

5.2 Flow Control for Power-Law Services

In this section, we study optimal control of the queueing system of Section 4.4, where Poisson customers are serviced with power-law holding times. The basis of our design will be the transposed multiscale birth-and-death process of Fig. 11, which is used for queueing analysis in Section 4.4. With this model, it is the service that is described in the form of a multiscale representation.

5.2.1 Multiscale Flow Control

The heavy-tailed customer distribution inherent in such queueing systems means that traditional flow control is problematic. In terms of system management, such controllers result

in high likelihood of buffer overflow, while from the customers' perspective, long delay and poor quality of service are experienced. In this section, we describe improved flow control policies that mitigate these problems. To achieve this, we allow the controller to vary the admission rate λ between 0 and the actual arrival rate $\overline{\lambda}$. As before, we first solve this problem assuming complete knowledge of the state of the system. Thus, we denote the input rate at state (i,j) by $\lambda_{i,j}$.

The objective function is again made up of two components, with a holding cost $h(i)$ which reflects buffer occupancy, and a throttling cost $c(\lambda)$ which penalizes loss of customers. We assume that the holding cost is monotonically nondecreasing in the number of customers, while the throttling cost is monotonically nonincreasing in the admission rate. In addition, we assume $h(0) = c(\overline{\lambda}) = 0$. The total cost is accumulated over time, with a discount rate β included:

$$J = E\left[\int_0^\infty e^{-\beta t}\left[h(i(t)) + c(\lambda(t))\right] dt\right], \tag{38}$$

where $i(t)$ is the number of customers in the queueing system at time t, and $\lambda(t)$ is the admission rate at time t.

Our approach to this problem will be very similar to the server design of Section 5.1. Applying rate uniformization to the transposed multiscale birth-and-death process of Fig. 11, we obtain the equivalent process of Fig. 20. Note that here we have also lumped the zeroth superstate into a single state to obtain a more realistic model; the scale of the next service in general cannot be deduced from an empty queue. We next recast this problem into its discrete-time equivalent, with the corresponding Bellman equations

$$V_0 = \min_{\lambda\in[0,\overline{\lambda}]} \frac{1}{\beta+\Omega}\left\{c(\lambda) + \lambda\sum_{j'=1}^{L}\sigma^2 q^{j'-1}V_{1,j'} + (\Omega-\lambda)\,V_0\right\} \tag{39}$$

$$\begin{aligned} V_{i,j} = \min_{\lambda\in[0,\overline{\lambda}]} \frac{1}{\beta+\Omega}\Bigg\{&h(i) + c(\lambda) + \frac{\mu}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}V_{i-1,j'} + \lambda V_{i+1,j} \\ &+ \left(\Omega - \frac{\mu}{\eta^{j-1}} - \lambda\right)V_{i,j}\Bigg\}, \quad i = 1,2,\ldots, \quad j = 0,1,\ldots,L-1, \end{aligned} \tag{40}$$

where $V_{i,j}$ is the total accumulated cost if the system commences in state (i,j). The optimal

stationary admission rate $\lambda^*_{i,j}$ when the system is in state (i,j) is the minimizing rate for the corresponding equation of (39) and (40). If the minimum is achieved at multiple values, we pick the highest rate. In contrast to the optimal server control problem of Section 5.1, the zeroth-order equation (39) is no longer trivial, since the throttling decision in the zeroth superstate does have an influence on the future.

Figs. 21 shows the optimal stationary flow control policy obtained by solving (39) and (40) via the value iteration method [17]. Throughout, a quadratic throttling cost and linear holding cost are used, i.e., $c(\lambda) = c_0(\overline{\lambda} - \lambda)^2$ and $h(i) = h_0 i$, where $c_0 = 1$ and $h_0 = 0.01$. Also, the discount rate is set at $\beta = 0$.

For a fixed number of customers i, the optimal admission rate $\lambda^*_{i,j}$ is seen to decrease for coarser scale service duration, i.e., larger j. Thus, if the current job is expected to require long service, fewer customers should be admitted to prevent high costs for the future. On the other hand, for a fixed scale j, the optimal admission rate is monotonic in the number of customers. Hence, higher degree of throttling is required for a busier system. In Appendix D, we prove that this monotonicity actually holds more generally for any convex holding cost $h(i)$. Comparing Figs. 21(a) and 21(b), we also see that lower γ cases require higher throttling. Due to the longer expected service time, fewer customers can be accommodated in these cases. Finally, as in the server control problem, practicable flow control can be efficiently realized from the idealized policy by integrating minimum probability-of-error (MAP) state estimates:

$$\hat{\jmath}(t) = \left\lceil \log_\eta \left(\frac{t(\eta-1)\mu}{\eta \ln(1/q)} \right) \right\rceil, \tag{41}$$

where t now represents the time elapsed since the last service completion.

6 Conclusion

In this paper, we have applied multiscale techniques in the study of fractal point processes in various realistic networking scenarios. Our analysis of these processes under random erasure and superposition is consistent with the broadly-observed self-similarity in aggregate traffic; further analysis with this point process model may lead to further insights into mechansims

by which self-similarity arises in networks. Our queueing analysis resulted in methods for computing various performance measures such as quality of service and resource consumption. This analysis identified several problems with the use of traditional queueing design techniques for these scenarios—in particular, the substantial underutilization of resources with memoryless service of fractal point process, and the inherently poor quality of service for Poisson process serviced with power-law holding times. To mitigate these effects, we have applied multiscale paradigms to develop systematic design methodologies for the operation of these queueing systems. Several practical optimal controllers were obtained from this approach. Through simulations, we have shown that by exploiting system history, a simple multiscale controller significantly out-performs simplistic controllers which ignore such past information. While we have focused on several specific problems mainly for illustrative purposes, our design algorithms are readily generalized to other queueing scenarios, such as those involving different cost structures.

A Derivation of the Counting Process Distribution Coefficients: Arrival-Observed Case

In this section, we derive $\left\{\pi_i^{(\mathrm{a})}(t); i = 0, 1, \dots\right\}$, the arrival-observed counting process distribution. First, as given in (9), a closed-form expression exists for the 0th-order term $\pi_0^{(\mathrm{a})}(t)$. To obtain Taylor's series coefficients for higher-order terms, we expand the z-transform of the probability distribution (8) as follows

$$\hat{\mathbf{p}}(z;t)\mathbf{1}^{\mathrm{T}} = \mathbf{q}\left(\mathbf{I} + \left(-\mathbf{B} + z\mathbf{b}^{\mathrm{T}}\mathbf{q}\right)(\lambda t) + \frac{\left(-\mathbf{B} + z\mathbf{b}^{\mathrm{T}}\mathbf{q}\right)^2(\lambda t)^2}{2!} + \cdots\right)\mathbf{1}^{\mathrm{T}}.$$

Extracting the coefficient of z^1, we get that

$$\pi_1^{(\mathrm{a})}(t) = a_1^{(1)}(\lambda t) - a_2^{(1)}\frac{(\lambda t)^2}{2!} + \cdots \tag{42}$$

where for $k \geq 1$,

$$a_k^{(1)} = \mathbf{q}\mathbf{B}^{k-1}\mathbf{b}^{\mathrm{T}}\mathbf{q}\mathbf{1}^{\mathrm{T}} + \mathbf{q}\mathbf{B}^{k-2}\mathbf{b}^{\mathrm{T}}\mathbf{q}\mathbf{B}\mathbf{1}^{\mathrm{T}} + \cdots + \mathbf{q}\mathbf{B}\mathbf{b}^{\mathrm{T}}\mathbf{q}\mathbf{B}^{k-2}\mathbf{1}^{\mathrm{T}} + \mathbf{q}\mathbf{b}^{\mathrm{T}}\mathbf{q}\mathbf{B}^{k-1}\mathbf{1}^{\mathrm{T}}. \tag{43}$$

But $\mathbf{b} = \mathbf{1B}$. Thus,

$$a_k^{(1)} = \mathbf{qB}^k\mathbf{1}^{\mathrm{T}}\mathbf{q1}^{\mathrm{T}} + \mathbf{qB}^{k-1}\mathbf{1}^{\mathrm{T}}\mathbf{qB1}^{\mathrm{T}} + \cdots + \mathbf{qB}^2\mathbf{1}^{\mathrm{T}}\mathbf{qB}^{k-2}\mathbf{1}^{\mathrm{T}} + \mathbf{qB1}^{\mathrm{T}}\mathbf{qB}^{k-1}\mathbf{1}^{\mathrm{T}}. \quad (44)$$

This is simply the sum of all terms of the form

$$\mathbf{qB}^m\mathbf{1}^{\mathrm{T}}\mathbf{qB}^n\mathbf{1}^{\mathrm{T}}$$

such that $m \geq 1$, $n \geq 0$, and $m + n = k$. Moreover, we recognize $\mathbf{qB}^l\mathbf{1}^{\mathrm{T}}$ as the lth moment of the random variable R, with distribution given in (12). Thus, we have

$$a_k^{(1)} = \sum_{l=1}^{k} M_l M_{k-l}, \quad (45)$$

with M_l denoting the lth moment of R.

Next we turn to the expansion of $\pi_i^{(\mathrm{a})}(t)$ for arbitrary i. First, it is clear that the coefficients of t^k are zero for $k < i$. Thus, we have

$$\pi_i^{(\mathrm{a})}(t) = a_i^{(i)}\frac{(\lambda t)^i}{i!} - a_{i+1}^{(i)}\frac{(\lambda t)^{i+1}}{(i+1)!} + \cdots. \quad (46)$$

Using the same argument as before, we have that each coefficient $a_k^{(i)}$ is the sum of all terms of the form

$$\mathbf{qB}^{m_1}\mathbf{1}^{\mathrm{T}}\mathbf{qB}^{m_2}\mathbf{1}^{\mathrm{T}}\cdots\mathbf{qB}^{m_i}\mathbf{1}^{\mathrm{T}}\mathbf{qB}^{m_{i+1}}\mathbf{1}^{\mathrm{T}}, \quad (47)$$

with $m_1, m_2, \ldots, m_i \geq 1$, $m_{i+1} \geq 0$, and $m_1 + m_2 + \cdots + m_{i+1} = k$. Such terms with $m_1 = l$, where $1 \leq l \leq k-i+1$, are those with $m_2, m_3, \ldots, m_i \geq 1$, $m_{i+1} \geq 0$, $m_2 + m_3 + \cdots + m_{i+1} = k - l$. But these are exactly the terms making up the sum $a_{k-l}^{(i-1)}$. Thus, we have

$$a_k^{(i)} = \sum_{l=1}^{k-i+1} M_l a_{k-l}^{(i-1)}. \quad (48)$$

B Derivation of the Counting Process Distribution Coefficients: Random Incidence Case

We first argue that the left null space of the matrix $\left(-\mathbf{B} + \mathbf{b}^{\mathrm{T}}\mathbf{q}\right)$ is spanned by the vector $\mathbf{qB}^{-1}$. Suppose the vector $\mathbf{v}$ is in the left null space, or

$$\mathbf{v}\left(-\mathbf{B} + \mathbf{b}^{\mathrm{T}}\mathbf{q}\right) = \mathbf{0}$$

where $\mathbf{0}$ is a zero row vector. So,

$$\mathbf{vB} = \mathbf{vb}^{\mathrm{T}}\mathbf{q} = \kappa\mathbf{q}$$

where κ is the inner product $\mathbf{v}\mathbf{b}^{\mathrm{T}}$. Right-multiplying both sides by $\mathbf{B}^{-1}$, then, we have

$$\mathbf{v} = \kappa\left(\mathbf{q}\mathbf{B}^{-1}\right).$$

We proceed to determine the coefficients in the counting process probability distribution $\left\{\pi_i^{(\mathrm{r})}(t); i = 0, 1, \ldots\right\}$. Again, as given in (13), we have a closed-form expression for the 0th-order term $\pi_0^{(\mathrm{r})}(t)$. For higher-order terms, we expand the z-transform of (8), with the initial condition assuming the value $\hat{\mathbf{p}}(z;0) = \tilde{\sigma}^2\mathbf{q}\mathbf{B}^{-1}$. Hence,

$$\hat{\mathbf{p}}(z;t)\mathbf{1}^{\mathrm{T}} = \tilde{\sigma}^2\mathbf{q}\mathbf{B}^{-1}\left(\mathbf{I} + \left(-\mathbf{B} + z\mathbf{b}^{\mathrm{T}}\mathbf{q}\right)(\lambda t) + \frac{(-\mathbf{B} + z\mathbf{b}^{\mathrm{T}}\mathbf{q})^2(\lambda t)^2}{2!} + \cdots\right)\mathbf{1}^{\mathrm{T}}. \tag{49}$$

