reproduces the statistical properties of (u_k^t, u_k^τ), from which one obtains

$$q_{t+1,\tau+1} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f\Big(B_t + \sqrt{C_{t,\tau}}z + \sqrt{C_{t,t} - C_{t,\tau}}u\Big) \times f\Big(B_\tau + \sqrt{C_{t,\tau}}z + \sqrt{C_{\tau,\tau} - C_{t,\tau}}v\Big)\, Dz\, Du\, Dv. \quad (31)$$

The bit-error probability can also be evaluated in the same way as above, resulting in

$$P_b^{t+1} = Q\left(\frac{B_t}{\sqrt{C_{t,t}}}\right). \quad (32)$$

ACKNOWLEDGMENT

The authors would like to thank Akira Mizutani at Tokyo Metropolitan University, Tokyo, Japan, and Dr. Yoshiyuki Kabashima at Tokyo Institute of Technology, Yokohama, Japan, for their helpful discussion.

REFERENCES

[1] T. Tanaka, "A statistical-mechanics approach to large-system analysis of CDMA multiuser detectors," *IEEE Trans. Inf. Theory*, vol. 48, no. 11, pp. 2888–2910, Nov. 2002.

[2] H. Nishimori, "Comment on 'Statistical mechanics of CDMA multiuser demodulation'," *Europhys. Lett.*, vol. 57, no. 2, pp. 302–303, Jan. 2002.

[3] D. Guo and S. Verdú, "Minimum probability of error of many-user CDMA without power control," in *Proc. IEEE Int. Symp. Information Theory*, Lausanne, Switzerland, Jun./Jul. 2002, p. 188.

[4] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco, CA: Morgan Kaufmann, 1988.

[5] S. Verdú, "Computational complexity of optimum multiuser detection," *Algorithmica*, vol. 4, pp. 303–312, 1989.

[6] M. K. Varanasi and B. Aazhang, "Multistage detection in asynchronous code-division multiple-access communications," *IEEE Trans. Commun.*, vol. 38, no. 4, pp. 509–519, Apr. 1990.

[7] ——, "Near-optimum detection in synchronous code-division multiple-access systems," *IEEE Trans. Commun.*, vol. 39, no. 5, pp. 725–736, May 1991.

[8] T. J. Richardson and R. L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 599–618, Feb. 2001.

[9] S. Amari and K. Maginu, "Statistical neurodynamics of associative memory," *Neural Netw.*, vol. 1, no. 1, pp. 63–73, 1988.

[10] M. Okada, "A hierarchy of macrodynamical equations for associative memory," *Neural Netw.*, vol. 8, no. 6, pp. 833–838, 1995.

[11] H. Nishimori, *Statistical Physics of Spin Glasses and Information Processing: An Introduction*. Oxford, U.K.: Oxford Univ. Press, 2001, ch. 7.

[12] K. H. Fischer and J. A. Hertz, *Spin Glasses*. New York: Cambridge Univ. Press, 1991.

[13] Y. Kabashima, "A CDMA multiuser detection algorithm on the basis of belief propagation," *J. Phys. A: Math. Gen.*, vol. 26, no. 43, pp. 11111–11121, Oct. 2003.

[14] D. Divsalar, M. K. Simon, and D. Raphaeli, "Improved parallel interference cancellation for CDMA," *IEEE Trans. Commun.*, vol. 46, no. 2, pp. 258–268, Feb. 1998.

[15] R. M. Buehrer and S. P. Nicoloso, "Comments on 'Partial parallel interference cancellation for CDMA'," *IEEE Trans. Commun.*, vol. 47, no. 5, pp. 658–661, May 1999.

Convexity in Queues With General Inputs

Michael J. Neely, *Member, IEEE,* and
Eytan Modiano, *Senior Member, IEEE*

***Abstract*—In this correspondence, we develop fundamental convexity properties of unfinished work and packet waiting time in a queue serving general stochastic traffic. The queue input consists of an uncontrollable background process and a rate-controllable input stream. We show that any moment of unfinished work is a convex function of the controllable input rate. The convexity properties are then extended to address the problem of optimally routing arbitrary input streams over a collection of K queues in parallel with different (possibly time-varying) server rates $(\mu_1(t), \dots, \mu_K(t))$. Our convexity results hold for stream-based routing (where individual packet streams must be routed to the same queue) as well as for packet-based routing where each packet is routed to a queue by probabilistic splitting. Our analysis uses a novel technique that combines sample path observations with stochastic equivalence relationships.**

***Index Terms*—G/G/1 queue, stochastic coupling.**

I. INTRODUCTION

In this correspondence, we examine a work-conserving queue with general stochastic inputs. We develop fundamental monotonicity and convexity properties of unfinished work and packet waiting time in the queue as a function of the packet arrival rate λ. The arrival process consists of two sets of input streams: an arbitrary and uncontrollable background stream $\theta(t)$, and a rate-controllable input stream $X(t)$ (Fig. 1). The rate-controllable stream $X(t)$ is composed of substreams $\{X_i(t)\}$, and its rate is varied in discrete steps by adding or removing one or more of these substreams as inputs to the queue. We show that any moment of unfinished work is a convex function of this discrete input rate. Under the special case of first-in first-out (FIFO) service, we show that waiting time moments are also convex. This convexity result is extended to treat continuous rate parameters λ, where the rate is determined by probabilistically splitting packets from an arbitrary stochastic input stream according to a splitting probability $p \in [0, 1]$.

We then apply these convexity results to address the problem of optimally routing input streams over a parallel collection of K queues with different server rates $(\mu_1, \dots, \mu_K)$, with the goal of minimizing a cost function. In the symmetric case, where the K queues are weighted equally in the cost function and have identical background processes, this convexity result implies that the uniform rate allocation minimizes cost. In the case of an asymmetric collection of K parallel queues, we present a sequentially greedy routing algorithm that is optimal.

Convexity of queue backlog and waiting time moments is an important structural property. For example, convexity is essential for establishing optimality of classical gradient-based routing algorithms [2], [13], and is also needed to prove optimality of threshold-based admission control strategies [3]. While it is intuitive that queue backlog increases convexly as input rates are increased, a precise formulation and proof of this result for general queueing systems has been a long-

Manuscript received January 18, 2002; revised September 22, 2004. The material in this correspondence was presented at IEEE INFOCOM, Anchorage, AK, April 2001.

M. J. Neely is with the Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089 USA (e-mail: mjneely@usc.edu).

E. Modiano is with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: modiano@mit.edu).

Coommunicated by L. Tassiulas, Associate Editor for Communication Networks.