As before, it is clear that the coefficients for t^k will be zero for $k < i$. Thus, we have

$$\pi_i^{(\mathrm{r})}(t) = r_i^{(i)}\frac{(\lambda t)^i}{i!} - r_{i+1}^{(i)}\frac{(\lambda t)^{i+1}}{(i+1)!} + \cdots. \tag{50}$$

Focusing on the kth power of t in the coefficient of z^i, we see that this is the sum of all terms of the form

$$\tilde{\sigma}^2\mathbf{q}\mathbf{B}^{-1}\mathbf{B}^{m_1}\mathbf{1}^{\mathrm{T}}\mathbf{q}\mathbf{B}^{m_2}\mathbf{1}^{\mathrm{T}}\cdots\mathbf{q}\mathbf{B}^{m_i}\mathbf{1}^{\mathrm{T}}\mathbf{q}\mathbf{B}^{m_{i+1}}\mathbf{1}^{\mathrm{T}}, \tag{51}$$

with $m_1, m_2, \ldots, m_i \geq 1$, $m_{i+1} \geq 0$, and $m_1 + m_2 + \cdots + m_{i+1} = k$. For those terms with $m_1 = 1$, the product reduces to

$$\tilde{\sigma}^2\mathbf{q}\mathbf{B}^{m_2}\mathbf{1}^{\mathrm{T}}\cdots\mathbf{q}\mathbf{B}^{m_i}\mathbf{1}^{\mathrm{T}}\mathbf{q}\mathbf{B}^{m_{i+1}}\mathbf{1}^{\mathrm{T}}, \tag{52}$$

since $\mathbf{q}\mathbf{1}^{\mathrm{T}} = 1$. But the sum of all terms of this form is just $\tilde{\sigma}^2 a_k^{(i-1)}$. On the other hand, terms with $m_1 > 1$ can be written as

$$\tilde{\sigma}^2\mathbf{q}\mathbf{B}^{\tilde{m}_1}\mathbf{1}^{\mathrm{T}}\mathbf{q}\mathbf{B}^{m_2}\mathbf{1}^{\mathrm{T}}\cdots\mathbf{q}\mathbf{B}^{m_i}\mathbf{1}^{\mathrm{T}}\mathbf{q}\mathbf{B}^{m_{i+1}}\mathbf{1}^{\mathrm{T}}, \tag{53}$$

with $\tilde{m}_1, m_2, \ldots, m_i \geq 1$, $m_{i+1} \geq 0$, and $\tilde{m}_1 + m_2 + \cdots + m_{i+1} = k - 1$. But this is precisely $\tilde{\sigma}^2 a_{k-1}^{(i)}$. Thus, we have shown the relation

$$r_k^{(i)} = (a_{k-1}^{(i)} + a_k^{(i-1)})\tilde{\sigma}^2. \tag{54}$$

C Optimal Fractal Queue Server Monotonicity

In this appendix, we show that the optimal service rate $\mu_{i,j}^*$ for a fractal renewal process input satisfies

$$\mu_{i-1,j}^* \leq \mu_{i,j}^*, \quad \text{for } i = 1, 2, \ldots, \tag{55}$$

if the holding cost function $h(i)$ is convex, i.e.,

$$h(i) - h(i-1) \leq h(i+1) - h(i), \quad \text{for } i = 1, 2, \ldots.$$

We first show that (55) holds if the first difference of $V_{i,j}$, defined as

$$\Delta_{i,j} \triangleq V_{i,j} - V_{i-1,j},$$

is nondecreasing in i. Rewriting the Bellman equations (34) for $i > 0$, we get that

$$\begin{aligned} V_{i,j} = \frac{1}{\beta + \Omega} \Bigg\{ h(i) + \frac{\lambda}{\eta^{j-1}} \sum_{j'=1}^{L} \sigma^2 q^{j'-1} V_{i+1,j'} + \left(\Omega - \frac{\lambda}{\eta^{j-1}} \right) V_{i,j} \\ + \min_{\mu \in [0, \overline{\mu}]} \Big[c(\mu) - \mu \Delta_{i,j} \Big] \Bigg\}, \quad i = 1, 2, \ldots, \quad j = 0, 1, \ldots, L-1. \end{aligned} \tag{56}$$

Now, suppose that $\Delta_{i,j}$ is indeed nondecreasing in i, and let i_1, i_2 be some nonnegative integers, with $i_2 > i_1$. Then, for every $\mu < \mu^*_{i_1,j}$,

$$\begin{aligned} c(\mu^*_{i_1,j}) - \mu^*_{i_1,j} \Delta_{i_2,j} &= c(\mu^*_{i_1,j}) - \mu^*_{i_1,j} \Delta_{i_1,j} - \mu^*_{i_1,j} (\Delta_{i_2,j} - \Delta_{i_1,j}) \\ &< c(\mu) - \mu \Delta_{i_1,j} - \mu^*_{i_1,j} (\Delta_{i_2,j} - \Delta_{i_1,j}) \\ &< c(\mu) - \mu \Delta_{i_1,j} - \mu (\Delta_{i_2,j} - \Delta_{i_1,j}) = c(\mu) - \mu \Delta_{i_2,j}, \end{aligned}$$

where the first inequality follows from the definition of $\mu^*_{i_1,j}$, and the second inequality follows from monotonicity of $\Delta_{i,j}$. So, $\mu^*_{i_1,j} \leq \mu^*_{i_2,j}$.

We proceed to show that $\Delta_{i,j}$ is indeed monotonic in i. For this, it suffices to show that at each stage k of the value iteration method, the first difference

$$\Delta_{i,j}(k) \triangleq V_{i,j}(k) - V_{i-1,j}(k) \tag{57}$$

has this property. We set the boundary value $\Delta_{0,j}(k)$ to 0, for all j, k.

Now, it is trivial that $\Delta_{i,j}(0)$ is nondecreasing in i. Assuming this holds for $\Delta_{i,j}(k)$, we consider the next iteration. Specifically, we rewrite the value iteration equation (36) as

$$\begin{aligned} V_{i,j}(k+1) = \frac{1}{\beta + \Omega} \Bigg\{ h(i) + \frac{\lambda}{\eta^{j-1}} \sum_{j'=1}^{L} \sigma^2 q^{j'-1} V_{i+1,j'}(k) + \left(\Omega - \frac{\lambda}{\eta^{j-1}} \right) V_{i,j}(k) \\ + \min_{\mu \in [0, \overline{\mu}]} \Big[c(\mu) - \mu \Delta_{i,j}(k) \Big] \Bigg\}, \quad i = 1, 2, \ldots. \end{aligned} \tag{58}$$

Defining the optimal service rate at each stage k of the value iteration method as $\mu^{(k)}_{i,j}$, i.e.,

$$\mu^{(k)}_{i,j} \triangleq \arg \min_{\mu \in [0, \overline{\mu}]} \Big[c(\mu) - \mu \Delta_{i,j}(k) \Big], \tag{59}$$

for every i, j, k, we have that

$$\begin{aligned}
\Delta_{i+1,j}(k+1) &= V_{i+1,j}(k+1) - V_{i,j}(k+1) \\
&\geq \frac{1}{\beta+\Omega}\Bigg\{h(i+1) + \frac{\lambda}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}V_{i+2,j'}(k) \\
&\qquad + \left(\Omega - \frac{\lambda}{\eta^{j-1}}\right)V_{i+1,j}(k) + c(\mu_{i+1,j}^{(k)}) - \mu_{i+1,j}^{(k)}\Delta_{i+1,j}(k) \\
&\qquad - h(i) - \frac{\lambda}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}V_{i+1,j'}(k) \\
&\qquad - \left(\Omega - \frac{\lambda}{\eta^{j-1}}\right)V_{i,j}(k) - c(\mu_{i+1,j}^{(k)}) + \mu_{i+1,j}^{(k)}\Delta_{i,j}(k)\Bigg\} \\
&= \frac{1}{\beta+\Omega}\Bigg\{h(i+1) - h(i) + \frac{\lambda}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}\Delta_{i+2,j'}(k) \\
&\qquad + \left(\Omega - \frac{\lambda}{\eta^{j-1}}\right)\Delta_{i+1,j}(k) - \mu_{i+1,j}^{(k)}\left(\Delta_{i+1,j}(k) - \Delta_{i,j}(k)\right)\Bigg\}.
\end{aligned} \tag{60}$$

Similarly, we get that

$$\begin{aligned}
\Delta_{i,j}(k+1) \leq \frac{1}{\beta+\Omega}\Bigg\{&h(i) - h(i-1) + \frac{\lambda}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}\Delta_{i+1,j'}(k) \\
&+ \left(\Omega - \frac{\lambda}{\eta^{j-1}}\right)\Delta_{i,j}(k) - \mu_{i-1,j}^{(k)}\left(\Delta_{i,j}(k) - \Delta_{i-1,j}(k)\right)\Bigg\}.
\end{aligned} \tag{61}$$

Subtracting these two inequalities, we get that

$$\begin{aligned}
(\beta+\Omega)(\Delta_{i+1,j}(k+1) - \Delta_{i,j}(k+1)) \geq \Bigg(&(h(i+1) - h(i)) - (h(i) - h(i-1)) \\
&+ \frac{\lambda}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}[\Delta_{i+2,j'}(k) - \Delta_{i+1,j'}(k)] \\
&+ \left(\Omega - \frac{\lambda}{\eta^{j-1}} - \mu_{i+1,j}^{(k)}\right)[\Delta_{i+1,j}(k) - \Delta_{i,j}(k)] \\
&+ \mu_{i-1,j}^{(k)}[\Delta_{i,j}(k) - \Delta_{i-1,j}(k)]\Bigg).
\end{aligned} \tag{62}$$

By the induction hypothesis, convexity of $h(i)$, and the fact that $\Omega \geq \lambda/\eta^{j-1} - \mu_{i+1,j}^{(k)}$, we get that $\Delta_{i+1,j}(k+1) \geq \Delta_{i,j}(k+1)$. Our assertion therefore follows by mathematical induction.

D Optimal Flow Control Policy Monotonicity

In this appendix, we show that the optimal admission rate $\lambda_{i,j}^*$ for a power-law queueing system servicing a Poisson input satisfies

$$\lambda_{i-1,j}^* \geq \lambda_{i,j}^*, \qquad \text{for } i = 1, 2, \ldots, \tag{63}$$

given that the holding cost function $h(i)$ is convex, i.e.,

$$h(i) - h(i-1) \leq h(i+1) - h(i), \qquad \text{for } i = 1, 2, \ldots.$$

Using exactly the same argument as in Appendix C, we can show that $\lambda^*_{i,j}$ is nonincreasing in i, if the first difference of $V_{i,j}$ is nondecreasing in i. We proceed to show that the first difference $\Delta_{i,j}$, defined as in Appendix C, is monotonically nondecreasing. Our approach will exploit the value iteration method, and prove by mathematical induction that the first difference of $V_{i,j}(k)$ at each stage of the iteration, is monotonic in i. Again, $\Delta_0(k)$ is set to be zero for all k. Also, it is clear that $\Delta_{i,j}(0)$ is nondecreasing in i, since $V_{i,j}(0)$ is identically zero. We assume the assertion holds for k, and analyze

$$\Delta_{i,j}(k+1) - \Delta_{i-1,j}(k+1).$$

For each k, we also define $V_{0,j}(k) \triangleq V_0(k)$, and $\Delta_{0,j}(k) \triangleq \Delta_0(k)$.

To this end, we write the value iteration equations as

$$V_0(k+1) = \frac{1}{\beta+\Omega}\left\{\Omega V_0(k) + \min_{\lambda\in[0,\overline{\lambda}]}\left[c(\lambda) + \lambda\sum_{j'=1}^{L}\sigma^2 q^{j'-1}\left(V_{1,j'}(k) - V_0(k)\right)\right]\right\} \tag{64}$$

$$\begin{aligned} V_{i,j}(k+1) = \frac{1}{\beta+\Omega}\Bigg\{h(i) + \frac{\mu}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}V_{i-1,j'}(k) + \left(\Omega - \frac{\mu}{\eta^{j-1}}\right)V_{i,j}(k) \\ + \min_{\lambda\in[0,\overline{\lambda}]}\left[c(\lambda) + \lambda\Delta_{i+1,j}(k)\right]\Bigg\}, \quad i = 1,2,\ldots, \quad j = 0,1,\ldots,L-1. \end{aligned} \tag{65}$$

We define the optimal service rate at each stage k of the value iteration method as $\lambda^{(k)}_{i,j}$, i.e.,

$$\lambda^{(k)}_{i,j} \triangleq \arg\min_{\lambda\in[0,\overline{\lambda}]}\left[c(\lambda) + \lambda\Delta_{i+1,j}(k)\right]. \tag{66}$$

Now, for $i > 1$, we get that

$$\begin{aligned} V_{i+1,j}(k+1) = \frac{1}{\beta+\Omega}\Bigg\{h(i+1) + \frac{\mu}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}V_{i,j'}(k) + \left(\Omega - \frac{\mu}{\eta^{j-1}}\right)V_{i+1,j}(k) \\ + c(\lambda^{(k)}_{i+1,j}) + \lambda^{(k)}_{i+1,j}\Delta_{i+2,j}(k)\Bigg\}, \end{aligned}$$

and

$$\begin{aligned} V_{i,j}(k+1) = \frac{1}{\beta+\Omega}\Bigg\{h(i) + \frac{\mu}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}V_{i-1,j'}(k) + \left(\Omega - \frac{\mu}{\eta^{j-1}}\right)V_{i,j}(k) \\ + c(\lambda^{(k)}_{i,j}) + \lambda^{(k)}_{i,j}\Delta_{i+1,j}(k)\Bigg\}. \end{aligned}$$

Subtracting, we get that

$$\begin{aligned}\Delta_{i+1,j}(k+1) &= V_{i+1,j}(k+1) - V_{i,j}(k+1) \\ &\geq \frac{1}{\beta+\Omega}\Bigg\{h(i+1) - h(i) + \frac{\mu}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}\Delta_{i,j'}(k) \\ &\qquad + \left(\Omega - \frac{\mu}{\eta^{j-1}}\right)\Delta_{i+1,j}(k) + \lambda_{i+1,j}^{(k)}\left(\Delta_{i+2,j}(k) - \Delta_{i+1,j}(k)\right)\Bigg\}.\end{aligned}$$

Similarly,

$$\begin{aligned}\Delta_{i,j}(k+1) &= V_{i,j}(k+1) - V_{i-1,j}(k+1) \\ &\leq \frac{1}{\beta+\Omega}\Bigg\{h(i) - h(i-1) + \frac{\mu}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}\Delta_{i-1,j'}(k) + \left(\Omega - \frac{\mu}{\eta^{j-1}}\right)\Delta_{i,j}(k) \\ &\qquad + \lambda_{i-1,j}^{(k)}\left(\Delta_{i+1,j}(k) - \Delta_{i,j}(k)\right)\Bigg\}.\end{aligned}$$

Thus,

$$\begin{aligned}(\beta+\Omega)(\Delta_{i+1,j}(k+1) - \Delta_{i,j}(k+1)) \geq \Bigg(&(h(i+1)-h(i)) - (h(i)-h(i-1)) \\ &+ \frac{\mu}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}[\Delta_{i,j'}(k) - \Delta_{i-1,j'}(k)] \\ &+ \left(\Omega - \frac{\mu}{\eta^{j-1}} - \lambda_{i-1,j}^{(k)}\right)[\Delta_{i+1,j}(k) - \Delta_{i,j}(k)] \\ &+ \lambda_{i+1,j}^{(k)}[\Delta_{i+2,j}(k) - \Delta_{i+1,j}(k)]\Bigg).\end{aligned} \tag{67}$$

Thus, it follows from the induction hypothesis, convexity of $h(i)$, and the fact that $\Omega \geq \frac{\mu}{\eta^{j-1}} + \lambda_{i-1,j}^{(k)}$, that $\Delta_{i,j}(k+1)$ is nondecreasing for $i \geq 2$. To complete the proof, we now show that $\Delta_{2,j}(k+1) \geq \Delta_{1,j}(k+1)$. We note that

$$\begin{aligned}\Delta_{2,j}(k+1) &= V_{2,j}(k+1) - V_{1,j}(k+1) \\ &\geq \frac{1}{\beta+\Omega}\Bigg\{h(2) - h(1) + \frac{\mu}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}\Delta_{1,j'}(k) + \left(\Omega - \frac{\mu}{\eta^{j-1}}\right)\Delta_{2,j}(k) \\ &\qquad + \lambda_{2,j}^{(k)}\left(\Delta_{3,j}(k) - \Delta_{2,j}(k)\right)\Bigg\},\end{aligned}$$

and

$$\begin{aligned}\Delta_{1,j}(k+1) &= V_{1,j}(k+1) - V_{0,j}(k+1) \\ &\leq \frac{1}{\beta+\Omega}\Bigg\{h(1) + \frac{\mu}{\eta^{j-1}}\sum_{j'=1}^{L}\sigma^2 q^{j'-1}\Delta_{0,j'}(k) + \left(\Omega - \frac{\mu}{\eta^{j-1}}\right)\Delta_{1,j}(k) \\ &\qquad + \lambda_{0,j}^{(k)}\left(\Delta_{2,j}(k) - \Delta_{1,j}(k)\right)\Bigg\}.\end{aligned}$$

Thus, again, our assertion follows for the case $i = 1$, for the same reasons.

References

[1] J. Beran, R. Sherman, M. S. Taqqu, and W. Willinger, "Long-range dependence in variable-bit-rate video traffic," *IEEE Trans. Commun.*, vol. 43, no. 2/3/4, pp. 1566–1579, 1995.

[2] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the self-similar nature of Ethernet traffic (extended version)," *IEEE/ACM Trans. Networking*, vol. 2, no. 1, pp. 1–15, 1994.

[3] B. Mandelbrot, "Self-similar error clusters in communication systems and the concept of conditional stationarity," *IEEE Trans. Comm. Tech.*, vol. 13, pp. 71–90, Mar. 1965.

[4] V. Paxson and S. Floyd, "Wide area traffic: The failure of Poisson modeling," *IEEE/ACM Trans. Networking*, vol. 3, no. 3, pp. 226–244, 1995.

[5] D. H. Johnson and A. R. Kumar, "Modeling and analyzing fractal point processes," in *Proc. Int. Conf. Acoust. Speech, Signal Processing*, 1990.

[6] S. B. Lowen and M. C. Teich, "Fractal renewal processes generate $1/f$ noise," *Physical Review E*, vol. 47, pp. 992–1001, Feb. 1993.

[7] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson, "Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level," *Computer communication review*, vol. 25, no. 4, pp. 100–113, 1995.

[8] W. M. Lam and G. W. Wornell, "Multiscale representation and estimation of fractal point processes," *IEEE Trans. Signal Processing*, vol. 43, no. 11, pp. 2606–2617, 1995.

[9] L. Kleinrock, *Queueing Systems*, vol. I. New York: John Wiley & Sons, 1975.

[10] S. M. Sussman, "Analysis of the Pareto model for error statistics on telephone circuits," *IEEE Trans. Comm. Sys.*, vol. 11, pp. 213–221, Jun. 1963.

[11] M. Neuts, *Matrix-Geometric Solutions in Stochastic Models.* Baltimore, Maryland: Johns Hopkins University Press, 1981.

[12] A. M. Law and W. D. Kelton, *Simulation modeling and analysis.* New York, NY: McGraw-Hill, 1991.

[13] R. G. Addie and R. E. Warfield, "Bandwidth switching and new network architectures," *Teletraffic Sci.*, vol. ITC-12, pp. 665–671, 1989.

[14] J. Y. Hui, "Resource allocation for broadband networks," *IEEE Trans. Commun.*, vol. 6, no. 9, pp. 1598–1608, 1988.

[15] R. A. Howard, *Dynamic programming and Markov processes.* Cambridge, Ma: Technology Press of Massachusetts Institute of Technology, 1960.

[16] R. E. Bellman, *Dynamic programming and modern control theory.* New York, NY: Academic Press, 1965.

[17] D. P. Bertsekas, *Dynamic Programming and Optimal Control.* Belmont, Massachusetts: Athena Scientific, 1995.

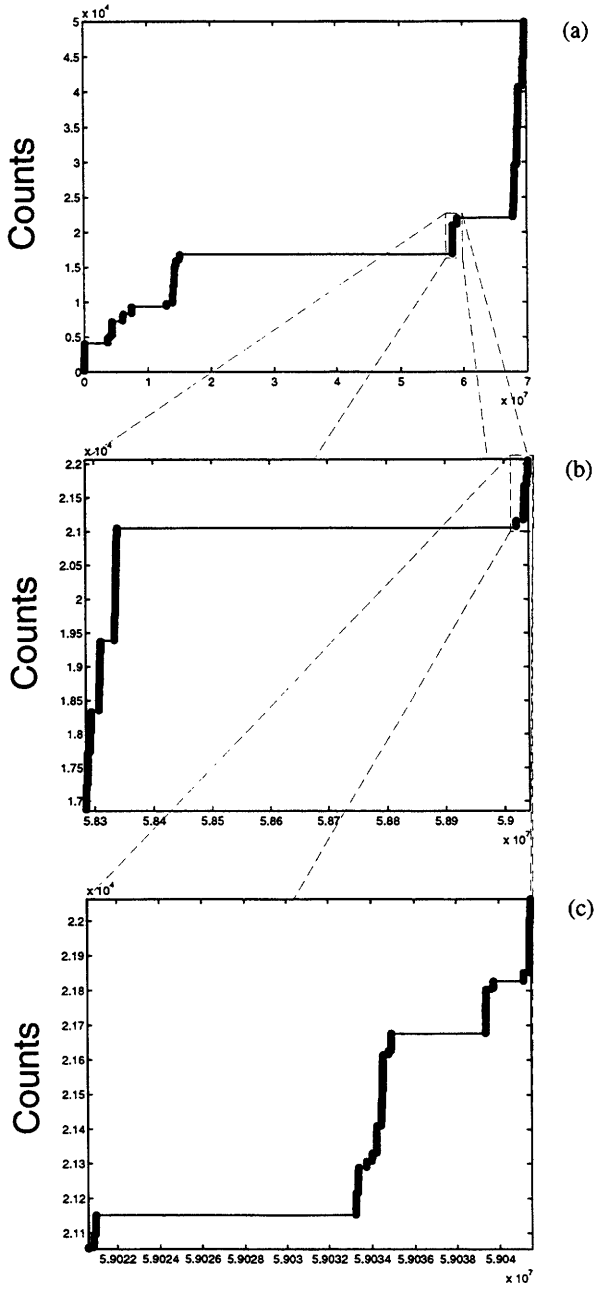


Figure 1: *Successive magnification of the counting process associated with a fractal renewal process; (a) the original process; (b) zoomed version of (a); (c) zoomed version of (b). Interarrivals were synthesized according to the power-law density function (1), with shape parameter* $\gamma = 1.5$

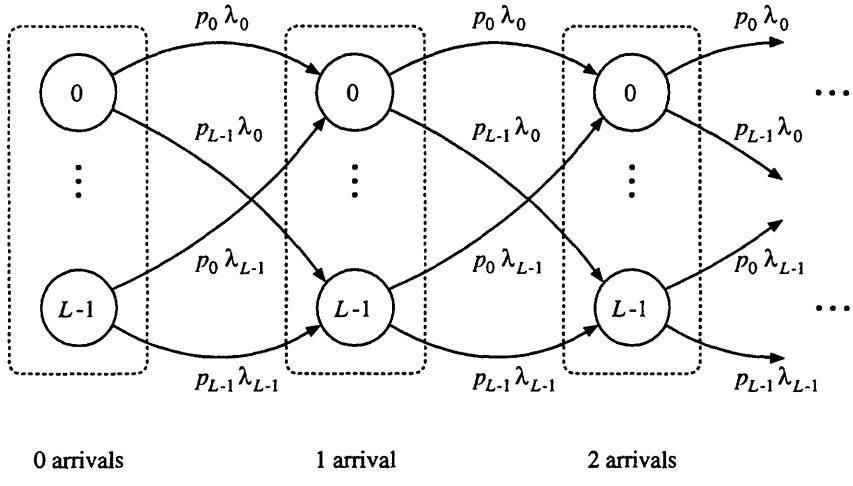


Figure 2: *The multiscale pure-birth process based on a finite-scale representation; dashed boxes denote conceptual grouping into superstates. The rate of departure from each state is a function the scale, $\lambda_j = \lambda/\eta^j$; the probability of entering scale j' of succeeding superstate is $p_{j'} = \sigma^2 q^{j'}$.*