Digital Object Identifier 10.1109/TIT.2004.840859

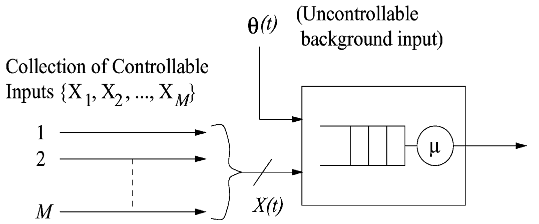
θ(t)
(Uncontrollable background input)
Collection of Controllable Inputs {X_1, X_2, ..., X_M}
1
2
M
X(t)
μ

Fig. 1. A work-conserving queue with server rate μ, a general background input $\theta(t)$, and rate-controllable inputs $X(t) = \{X_1(t), \ldots, X_M(t)\}$.

standing open problem. This correspondence solves the problem using a novel combination of sample path properties and stochastic equivalence relationships. Our analysis also reveals situations when the convexity property does not hold. Indeed, we show that the convexity property for unfinished work extends to systems with time-varying server rates $\mu(t)$, but that waiting time moments are not necessarily convex in this context.

As a motivating example, below we present a simple and well-known result concerning convexity of unfinished work as a function of the server rate μ. Let $X(t)$ be an input process to a queue that is initially empty, where $X(t)$ represents the number of bits that arrived to the queue up to time t. Let $U(t)$ represent the "unfinished work," or number of unprocessed bits, in the queue at time t. It is well known that $U(t)$ can be expressed using a supremum operator

$$U(t) = \sup_{\tau \geq 0} \{X(t) - X(t-\tau) - \mu\tau\}.$$

By convexity of this supremum operator, it immediately follows that the value of $U(t)$ at any time t is convex in the μ parameter. This is a sample path result that holds for any input $X(t)$, and it follows that averages and higher moments of unfinished work are also convex in μ. This observation is extended to finite buffer systems in [4] and [5].

However, now consider the example problem of sequentially applying input streams $X_1(t), X_2(t), \ldots, X_n(t)$, and showing that average unfinished work at a particular time t grows convexly with the number of streams added. Specifically, assume that $X_1(t)$ is any stochastic arrival process with arbitrarily correlated interarrival and service times, and that all streams $\{X_i(t)\}$ are independent but distributed identically to $X_1(t)$. The unfinished work can again be expressed in terms of the supremum operator

$$U(t) = \sup_{\tau \geq 0} \left\{ \sum_{i=1}^{n} [X_i(t) - X_i(t-\tau)] - \mu\tau \right\}.$$

However, in this case, the supremum operator is not helpful, as particular sample paths may not be convex in n (consider the example where the next stream added happens to have no arrivals during the interval $[0, t]$). In this correspondence, we use an alternate and novel technique to establish convexity when input streams $\{X_i(t)\}$ are *exchangeable* (which includes the case described above where all streams are independent). We first introduce a new function of the superposition of two streams that we call the *blocking function*. Analysis is performed by combining sample path properties of the blocking function with simple stochastic equivalence relationships, and all results follow directly from first principles of queueing systems.

Previous work on stochastic monotonicity and convexity in queues considers traffic with independence assumptions on packet inter-arrival times, service times, or both [6]–[12]. Convexity properties of parallel "GI/GI/1" queues with packet-based probabilistic routing are developed in [10]–[12], where it is shown under various independence assumptions that backlog moments in each queue are convex functions of the splitting probability, and hence uniform probabilistic splitting minimizes expected backlog in homogeneous systems among the class of all probabilistic splittings. A related result for homogeneous systems in [14] shows that uniform splitting is optimal for *arbitrary arrivals* in a system of parallel queues with independent and identically distributed (i.i.d.) exponential servers. These results are largely based on a theory of majorization and Schur-convex functions. Our approach is quite different and enables general analysis of both stream-based routing and packet-based probabilistic splitting. Independence assumptions are not required for the analysis, and our convexity result is the first of its kind to treat general stochastic inputs.

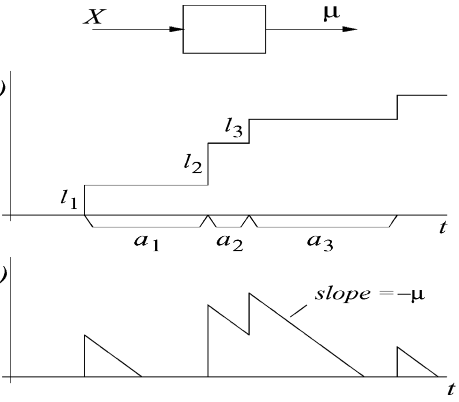

Fig. 2. A work-conserving queue and typical sample paths of accumulated and unfinished work.

In the next section, we define the blocking function. In Section III, we establish convexity properties of unfinished work and packet waiting time in terms of a discrete set of input streams. Probabilistic splitting and continuous input rates are treated in Section IV, and in Section V, we consider applications to routing over parallel queues. Time-varying server rates are treated in Section VI.

II. The Blocking Function

Consider a work-conserving queue with a single server that can process packets at a constant rate of μ bits per second. The queue is assumed to be initially empty at time $t = 0$. Variable-length packets from input stream X flow into the queue and are processed at the single server according to any work-conserving service discipline (such as first-in first-out (FIFO), last-in first-out, shortest packet first, generalized processor sharing, etc.). The input stream is characterized by two random processes: i) the sequence $\{a_k\}$ of inter-arrival times, and ii) the sequence $\{l_k\}$ of packet lengths.

The processes $\{a_k\}$ and $\{l_k\}$ are assumed to be ergodic with arrival rate λ and average packet length $\mathbb{E}\{L\}$, respectively. In general, inter-arrival times may be correlated with each other as well as jointly correlated with the packet length process. We maintain this generality by describing the input to the queue by the single random process $X(t)$, which represents the amount of bits brought into the queue as a function of time. As shown in Fig. 2, a particular input $X(t)$ is a nondecreasing staircase function. Jumps in the $X(t)$ function occur at packet arrival epochs, and the amount of increase at these times is equal to the length of the entering packet.