Table 1: *Performance of various queueing server controllers servicing fractal renewal process input*

	Multiscale Controller	M/M/1 Controller	Modified M/M/1 Controller	Modified Fine-Scale Controller
Overall Cost	0.9963e6	1.161e6	1.009e6	1.017e6
Average Delay	44.4	115.0	46.8	49.0

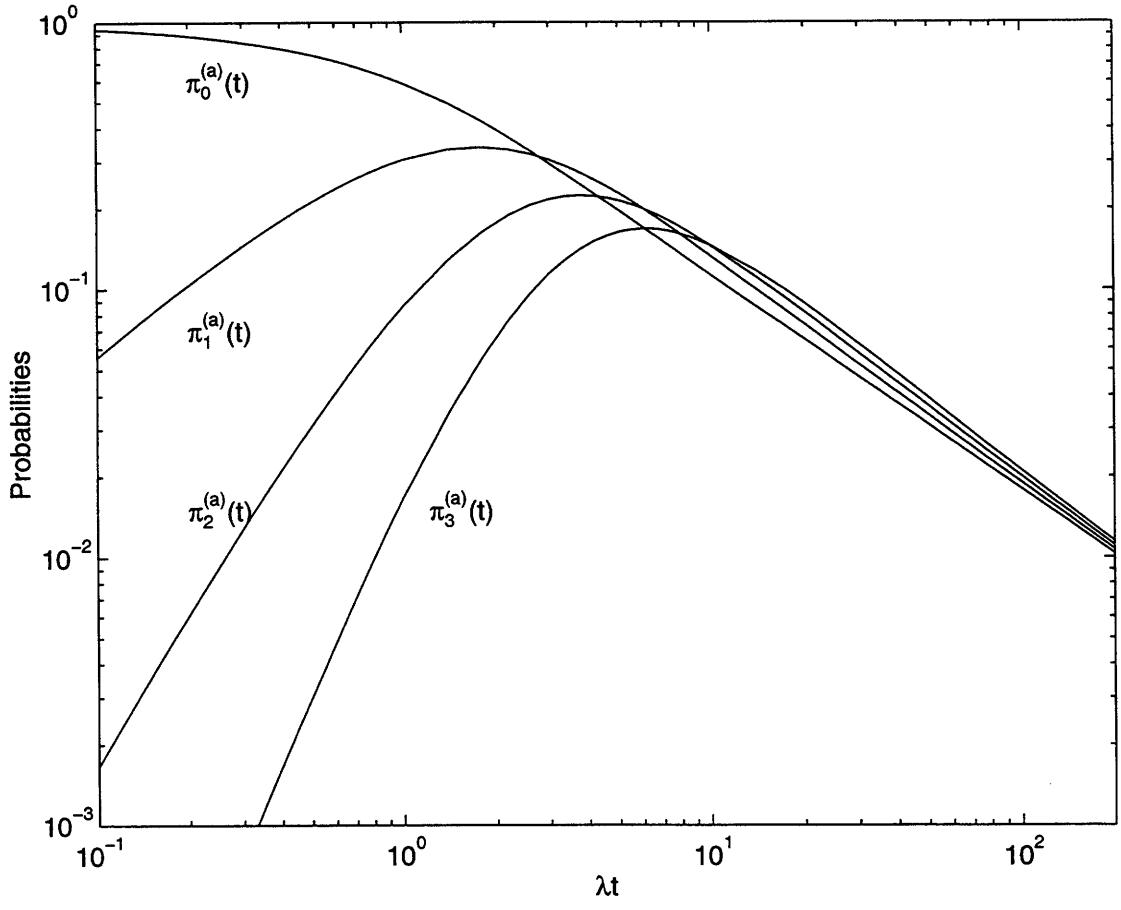


Figure 3: *First 4 terms in the arrival-observed counting process probability distribution for a fractal renewal process, computed with a 20-scale dyadic representation; the shape parameter of the interarrival distribution is* $\gamma = 1.8$. *The time axis is normalized with respect to the finest-scale arrival rate* λ.

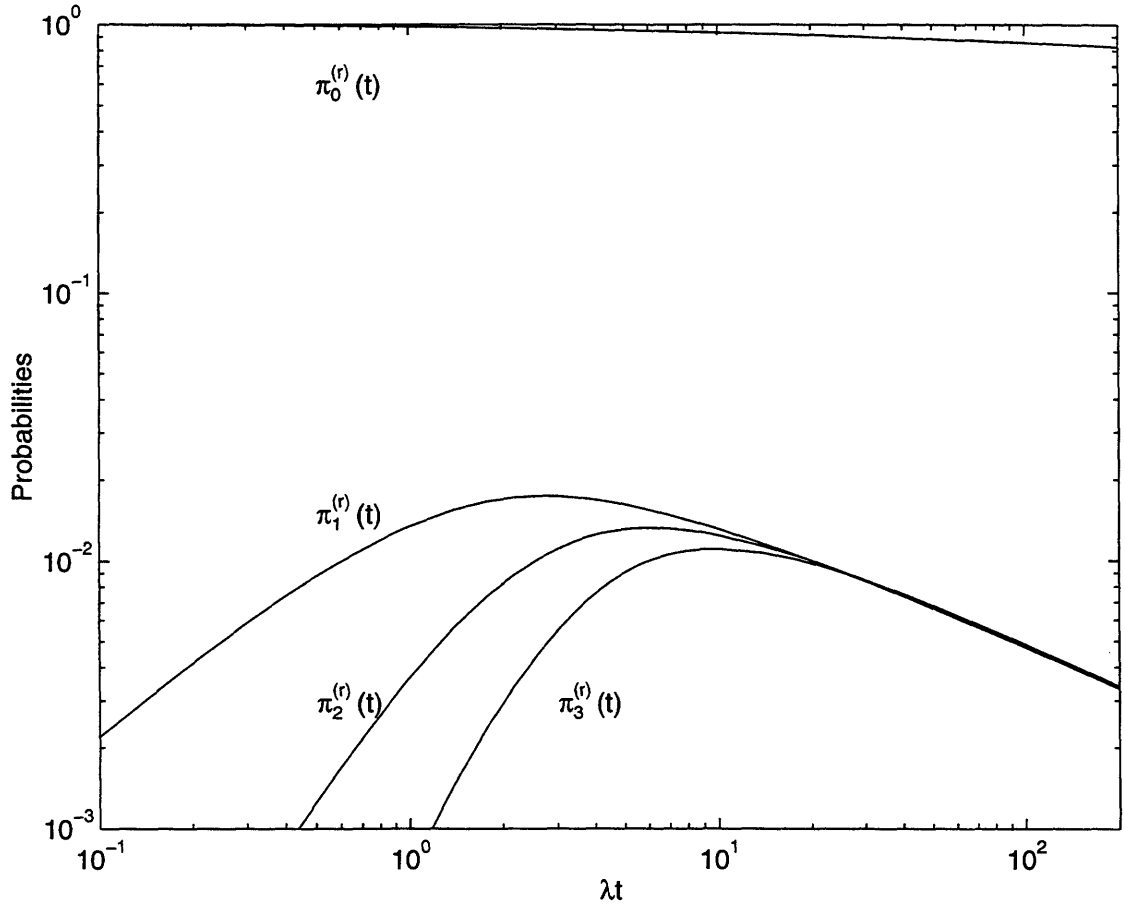


Figure 4: *First 4 terms in the random incidence counting process probability distribution for a fractal renewal process, computed with a 20-scale dyadic representation; the shape parameter of the interarrival distribution is* $\gamma = 1.8$. *The time axis is normalized with respect to the finest-scale arrival rate* λ.

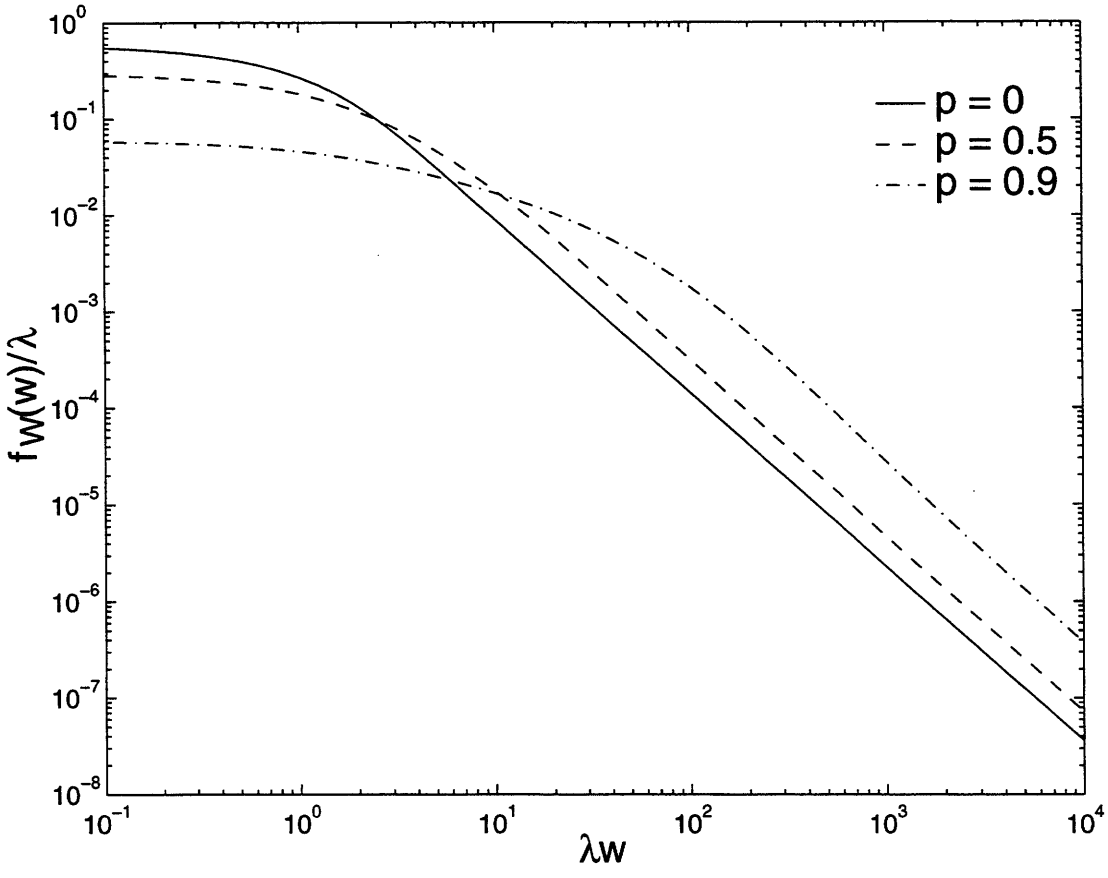


Figure 5: *Interarrival density function of a fractal renewal process under Bernoulli erasure, with erasure probability p. The shape parameter of the original process is $\gamma = 1.8$. These computations were performed with a 20-scale dyadic representation. The plots are normalized with respect to the finest-scale arrival rate λ.*