For a given queue with input process $X(t)$, we define the unfinished work process $U_X(t)$ as the total amount of unprocessed bits in the queueing system (buffer plus server) as a function of time. Note that for a system with a processor of rate μ and an amount of unfinished work $U_X(t)$, the quantity $U_X(t)/\mu$ represents the time required for the system to empty if no other packets were to arrive. It is clear that $U_X(t)$ is the same for all work-conserving service disciplines. It is completely determined by $X(t)$ as well as the server rate μ. An example unfinished

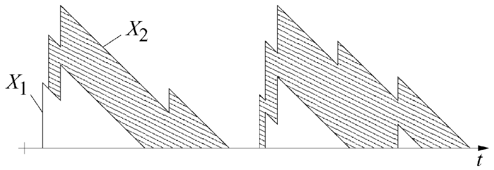

Fig. 3. An example sample path of the unfinished work function $U_{X_1+X_2}(t)$ in a system where X_1 packets have preemptive priority.

work function $U_X(t)$ is shown in Fig. 2. Notice the triangular structure and the fact that each new triangle emerges at packet arrival times and has a downward slope of $-\mu$.

We define the *superposition* of two input streams $X_1(t), X_2(t)$ as the sum process $X_1(t)+X_2(t)$. The following sample path observation holds for any arbitrary sample paths for processes $X_1(t), X_2(t)$:

Observation 1: For all times t, we have

$$U_{X_1+X_2}(t) \geq U_{X_1}(t) + U_{X_2}(t). \tag{1}$$

Thus, for any two inputs X_1 and X_2, the amount of unfinished work in a work-conserving queueing system with the superposition process $X_1 + X_2$ is always greater than or equal to the sum of the work in two identical queues with these same processes X_1 and X_2 entering them individually. Note that a simple special case of this observation is the fact that busy periods in a queue with input $X_1(t)$ alone are subintervals of busy periods in a queue with the superposition input $X_1(t)+X_2(t)$.

Proof: (Observation 1) We compare a queue with input $X_1(t)$ alone to a queue with $X_1(t)+X_2(t)$. Since $U_{X_1+X_2}(t)$ is the same for all work-conserving service disciplines, we can imagine that packets from the X_1 stream have preemptive priority over X_2 packets. The queueing dynamics of the X_1 packets are therefore unaffected by any low-priority packets from the X_2 stream. Thus, the $U_{X_1+X_2}(t)$ function can be written as $U_{X_1}(t)$ plus an extra amount $extra_{X_2}(t)$ due to the X_2 packets, as shown in Fig. 3. This extra amount (represented as the striped region in Fig. 3) can be viewed as the amount of unfinished work remaining in a queue with the X_2 input stream alone, where the server goes on idle "vacations" exactly at times when $U_{X_1}(t)$ is nonzero. Clearly, this unfinished work is greater than or equal to the unfinished work there would be if the server did not go on vacations—which is $U_{X_2}(t)$. Thus,

$$U_{X_1+X_2}(t) = U_{X_1}(t) + extra_{X_2}(t) \geq U_{X_1}(t) + U_{X_2}(t). \quad \square$$

This simple observation motivates the following definition.

Definition 1: The *Blocking Function* $\beta_{X_1,X_2}(t)$ between two streams X_1 and X_2 is the function

$$\beta_{X_1,X_2}(t) \triangleq U_{X_1+X_2}(t) - U_{X_1}(t) - U_{X_2}(t). \tag{2}$$

Thus, the blocking function is a random process that represents the extra amount of unfinished work in the system due to the blocking incurred by packets from the X_1 stream mixing with the X_2 stream.

Lemma 1: The blocking function has the following properties for all times t:

$$\begin{aligned} &\beta_{X_1,X_2}(t) \geq 0 && \text{(Nonnegativity)} \\ &\beta_{X_1,X_2}(t) = \beta_{X_2,X_1}(t) && \text{(Symmetry)} \\ &\beta_{X_1+X_2,X_3}(t) \geq \beta_{X_1,X_3}(t) && \text{(Monotonicity)}. \end{aligned}$$

The nonnegativity lemma is just a restatement of (1), while the symmetry property is obvious from the blocking function definition. The monotonicity property is the most interesting. Intuitively interpreted, the monotonicity property means that the amount of blocking incurred by the $(X_1 + X_2)$ process intermixing with the X_3 process is larger than the amount incurred by the X_1 process alone mixing with the X_3 process.

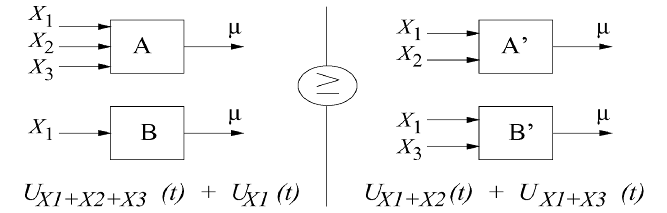
X_1
X_2
X_3
A
μ
X_1
B
μ
≥
X_1
X_2
A'
μ
X_1
X_3
B'
μ
$U_{X1+X2+X3}$ (t) + U_{X1} (t)
U_{X1+X2}(t) + U_{X1+X3} (t)

Fig. 4. A queueing illustration of the monotonicity property of the blocking function.

Proof: (Monotonicity) From the definition of the blocking function in (2), we find that the monotonicity statement is equivalent to the following inequality at every time t:

$$\begin{aligned} U_{X_1+X_2+X_3}(t) - U_{X_1+X_2}(t) - U_{X_3}(t)& \\ \geq U_{X_1+X_3}(t) - U_{X_1}(t) - U_{X_3}(t)&. \end{aligned}$$

Canceling and shifting terms, it follows that we must prove

$$U_{X_1+X_2+X_3}(t) + U_{X_1}(t) \geq U_{X_1+X_2}(t) + U_{X_1+X_3}(t). \tag{3}$$

We have illustrated (3) in Fig. 4. We thus prove that the sum of the unfinished work in Systems A and B of Fig. 4 is greater than or equal to the sum in A' and B'.

In a manner similar to the proof of Observation 1, we give packets from both the X_1 and X_2 streams preemptive priority over X_3 packets. The queues of Fig. 4 can thus be treated as having servers that take "vacations" from serving X_3 packets during busy periods caused by the other streams. Comparing the A and A' Systems, as well as the B and B' Systems, we have

$$U_{X_1+X_2+X_3}(t) = U_{X_1+X_2}(t) + \text{extra\_in\_System\_}A(t) \tag{4}$$

$$U_{X_1+X_3}(t) = U_{X_1}(t) + \text{extra\_in\_System\_}B'(t) \tag{5}$$

where extra_in_System_$A(t)$ represents the amount of unfinished work from X_3 packets in a queue whose server takes vacations during busy periods caused by the X_1 and X_2 streams. Likewise, extra_in_System_$B'(t)$ represents the amount of unfinished work from X_3 packets when vacations are only during X_1 busy periods. Since busy periods caused by the X_1 stream are subintervals of busy periods caused by the combined $X_1 + X_2$ stream, the X_3 packets in System A experience longer server vacations, and we have:

$$\text{extra\_in\_System\_}A(t) \geq \text{extra\_in\_System\_}B'(t). \tag{6}$$

Using (4)–(6) verifies (3) and concludes the proof. □

The three properties of Lemma 1 are sufficient to develop some very general convexity results for stochastic queues.