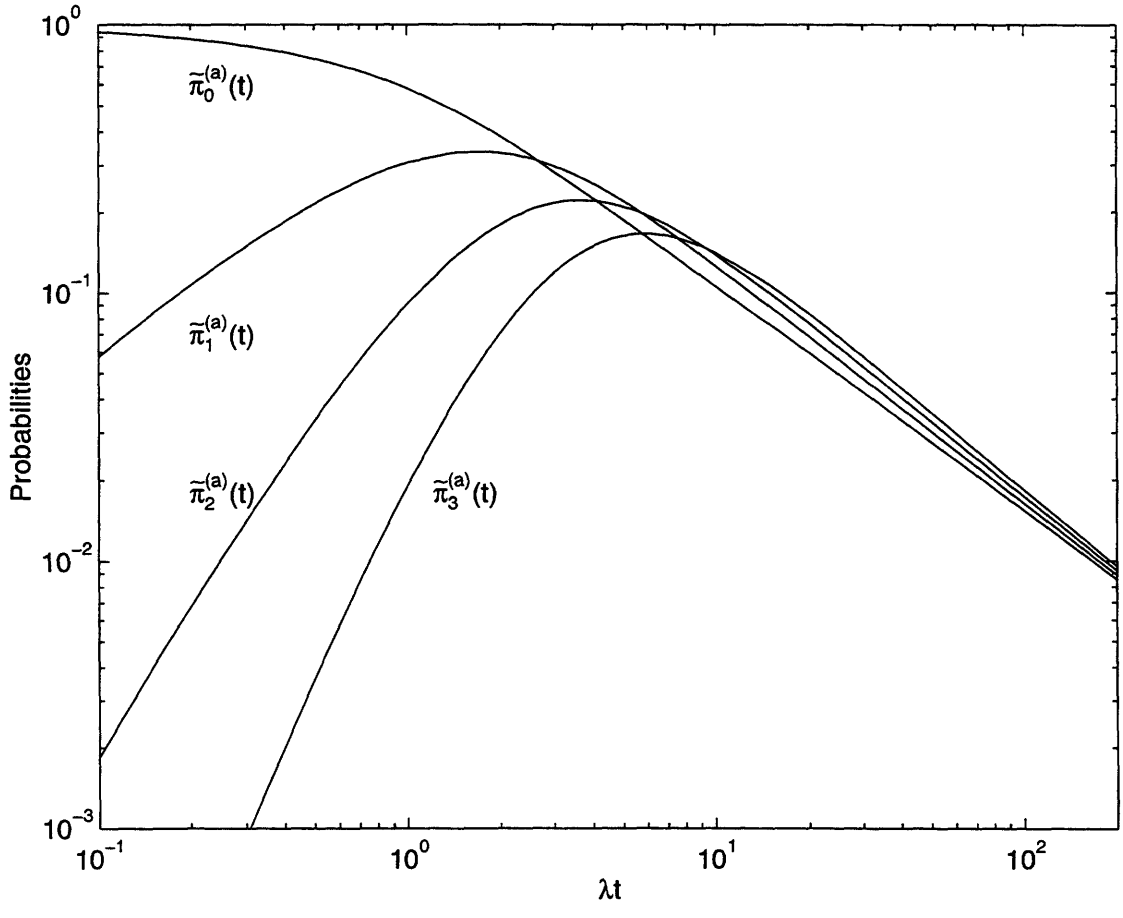


Figure 6: *Arrival-observed counting process distribution of the superposition of two independent fractal renewal processes, generated with the counting process results of Figs. 3 and 4. The shape parameter of both constituents is* $\gamma = 1.8$. *The time axis is normalized with respect to the finest-scale arrival rate* λ.

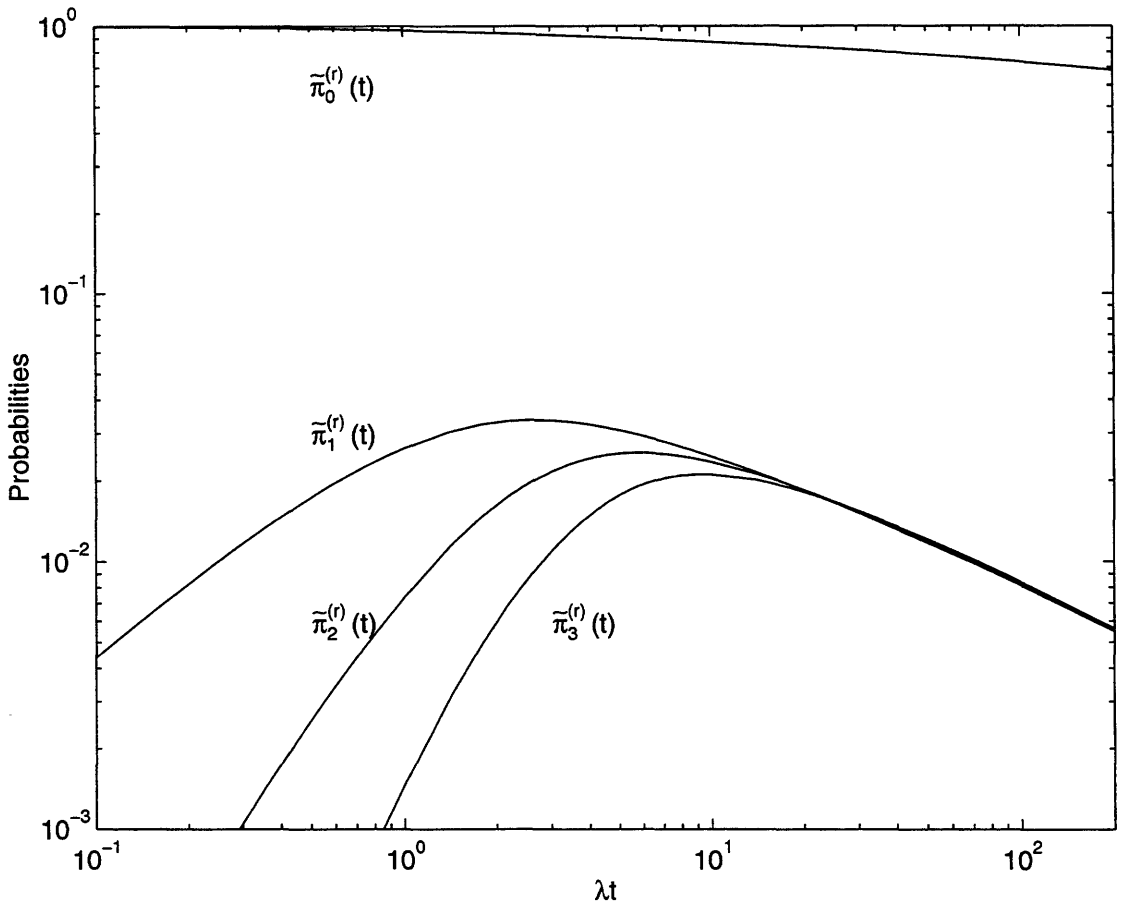


Figure 7: *Random incidence counting process distribution of the superposition of two independent fractal renewal processes, generated with the counting process results of Fig. 4. The shape parameter of both constituents is* $\gamma = 1.8$. *The time axis is normalized with respect to the finest-scale arrival rate* λ.

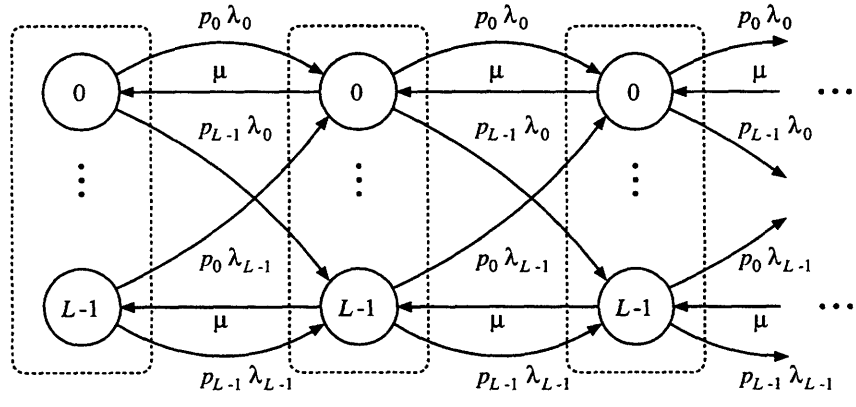


Figure 8: *The multiscale birth-and-death process based on a finite-scale representation. This is obtained by adding death transitions to the multiscale pure-birth process of Fig. 2. To model a single-server memoryless queueing system, all death transitions occur at equal rate μ.*

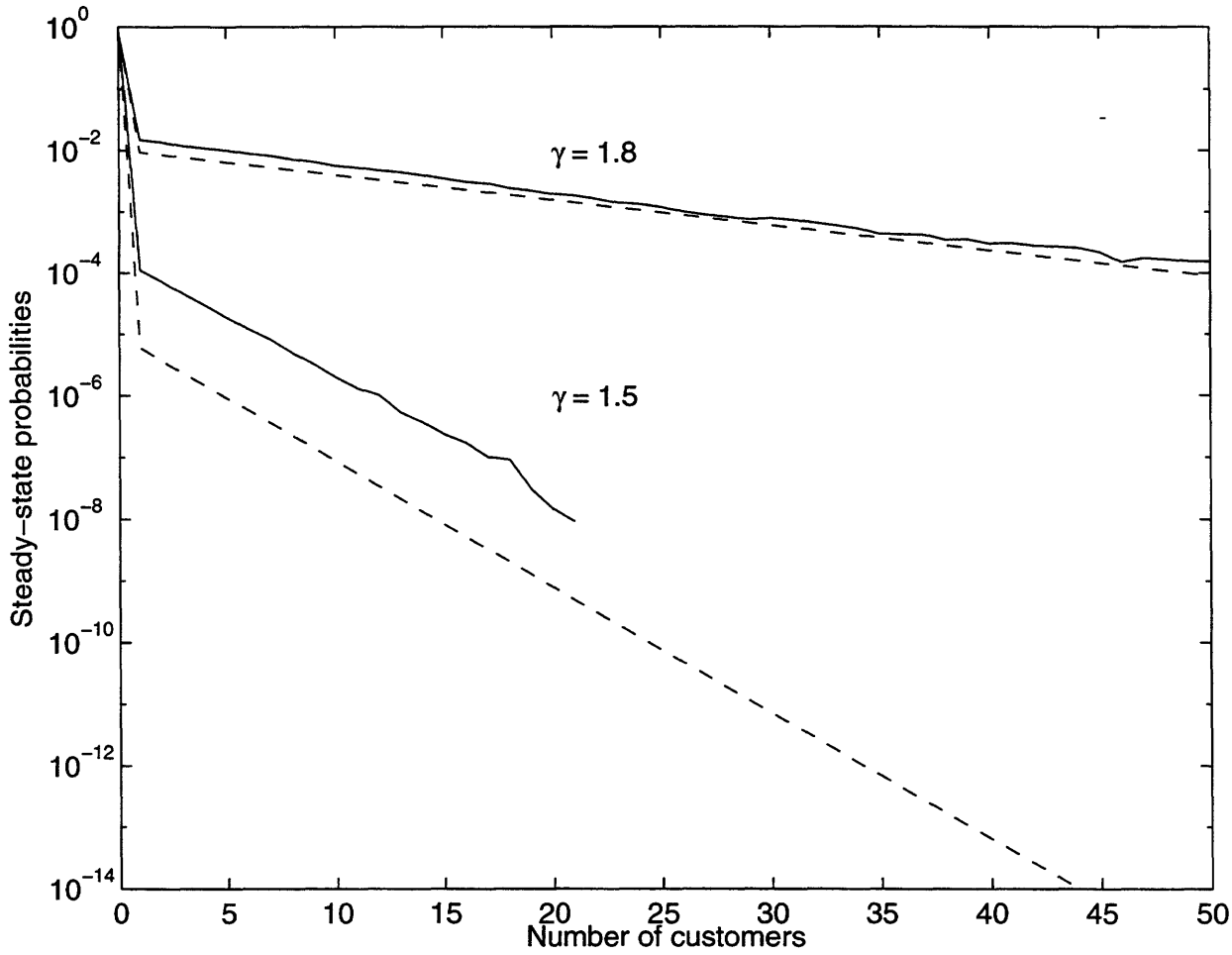


Figure 9: *First 21 terms in the steady-state customer distribution corresponding to a memoryless single-server queueing system with fractal customer arrivals. The input process is modeled with a 20-scale dyadic representation, with* $\gamma = 1.8$, *and* $1/\rho = \mu/\lambda = 0.8$. *The solid curves represent simulation results obtained with the next-event time advance simulation [12], while the dashed curves represent computed results.*

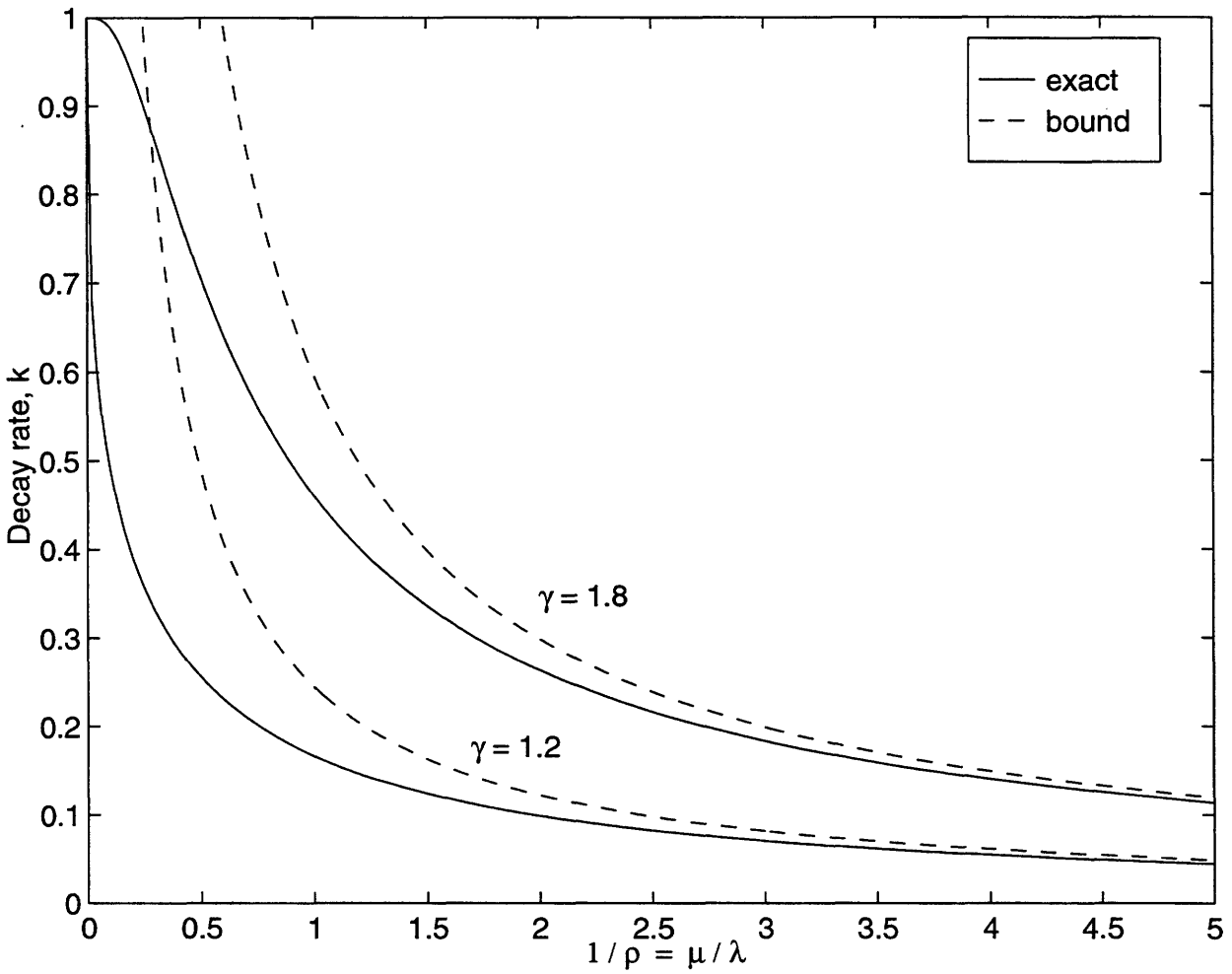


Figure 10: *Decay rate of the steady-state customer distribution in a single-server queueing system with fractal renewal process input. The solid curves are obtained via bisection-search solution to (26). The dashed curves are the closed-form upper bound prescribed by (28). A 20-scale dyadic representation is used in this computation.*

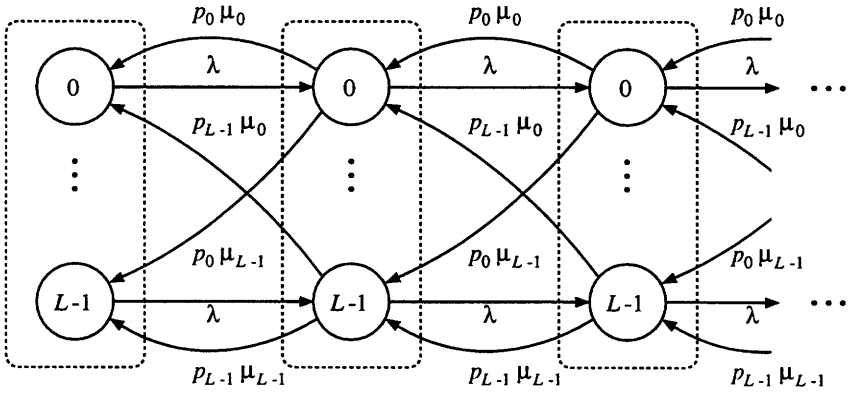


Figure 11: *The transposed multiscale birth-and-death process for memoryless input serviced by power-law server. This process is obtained by reversal of the roles of births and deaths in the process of Fig. 8.*

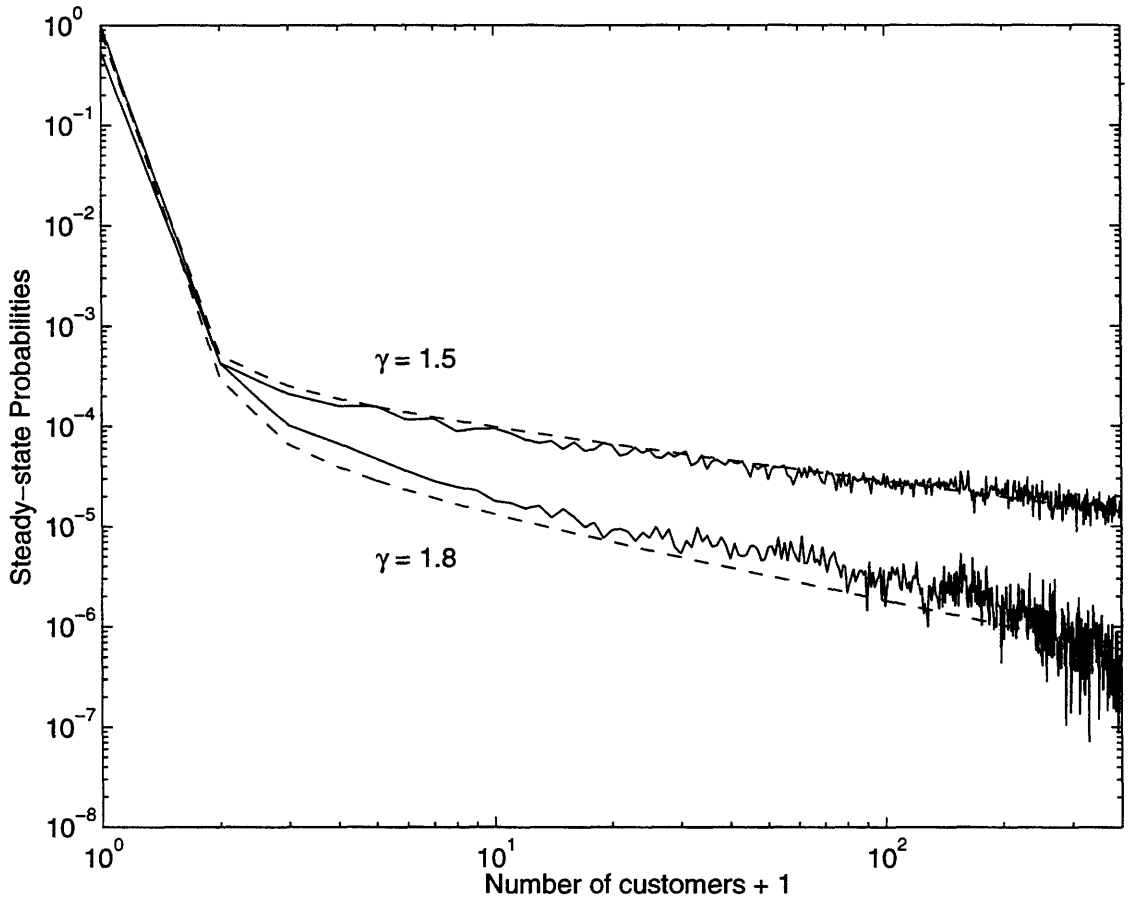


Figure 12: *Comparison of simulated and theoretical steady-state customer distribution for a queueing system with power-law holding time servicing memoryless input. The solid curves represent results of discrete-event simulation of the queueing system. The dashed curves represent theoretically-predicted distributions for the corresponding queueing scenarios. For the case* $\gamma = 1.5$, $\rho = \lambda/\mu = 2 \times 10^{-7}$, *while for the case* $\gamma = 1.8$, $\rho = \lambda/\mu = 1 \times 10^{-5}$.

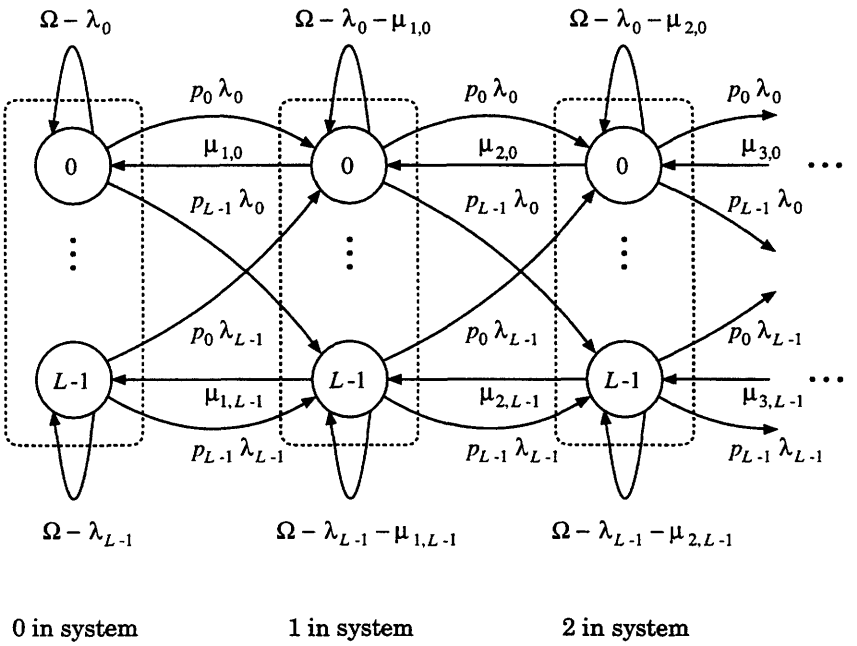
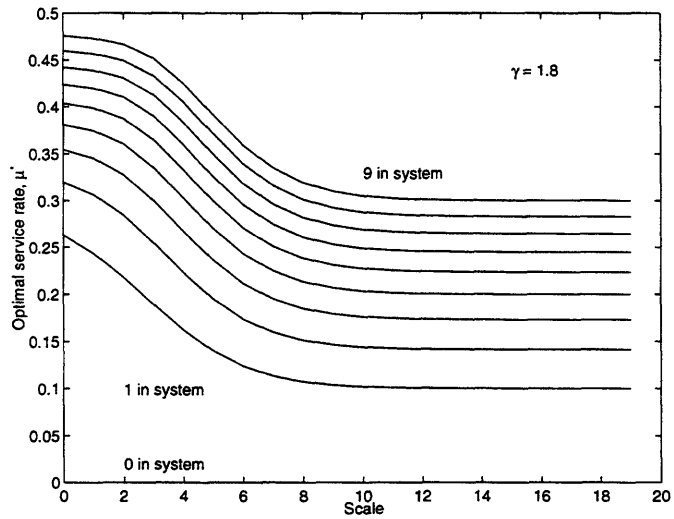
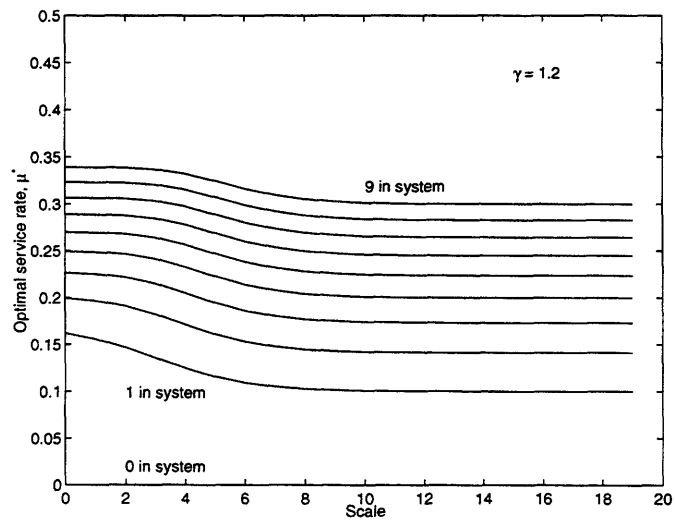


Figure 13: *The continuous-time Markov process employed in server controller design. This process is obtained from the multiscale birth-and-death process of Fig. 8 by adding self-transitions such that the total rate leaving each state is Ω. Note that system dynamics are preserved.*



(a)



(b)