III. Exchangeable Inputs and Convexity

In this section, we use the blocking function to show that any moment of unfinished work in a queue is a convex function of the input rate λ. To do this, we must first specify how an arbitrary input process can be parameterized by a single rate value. The parameterization should be such that an input stream of rate 2λ can be viewed as being composed of two similar streams of rate λ. Otherwise, it is clear that the convexity result may not hold. Indeed, consider an input stream $X_1(t)$ delivering bursty data at rate λ, and another stream $X_2(t)$ also delivering data at rate λ according to some other, less bursty process. If $X_1(t)$ and $X_2(t)$ are sequentially added as inputs to a queue, the expected increment in unfinished work due to the additional $X_2(t)$ input may not be as large as the initial increment due to the $X_1(t)$ input. This happens if the $X_2(t)$ process is much smoother than $X_1(t)$, or if it is constructed to have packet arrivals precisely at idle periods of the queue with the $X_1(t)$ input alone.

Here, we consider the input rate λ as a discrete quantity that is varied by adding or removing substreams of the same "type" from the overall input process. We begin by developing the notion of exchangeable random variables [7].

Definition 2: A collection of M random variables are *exchangeable* if

$$p_{X_1,X_2,\ldots,X_M}(x_1,\ldots,x_M) = p_{\tilde{X}_1,\tilde{X}_2,\ldots,\tilde{X}_M}(x_1,\ldots,x_M) \tag{7}$$

for every $(\tilde{X}_1,\ldots,\tilde{X}_M)$ permutation of $(X_1,\ldots,X_M)$, where $p_{X_1,X_2,\ldots,X_M}(x_1,\ldots,x_M)$ is the joint density function.

Thus, exchangeable random variables exhibit a simple form of symmetry in their joint distribution functions. Definitions for random variables to be *conditionally exchangeable* given some event ω can be similarly defined: The distributions in (7) are simply replaced by conditional distributions. It is clear that any set of i.i.d. random variables are exchangeable. Thus, exchangeable variables form a wider class than i.i.d. variables, and hence statements that apply to exchangeable variables are more general. Unlike i.i.d. variables, however, it can be seen that if random variables $(X_1,\ldots,X_M)$ are conditionally exchangeable given some other random variable θ, then they are exchangeable.

We can extend this notion of exchangeability to include random *processes* that represent packet-arrival streams. The following definition captures the idea that for any sample path realization of exchangeable processes $(X_1(t),\ldots,X_M(t))$, the permuted sample path $(\tilde{X}_1(t),\ldots,\tilde{X}_M(t))$ is "equally likely."

Definition 3: Random processes $(X_1(t),\ldots,X_M(t))$ are *exchangeable* if for any permutation $(\tilde{X}_1(t),\ldots,\tilde{X}_M(t))$, we have

$$\mathbb{E}\left\{\Phi(X_1,\ldots,X_M)\right\} = \mathbb{E}\left\{\Phi(\tilde{X}_1,\ldots,\tilde{X}_M)\right\}$$

for every measurable operator $\Phi(\cdot)$ that maps the processes to a single real number.

Definition 4: Random processes $(X_1(t),\ldots,X_M(t))$ are conditionally exchangeable given process $\theta(t)$ if for every permutation $(\tilde{X}_1(t),\ldots,\tilde{X}_M(t))$, we have

$$\mathbb{E}\left\{\Phi(X_1,\ldots,X_M,\theta)\right\} = \mathbb{E}\left\{\Phi(\tilde{X}_1,\ldots,\tilde{X}_M,\theta)\right\}$$

for every real valued operator $\Phi(\cdot)$ that acts on the processes.

Hence, random processes are exchangeable if their joint statistics are invariant under every permutation. Note that the $\Phi(\cdot)$ operator maps a set of *sample paths* to a real number. For example, it could correspond to the mapping of the input process $X(t)$ to its unfinished work at a particular time t^*. Exchangeable processes have the same properties as their random variable counterparts. In particular, if processes $(X_1,\ldots,X_M)$ are exchangeable given a process θ, then

- processes $(X_1,\ldots,X_M)$ are exchangeable;
- processes $(X_n,\ldots,X_M)$ are exchangeable given processes $X_1,\ldots,X_{n-1},\theta$;
- if $\Psi(\cdot)$ is an operator that maps processes $X_1(t), X_2(t)$ and $\theta(t)$ to another process $Z(t) = \Psi(X_1,X_2,\theta)$, then $\Psi(X_1,X_2,\theta)$ and $\Psi(X_2,X_1,\theta)$ are exchangeable processes given $\theta(t)$.

The above properties are simple consequences of the definitions, where the last property follows by defining the operator

$$\tilde{\Phi}(X_1,X_2,\theta) \triangleq \Phi(\Psi(X_1,X_2,\theta),\Psi(X_2,X_1,\theta),\theta).$$

Below we provide three examples of exchangeable input processes that can act as input streams to a queueing system.

Example 1: Any general arrival processes $\{X_i(t)\}$ i.i.d. over M input lines.

Example 2: Any general arrival process $X(t)$ which is split into M streams by independently routing each packet to stream $i \in \{1,\ldots,M\}$ with equal probability.

Example 3: Any arbitrary collection of M processes $(X_1(t),\ldots,X_M(t))$ which are randomly permuted (with each permutation equally likely).

Notice that Example 1 demonstrates the fact that i.i.d. inputs are exchangeable. However, Example 2 illustrates that exchangeable inputs form a more general class of processes by providing an important set of input streams which are not independent yet are still exchangeable. Notice that this probabilistic routing can be extended to include "state-dependent" routing where the probability of routing to stream i depends on where the last packet was placed. The third example shows that an exchangeable input assumption is a good *a priori* model to use when an engineer is given simply a "collection of wires" from various sources, and has no *a priori* way of distinguishing the process running over "wire 1" from the process running over "wire 2."