Figure 14: *State-based multiscale service policy for a queueing system with fractal customer arrivals. The shape parameter of the input is* $\gamma = 1.8$ *in (a) and* $\gamma = 1.2$ *in (b). The holding cost* $h(i) = 0.01i$*, service rate cost* $c(\mu) = \mu^2$*, and discount rate* $\beta = 0$ *are used. Note that with no customer present, optimal service rate is identically 0.*

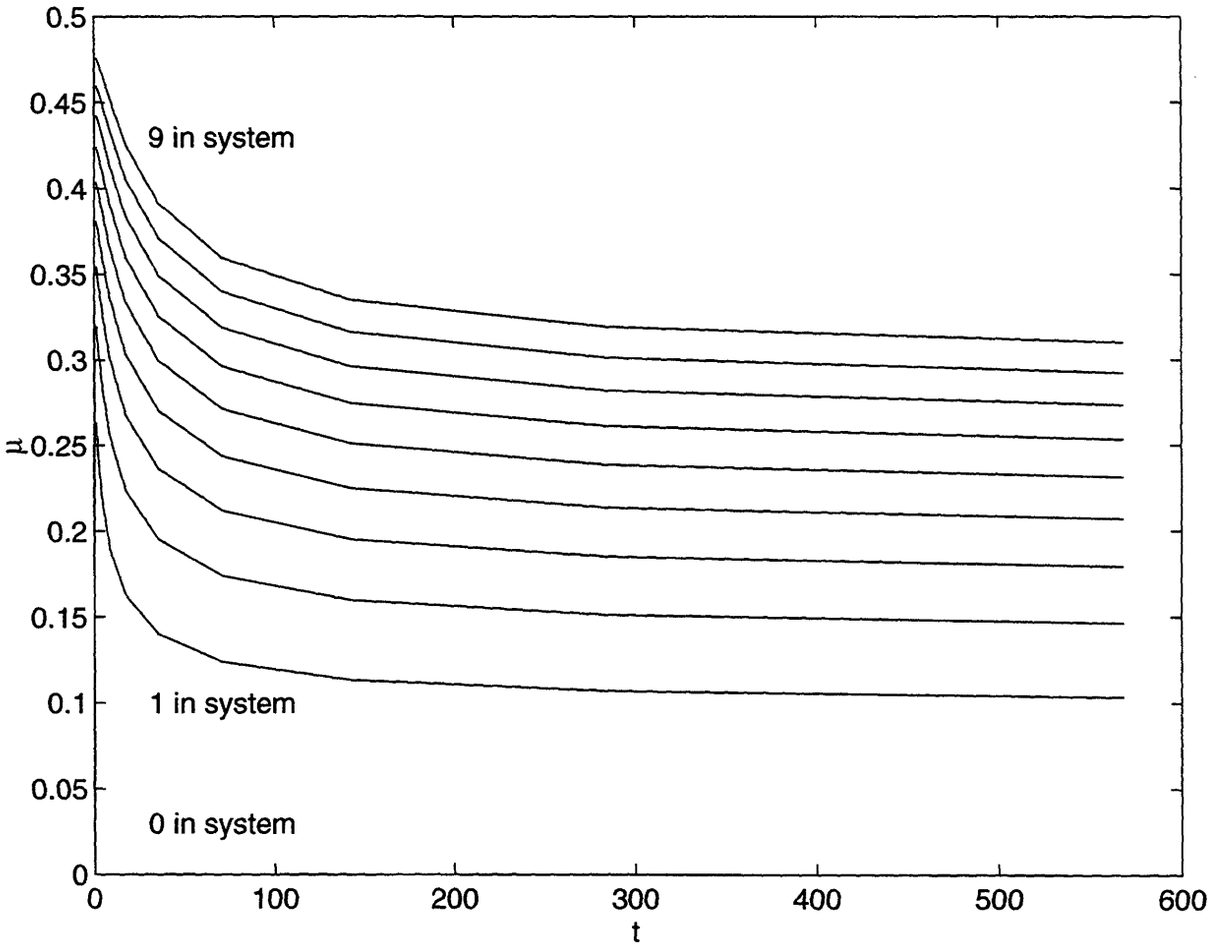


Figure 15: *Realizable optimal server for fractal renewal process input, based on the number of customers in the system, and the time elapsed since the last arrival. The depicted policy is for the case* $\gamma = 1.8, c(\mu) = \mu^2, h(i) = 0.01i$, *and is obtained via a rewarping of the horizontal axis in the graph in Fig. 14(a).*

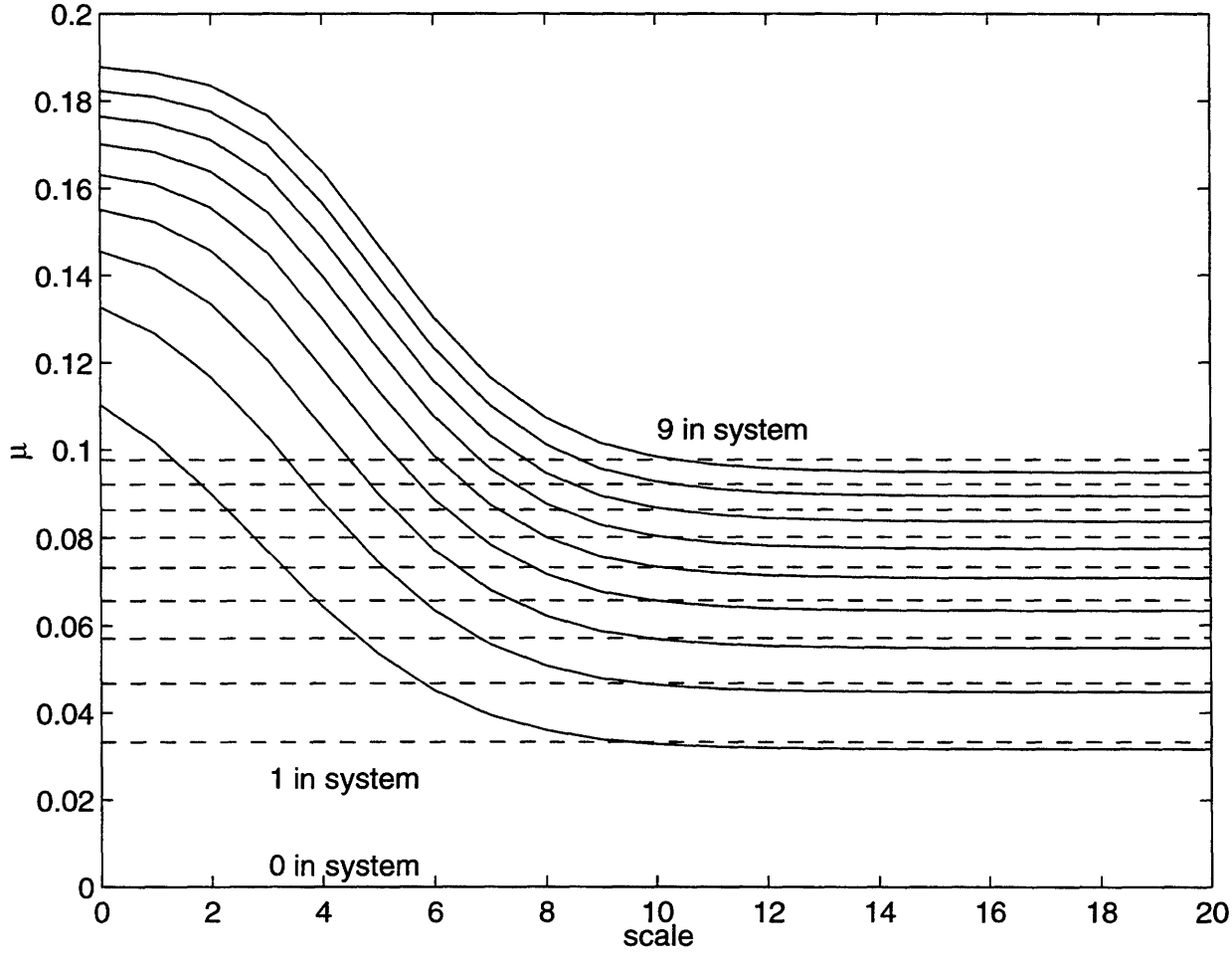


Figure 16: *Stationary server control policies for queueing systems servicing fractal renewal process input, for the case* $\gamma = 1.8$, $c(\mu) = 10\mu^2$, $h(i) = 0.01i$, $\beta = 0.0$. *The solid curves represent the optimal multiscale server. The dashed curves denote an M/M/1 service controller designed with the input treated as a Poisson process.*

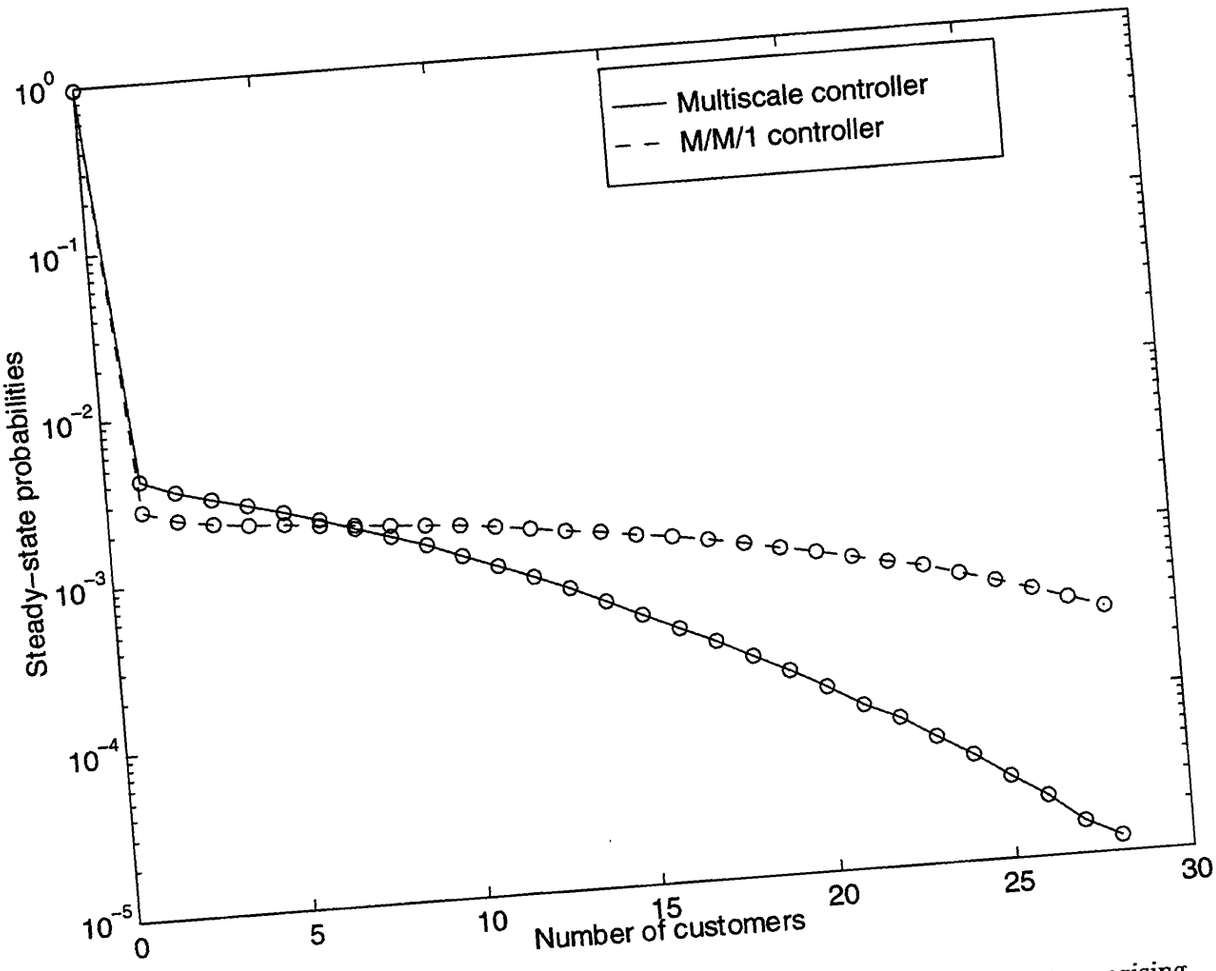


Figure 17: *Estimated steady-state customer distribution for a queueing system arising from the optimal multiscale controller and the M/M/1 controller, for the case* $\gamma = 1.8$, $c(\mu) = 10\mu^2$, $h(i) = 0.01i$, $\beta = 0.0$. *These estimates were formed using 500 000 arrivals.*