We now examine how the unfinished work in a queue changes when a sequence of exchangeable inputs are added. Let $\theta(t)$ be an arbitrary background input process, and let $X_1(t)$ and $X_2(t)$ be two processes which are exchangeable given $\theta(t)$. Let $U_X(t)$ represent the unfinished work process as a function of time in a queue with an input process $X(t)$ running through it. Furthermore, let $f(u)$ represent any convex, nondecreasing function of the real number u for $u \geq 0$. We assume that the expected value of $f(U_X(t))$ is well defined for all t. (Note that expectations over functions of the form $f(u) = u^k$ represent kth moments of unfinished work). The following theorem shows that incremental values of queue cost are nondecreasing with each additional input.

Theorem 1: For any particular time t^*, we have

$$\mathbb{E}f\left(U_{\theta+X_1+X_2}(t^*)\right) - \mathbb{E}f\left(U_{\theta+X_1}(t^*)\right) \geq \mathbb{E}f\left(U_{\theta+X_1}(t^*)\right) - \mathbb{E}f\left(U_\theta(t^*)\right).$$

Proof: Define the following processes:

$$\begin{aligned}\Delta_1(t) &\triangleq U_{\theta+X_1}(t) - U_\theta(t) \\ \Delta_2(t) &\triangleq U_{\theta+X_1+X_2}(t) - U_{\theta+X_1}(t).\end{aligned} \tag{8}$$

By using the blocking function properties developed in the previous section, we find that for any time t we have

$$\begin{aligned}\Delta_2(t) &= U_{X_2}(t) + \beta_{\theta+X_1,X_2}(t) \\ &\geq U_{X_2}(t) + \beta_{\theta,X_2}(t) \qquad (9)\\ &= U_{\theta+X_2}(t) - U_\theta(t) \triangleq \tilde{\Delta}_1(t) \qquad (10)\end{aligned}$$

where (9) follows by the monotonicity property of the blocking function, and where we have defined a new process

$$\tilde{\Delta}_1(t) \triangleq U_{\theta+X_2}(t) - U_\theta(t)$$

in (10). Because $X_2(t)$ and $X_1(t)$ are exchangeable given $\theta(t)$, and because the $\tilde{\Delta}_1(t)$ and $\Delta_1(t)$ processes are derived from the same operator that maps inputs to differences in unfinished work (compare (8) and (10)), it follows that $\tilde{\Delta}_1(t)$ and $\Delta_1(t)$ are exchangeable processes given $\theta(t)$. Thus, for any time t^*, inequality (10) states that $\Delta_2(t^*)$ is a random variable that is always greater than or equal to another random variable which has the same distribution as $\Delta_1(t^*)$.

We now use an increasing increments property of nondecreasing, convex functions $f(u)$.

Fact: For nonnegative real numbers a, b, x, where $a \geq b$, we have

$$f(a + x) - f(a) \geq f(b + x) - f(b). \tag{11}$$

Using this fact and defining $a \triangleq U_{\theta+X_1}(t^*)$, $x \triangleq \Delta_2(t^*)$ and $b \triangleq U_\theta(t^*)$ we have

$$\begin{aligned} f(U_{\theta+X_1}(t^*) + \Delta_2(t^*)) - f(U_{\theta+X_1}(t^*)) \\ \geq f(U_\theta(t^*) + \Delta_2(t^*)) - f(U_\theta(t^*)) \quad (12) \\ \geq f(U_\theta(t^*) + \tilde{\Delta_1}(t^*)) - f(U_\theta(t^*)). \quad (13) \end{aligned}$$

Inequality (12) follows from from (11) and the fact that $U_{\theta+X_1}(t^*) \geq U_\theta(t^*)$ (from (1)). Inequality (13) follows because $f(u)$ is nondecreasing, and because $\Delta_2(t^*) \geq \tilde{\Delta}_1(t^*)$ (from (10)). Taking expectations of the inequality above, we find

$$\begin{aligned} \mathbb{E}f\left(U_{\theta+X_1}(t^*) + \Delta_2(t^*)\right) - \mathbb{E}f\left(U_{\theta+X_1}(t^*)\right) \\ \geq \mathbb{E}f\left(U_\theta(t^*) + \tilde{\Delta}_1(t^*)\right) - \mathbb{E}f\left(U_\theta(t^*)\right). \quad (14) \end{aligned}$$

Using the fact that $\tilde{\Delta}_1(t)$ and $\Delta_1(t)$ are exchangeable given $\theta(t)$, we can replace the $\mathbb{E}f\left(U_\theta(t^*) + \tilde{\Delta}_1(t^*)\right)$ term on the right-hand side of (14) with $\mathbb{E}f\left(U_\theta(t^*) + \Delta_1(t^*)\right)$, which yields the desired result. □

The preceding theorem can be used to immediately establish a convexity property of unfinished work in a work-conserving queue with a collection of exchangeable inputs. Assume we have such a collection of M streams $(X_1, \ldots, X_M)$ which are exchangeable given another background stream $\theta(t)$. Assume that each of the streams X_i has rate λ_δ. The total input process to the queue can then be viewed as a function of a discrete set of rates $\lambda = n\lambda_\delta$ for $n \in \{0, 1, \ldots, M\}$. Let $\mathbb{E}f\left(U[n\lambda_\delta]\right)$ represent the expectation of a function $f(\cdot)$ of the unfinished work (at some particular time t^*, which is suppressed for notational simplicity) when the input process consists of stream $\theta(t)$ along with a selection of n of the M exchangeable streams.

Hence,

$$\mathbb{E}f\left(U[n\lambda_\delta]\right) \triangleq \mathbb{E}f\left(U_{\theta+X_1+\cdots+X_n}(t^*)\right) \qquad (0 \leq n \leq M). \tag{15}$$

Theorem 2: At any specific time t^*, the function $\mathbb{E}f\left(U[\lambda]\right)$ is monotonically increasing and convex in the discrete set of rates λ $(\lambda = n\lambda_\delta,\ n \in \{0, 1, \ldots, M\})$. In particular, any moment of unfinished work is convex.

Proof: Convexity of a function on a discrete set of equidistant points is equivalent to proving successive increments are nondecreasing. Hence, the statement is equivalent to

$$\begin{aligned} \mathbb{E}f\left(U[(n+2)\lambda_\delta]\right) - \mathbb{E}f\left(U[(n+1)\lambda_\delta]\right) \\ \geq \mathbb{E}f\left(U[(n+1)\lambda_\delta]\right) - \mathbb{E}f\left(U[n\lambda_\delta]\right). \quad (16) \end{aligned}$$

Defining the "background stream" $\phi(t) = \theta(t) + X_1(t) + \cdots + X_n(t)$, we find that inequality (16) follows immediately from Theorem 1. □

A. Waiting Times

Notice that in Theorems 1 and 2, expectations were taken at any particular time t^*. It is not difficult to show that this property implies steady-state unfinished work is convex, whenever such steady-state limits exist. Moreover, we can allow t^* to be a time of special interest, such as the time when a packet from the X_1 stream enters the system. In FIFO queues, the unfinished work in the system at this special time represents the amount of waiting time W that the entering packet spends in the queue before receiving service. In this way, we show that waiting time increments are convex after the first stream is added. Specifically, for a system with a background input $\theta(t)$ and M inputs $\{X_1, \ldots, X_M\}$ which are exchangeable given $\theta(t)$, we define the following steady-state moments (which are functions of the discrete set of input rates $\lambda \in \{0, \lambda_\delta, 2\lambda_\delta, \ldots, M\lambda_\delta\}$).

$\mathbb{E}f\left(W_\theta^{(q)}[\lambda]\right), \mathbb{E}f\left(W_\theta[\lambda]\right)$ = Steady-state waiting time moment corresponding to the time a packet from background stream $\theta(t)$ spends in the *queue* and in the *system*, respectively, when the controllable input rate is λ.

$\mathbb{E}f\left(W_X^{(q)}[\lambda]\right), \mathbb{E}f\left(W_X[\lambda]\right)$ = Steady-state waiting time moment corresponding to the time a packet from a controllable input stream spends in the *queue* and in the *system*, respectively, when the controllable input rate is λ.

$\mathbb{E}f\left(N[\lambda]\right)$ = Steady-state moment of the number of packets in the system (from both the background and controllable input streams) when the controllable input rate is λ.

Formally, the steady-state waiting time moments are defined

$$\mathbb{E}f(W) \triangleq \lim_{K\to\infty} \frac{1}{K} \sum_{k=1}^{K} \mathbb{E}f(W_k)$$

where W_k represents the waiting time of the kth packet of the appropriate input stream. Likewise, the steady-state occupancy moment is defined

$$\mathbb{E}f(N) \triangleq \lim_{t\to\infty} \frac{1}{t} \int_0^t \mathbb{E}f\left(N(\tau)\right) d\tau.$$

Note that we have distinguished between waiting times of packets from the controllable input stream X and from the background input θ. This distinction is important for establishing convexity, as described by the following corollary and the subsequent example. Assuming these steady-state moments exist for the convex increasing function $f(u)$ of interest, we have the following.

Corollary 1: In FIFO queueing systems

a) $\mathbb{E}f\left(W_\theta^{(q)}[\lambda]\right)$ and $\mathbb{E}f\left(W_\theta[\lambda]\right)$ are nondecreasing and convex in the discrete set of rates $\lambda \geq 0$ (i.e., $\lambda = n\lambda_\delta$, $n \in \{0, 1, \ldots, M\}$);

b) $\mathbb{E}f\left(W_X^{(q)}[\lambda]\right)$ and $\mathbb{E}f\left(W_X[\lambda]\right)$ are nondecreasing and convex in the discrete set of rates $\lambda > 0$;

c) $\mathbb{E}\{N[\lambda]\}$ is nondecreasing and convex in the discrete set of rates $\lambda \geq 0$.

Caveat: Note that in b), waiting times for packets from the controllable input streams are not defined when $\lambda = 0$. Thus, convexity in this case is defined only for $\lambda > 0$. Further note that in c), the function $f(\cdot)$ is intentionally absent from the expectation, as we can only establish convexity of the first moment of packet occupancy in this general setting with variable length packets.

Proof: To prove a), let p be a certain packet from the θ input stream which arrives to the system at time t_p. When n of the controllable inputs are applied to the system, $U_{\theta+X_1+\cdots+X_n}(t_p^-)/\mu$ represents the amount of time packet p is buffered in the queue, and $U_{\theta+X_1+\cdots+X_n}(t_p^+)/\mu$ is the total system time for packet p. From Theorem 2, $\mathbb{E}f\left(U[\lambda]\right)$ is a nondecreasing, convex function of $\lambda \geq 0$ when unfinished work is evaluated at any time t^*, including times $t^* = t_p^-$ and $t^* = t_p^+$. Hence, the expected waiting time of packet p in the queue and in the system is a nondecreasing convex function of the controllable input rate. Because this holds for any packet p from the θ stream, the expected waiting time averaged over all θ packets is also convex, completing the proof.

To prove b), now let packet p represent a packet from the first controllable input stream X_1. Considering the sum process $\theta(t) + X_1(t)$ as a combined background stream with respect to inputs $\{X_2, \ldots, X_n\}$ (and noting that $\{X_2, \ldots, X_n\}$ remain exchangeable given $\theta(t) + X_1(t)$), from a) we know that the expected queueing time and system time of packet p is a nondecreasing convex function of $\lambda \geq \lambda_\delta$. Because inputs $\{X_1, \ldots, X_M\}$ are exchangeable, the expected waiting time of a packet from stream X_1 is not different from the expected waiting time of a packet from stream X_k (provided that stream X_k is also applied to the system), and the result follows.

To prove c), let $f(x) = x$. From b) we know that $\mathbb{E}\{W_X[\lambda]\}$ is nondecreasing and convex for $\lambda > 0$. It is straightforward to verify that for any such function, the function $\lambda\mathbb{E}\{W_X[\lambda]\}$ is nondecreasing and convex for $\lambda \geq 0$, where $\lambda\mathbb{E}\{W_X[\lambda]\}$ is defined to be 0 at $\lambda = 0$. Let λ_θ represent the rate of the $\theta(t)$ stream. By Little's theorem, it follows that

$$\mathbb{E}\{N[\lambda]\} = \lambda\mathbb{E}\{W_X(\lambda)\} + \lambda_\theta\mathbb{E}\{W_\theta[\lambda]\}$$

is nondecreasing and convex in λ, as this is the sum of nondecreasing convex functions. □

One might expect the waiting time $\overline{W}_{\text{av}}$ averaged over packets from both the controllable and uncontrollable input streams to be convex. However, note that

$$\overline{W}_{\text{av}}[\lambda] = \left(\frac{\lambda_\theta}{\lambda + \lambda_\theta}\right)\mathbb{E}\{W_\theta[\lambda]\} + \left(\frac{\lambda}{\lambda + \lambda_\theta}\right)\mathbb{E}\{W_X[\lambda]\}$$

is not necessarily convex even though both $\mathbb{E}\{W_\theta[\lambda]\}$ and $\mathbb{E}\{W_X[\lambda]\}$ are. Indeed, the following simple example shows that $\overline{W}_{\text{av}}[\lambda]$ may even *decrease* as λ increases.