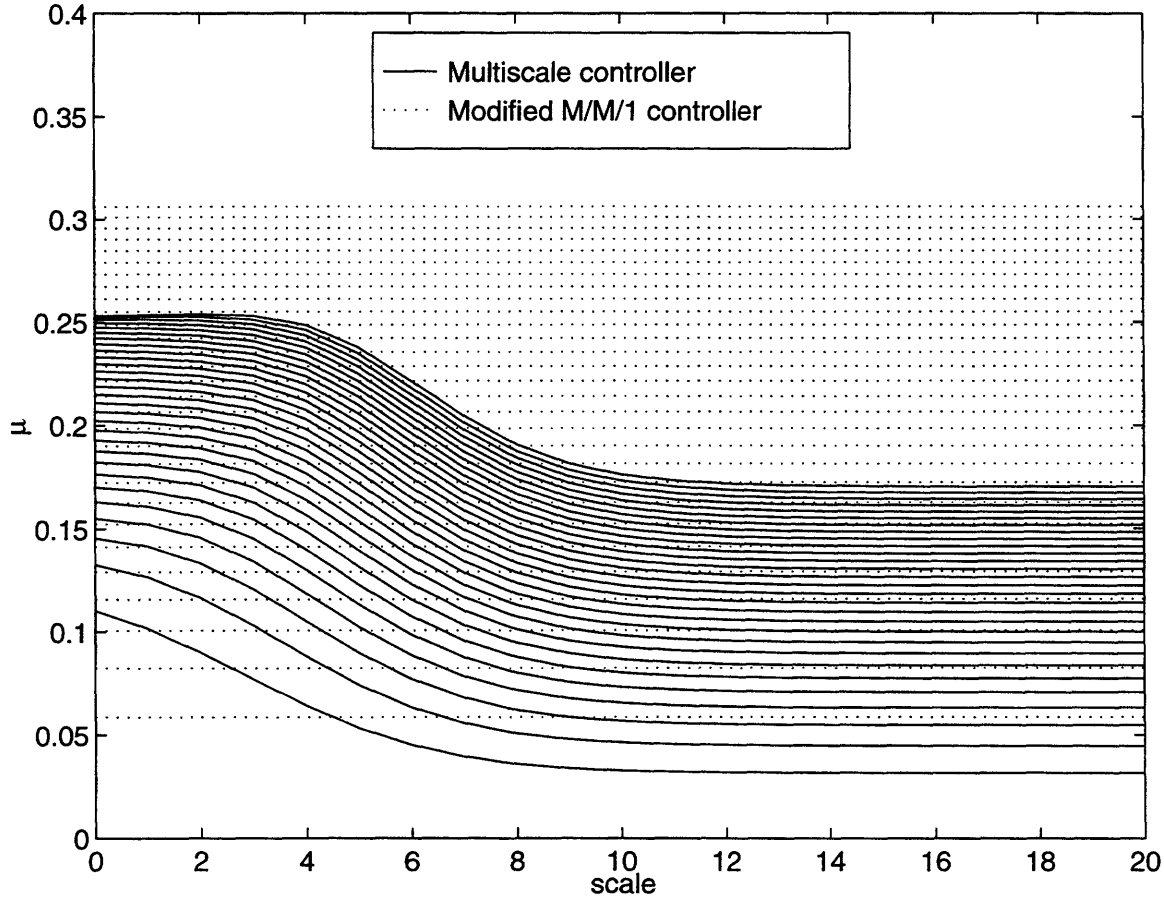


Figure 18: *Stationary server control policies for queueing systems servicing fractal renewal process input, for the case* $\gamma = 1.8$, $c(\mu) = 10\mu^2$, $h(i) = 0.01i$, $\beta = 0.0$. *The solid curves represent the optimal multiscale server. The dotted curves denote the policy for the optimized variation of the M/M/1 queueing controller obtained by scaling the service rates.*

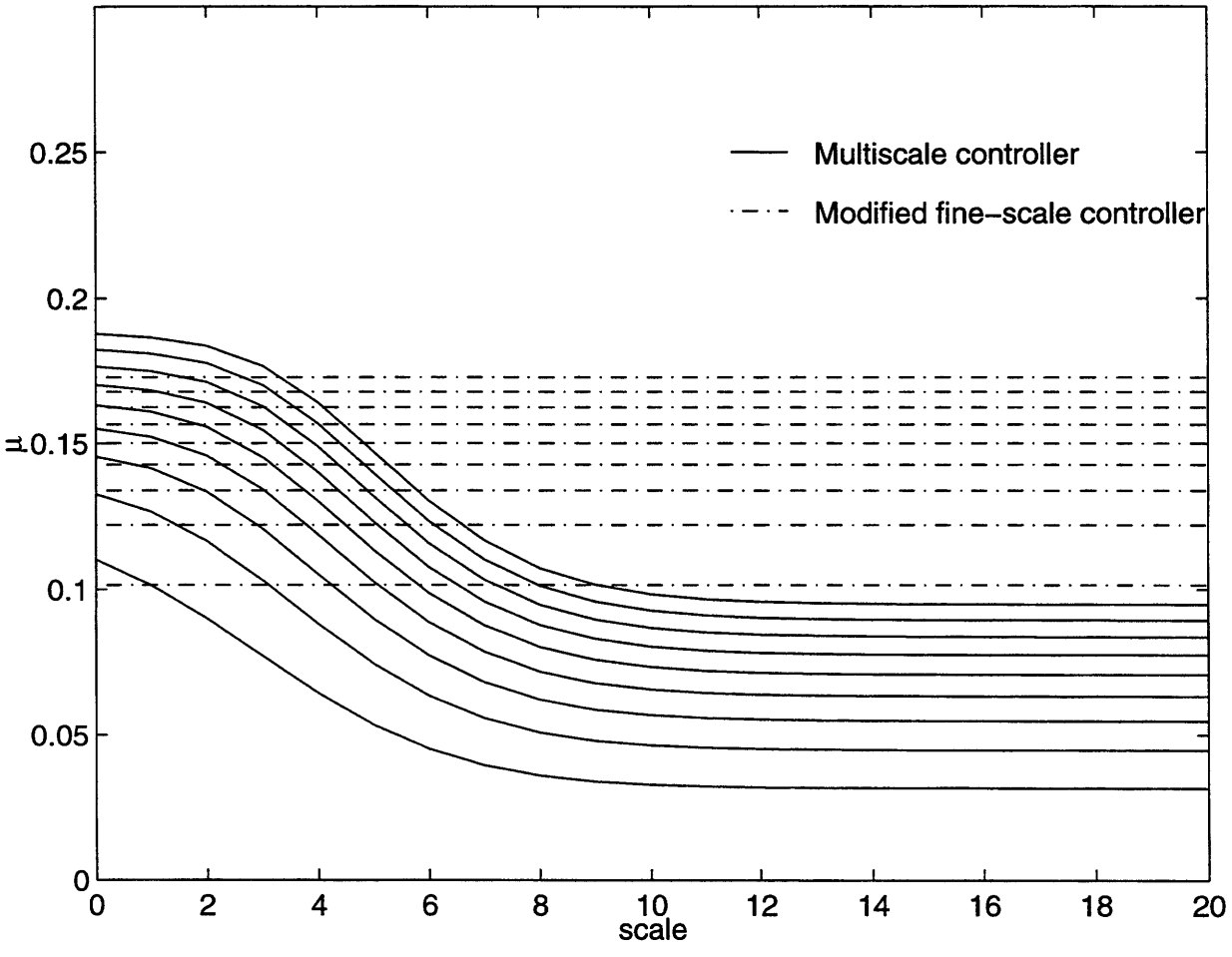


Figure 19: *Stationary server control policies for queueing systems servicing fractal renewal process input, for the case* $\gamma = 1.8$, $c(\mu) = 10\mu^2$, $h(i) = 0.01i$, $\beta = 0.0$. *The solid curves represent the optimal multiscale server. The dot-dashed curves denote the optimized policy obtained by scaling the fine-scale service rates taken from the multiscale policy.*

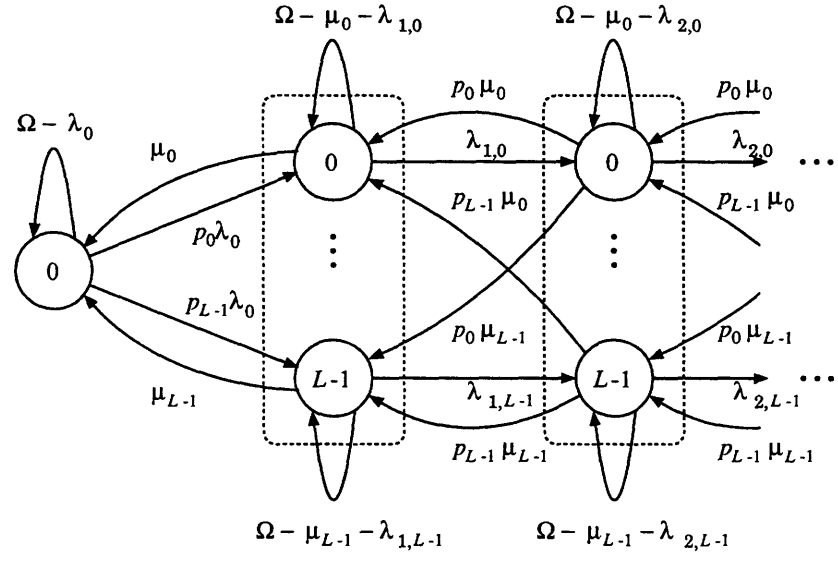
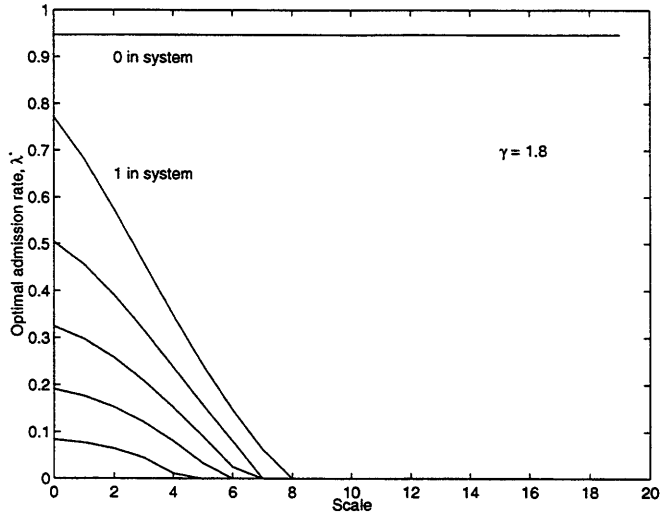
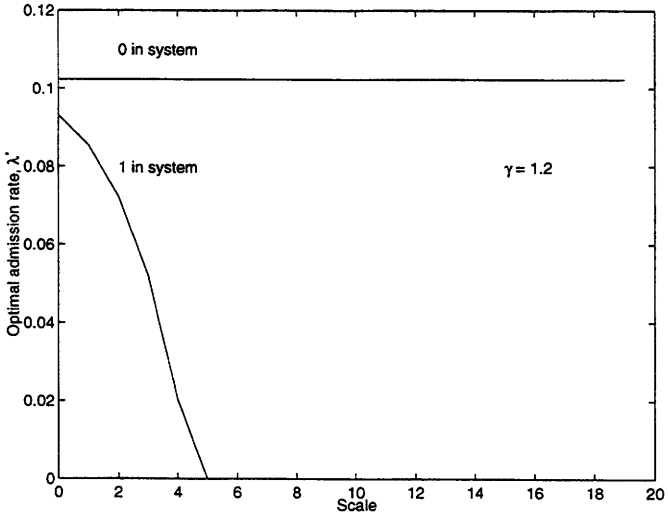


Figure 20: *The continuous-time Markov process used in flow control policy design. This process is obtained from the transposed multiscale birth-and-death process of Fig. 11 by adding self-transitions such that the total rate leaving each state is Ω. Also, the zeroth superstate is lumped into a single state.*



(a)



(b)

Figure 21: *State-based multiscale flow control for a queueing system with power-law service time and Poisson customer arrivals. The shape parameter of the service duration distribution is* $\gamma = 1.8$ *in (a) and* $\gamma = 1.2$ *in (b). Holding cost* $h(i) = 0.01i$*, throttling cost* $c(\lambda) = (\overline{\lambda} - \lambda)^2$*, and discount rate* $\beta = 0$ *are used.*