Example: Let background input $\theta(t)$ periodically produce a new packet of service time 10 at times $t = \{0, 100, 200, \ldots\}$. Let input $X_1(t)$ consist of packets of service time 2 occurring periodically at times $t = \{50, 150, 250, \ldots\}$. Hence, packets from $\theta(t)$ and $X_1(t)$ never interfere with each other. We thus have $\overline{W}_{\text{av}}(0) = 10$ and $\overline{W}_{\text{av}}[\lambda_\delta] = (10+2)/2 = 6$.

IV. Convexity Over a Continuous Rate Parameter

In the previous section, we dealt with streams of inputs and demonstrated convexity of unfinished work and waiting time moments as streams are removed or added. Here, we extend the theory to include input processes that are parameterized by a continuous rate variable λ. The example to keep in mind in this section is packet-by-packet probabilistic splitting, where individual packets from an arbitrary packet stream are independently sent to the queue with some probability p. However, the results apply to any general "infinitely splittable" input, which are inputs that can be split into *substreams* according to some splitting method, as described as follows.

Definition 5: A packet input process $X(t)$ together with a splitting method is said to be *infinitely splittable* if

1) there exists a method of splitting $X(t)$ into *substreams*;
2) $X(t)$ or any of its substreams can be split into disjoint substreams of arbitrarily small rate. Any superposition of disjoint substreams of $X(t)$ is considered to be a substream;
3) any two (potentially nondisjoint) substreams that have the same rate are conditionally exchangeable given the rest of the process.

We emphasize that the above definition incorporates both the input process $X(t)$ and the method of splitting. Notice that any stochastic arrival process $X(t)$ is infinitely splittable when using the probabilistic splitting method of independently including packets in a new substream i with some probability p_i. Likewise, probabilistic splitting of the lead packet in systems where blocks of K sequential packets must be routed together can be shown to satisfy the conditions of infinite splittability.

However, not all splitting methods satisfy the above definition. Consider for example a "divide by 2" splitting method, where an input stream is split into two substreams by alternately routing packets to the first stream and then the second. Suppose the base input stream $X(t)$ has rate λ and consists of fixed-length packets of unit size. Under this splitting method, any substream of rate $\lambda\frac{k}{2^n}$ can be formed by collecting superpositions of disjoint substreams of rate $\lambda/2^n$ (where k and n are any integers such that $k \leq 2^n$). Thus, the first two conditions of infinite splittability are satisfied. However, it is not clear how a substream $\tilde{X}(t)$ of rate $\lambda/2$ is distributed. For example, the original stream $X(t)$ could be split into two substreams, one of which is randomly chosen as $\tilde{X}(t)$ and consists of every other packet arrival from $X(t)$. Alternately, the "divide by 2" splitting method might be used to form $\tilde{X}(t)$ by iteratively splitting $X(t)$ into eight substreams of rate $\lambda/8$, a random four of which are grouped together to form the rate $\lambda/2$ substream. Clearly, the two approaches to building a rate $\lambda/2$ substream do not generally lead to identically distributed processes, as the first approach leads to a rate $\lambda/2$ substream that never contains two successive packets from the original stream, while the second approach leads to a $\lambda/2$ substream that might contain two successive packets. Thus, the divide-by-2 splitting method satisfies the first two conditions of Definition 5 but not the third.

With the above definition, it can be seen that an infinitely splittable input process $X(t)$ can be written as the sum of a large number of exchangeable substreams. Specifically, it has the property that for any $\epsilon > 0$, there exists a large integer M such that

$$X(t) = \sum_{i=1}^{M} x_i(t) + \tilde{x}(t)$$

where $(x_1(t), \ldots, x_M(t))$ are exchangeable substreams, each with rate λ_δ, $\tilde{x}(t)$ has rate $\tilde{\lambda}_\delta$, and $\tilde{\lambda}_\delta < \lambda_\delta < \epsilon$.

We now use the blocking function to establish continuity of expected moments of unfinished work as a function of the continuous rate parameter λ. As before, these results also apply to waiting times in FIFO systems.

Again we assume that $f(u)$ is a nondecreasing convex function over $u \geq 0$. Suppose $X(t)$ is an infinitely splittable input process with total rate λ_{tot}. Suppose also that all exchangeable component processes of $X(t)$ are also exchangeable given the background input process $\theta(t)$. Let $\mathbb{E}f(U[\lambda_{\text{tot}}])$ represent the expectation of a function of unfinished work at a particular time t^* in a queue with this input and background process. We assume here that $\mathbb{E}f(U[\lambda_{\text{tot}}])$ is finite.

Theorem 3: $\mathbb{E}f(U[\lambda])$ can be written as a pure function of the continuous rate parameter λ, where $\lambda \in [0, \lambda_{\text{tot}}]$ is a rate achieved by some substream of the infinitely splittable $X(t)$ input. Furthermore, $\mathbb{E}f(U[\lambda])$ is a monotonically increasing and continuous function of λ.

Proof: The proof uses the machinery of the blocking function, and is given in the Appendix. □

The continuity property of Theorem 3 allows us to easily establish the convexity of any moment of unfinished work (and packet waiting time) in a general queue as a function of the continuous input rate λ. Let $X(t)$ be an infinitely splittable input process, and suppose that every collection of exchangeable components of $X(t)$ are also exchangeable given the background process $\theta(t)$. Then we have the following.

Theorem 4: At any particular time t^*, the function $\mathbb{E}f(U[\lambda])$ is convex over the continuous variable $\lambda \in [0, \lambda_{\text{tot}}]$. Likewise, if service is FIFO, then $\mathbb{E}f(W[\lambda])$ is also convex.

Proof: We wish to show that the function $\mathbb{E}f(U[\lambda])$ always lies below its chords. Thus, for any three rates $\lambda_1 < \lambda_2 < \lambda_3$, we must verify that

$$\mathbb{E}f(U[\lambda_2]) \leq \mathbb{E}f(U[\lambda_1]) + (\lambda_2 - \lambda_1)\frac{(\mathbb{E}f(U[\lambda_3]) - \mathbb{E}f(U[\lambda_1]))}{(\lambda_3 - \lambda_1)}. \tag{17}$$

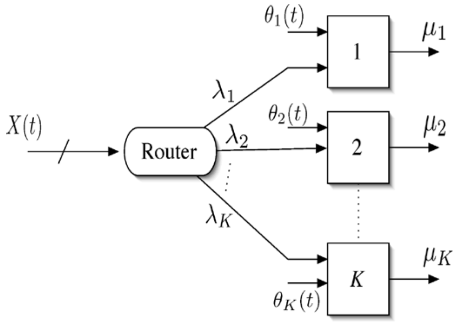

Fig. 5. Multiple queues in parallel with different background processes $\theta_i(t)$ and server rates μ_i.

We know from Theorem 2 in Section III that the unfinished work function is convex over a discrete set of rates when the input process is characterized by a finite set of M exchangeable streams $(x_1, \ldots, x_M)$. We therefore consider a discretization of the rate axis by considering the subprocesses $(x_1, \ldots, x_M)$ of the infinitely splittable process $X(t)$, where each x_i has a small rate δ. In this discretization, we have rates

$$\tilde{\lambda}_1 = k_1\delta, \quad \tilde{\lambda}_2 = k_2\delta, \quad \tilde{\lambda}_3 = k_3\delta \tag{18}$$

where the rates $(\tilde{\lambda}_1, \tilde{\lambda}_2, \tilde{\lambda}_3)$ can be made arbitrarily close to their counterparts $(\lambda_1, \lambda_2, \lambda_3)$ by choosing an appropriately small value of δ. Now, from the discrete convexity result, we know

$$\mathbb{E}f\left(U[\tilde{\lambda}_2]\right) \le \mathbb{E}f\left(U[\tilde{\lambda}_1]\right) + (\tilde{\lambda}_2 - \tilde{\lambda}_1)\frac{\mathbb{E}f\left(U[\tilde{\lambda}_1]\right) - \mathbb{E}f\left(U[\tilde{\lambda}_1]\right)}{(\tilde{\lambda}_3 - \tilde{\lambda}_1)}. \tag{19}$$

By continuity of the $\mathbb{E}f\,(U[\lambda])$ function, we can choose the discretization unit δ to be small enough so that the right-hand side of (19) is arbitrarily close to the right-hand side of the (currently unproven) inequality (17). Simultaneously, we can ensure that the left-hand sides of the two inequalities are arbitrarily close. Thus, the known inequality (19) for the discretized inputs implies inequality (17) for the infinitely splittable input. We thus have convexity of unfinished work at any point in time, which also implies convexity of waiting time in FIFO systems. □

V. Multiple Queues in Parallel

We now consider the system of K queues in parallel as shown in Fig. 5. The server for each queue k has rate μ_k and arbitrary background packet input processes $\theta_k(t)$. An arbitrary input process $X(t)$ also enters the system, and $X(t)$ is rate controllable in that a router can split $X(t)$ into substreams of smaller rate. These substreams can be distributed according to a K-tuple rate vector $(\lambda_1, \ldots, \lambda_K)$ over the multiple queues.

We consider both the case when $X(t)$ is an infinitely splittable process (as in packet-based probabilistic splitting), and the case when $X(t)$ is composed of a finite collection of M exchangeable streams. The problem in both cases is to route the substreams by forming an optimal rate vector that minimizes some network cost function. We assume the cost function is a weighted summation of unfinished work and/or waiting time moments in the queues. Specifically, we let $\{f_k(u)\}$ be a collection of convex, nondecreasing functions on $u \ge 0$. Suppose that the queues reach a steady-state behavior, and let $\mathbb{E}f\,(U_k[\lambda_k])$ represent the steady-state moment of unfinished work in queue k when an input stream of rate λ_k is applied. Let $\mathbb{E}f\,(W_k[\lambda_k])$ represent the steady-state moment of waiting time for queue k.

Theorem 5: If queues are work conserving and $X(t)$ is either a finitely[1] or infinitely rate splittable process given $\{\theta_k(t)\}$, then we have the following.

a) Cost functions of the form
$$C(\lambda_1, \ldots, \lambda_K) = \sum_{k=1}^{K} \mathbb{E}f_k(U_k[\lambda_k]) \tag{20}$$
are convex in the multivariable rate vector $(\lambda_1, \ldots, \lambda_K)$.

b) If service is FIFO, then cost functions of the form
$$C(\lambda_1, \ldots, \lambda_K) = \sum_{k=1}^{K} \lambda_k \mathbb{E}f_k(W_k[\lambda_k]) \tag{21}$$
are convex (where the $W_k[\lambda_k]$ values represent waiting times of packets from the controllable inputs).

c) If service is FIFO and $N_k(\lambda_k)$ represents the number of packets in queue k in steady state, then cost functions of the following form are convex:
$$C(\lambda_1, \ldots, \lambda_K) = \sum_{k=1}^{K} c_k \mathbb{E}\{N_k[\lambda_k]\} \quad (\{c_k\} \ge 0). \tag{22}$$

Proof: Since $\mathbb{E}f_k(W_k[\lambda_k])$ is convex and nondecreasing for $\lambda_k > 0$, the function $\lambda_k \mathbb{E}f_k(W_k[\lambda_k])$ is convex on $\lambda_k \ge 0$. Thus, the cost functions in a) and b) are summations of convex functions, so they are convex. Part c) follows from b) by noting that, from Little's theorem, $\mathbb{E}\{N\} = \lambda\mathbb{E}\{W\}$. □

Convexity of the cost function $C(\lambda_1, \ldots, \lambda_K)$ can be used to develop optimal rate distributions $(\lambda_1^*, \ldots, \lambda_K^*)$ over the simplex constraint $\lambda_1 + \cdots + \lambda_K = \lambda_{\text{tot}}$. For symmetric cost functions, which arise when the background processes $\{\theta_k(t)\}$ and the server rates $\{\mu_k\}$ are the same for all queues $k \in \{1, \ldots, K\}$, the optimal solution is particularly simple. Indeed, for the case of packet-based routing with continuous splitting rates, the uniform splitting $(\lambda/M, \ldots, \lambda/M)$ is optimal for symmetric systems. In the case of routing a discrete set of M streams over the queues, the optimal routing is the load balanced assignment of $\lceil M/K \rceil$ streams to $(M) \bmod (K)$ of the queues, and $\lfloor M/K \rfloor$ streams to the remaining queues.

In the nonsymmetric case with continuous splitting, convexity implies that optimal routing splits can be determined by a simple Lagrange multiplier calculation [2], [15]. In the case of stream-based routing, the optimal assignment is given by the following greedy algorithm. Let $C(M_1, \ldots, M_K)$ represent the cost function for the routing assignment $(M_1, \ldots, M_K)$. Assume $C(\cdot)$ is either fully known, or that it can be estimated.

Lemma 2: Given a convex cost function $C(M_1, \ldots, M_K)$ of the form specified in Theorem 5, the optimal allocation vector can be obtained by sequentially adding streams, greedily choosing at each iteration the queue that increases the total cost $C(\cdot)$ the least. This yields a cost-minimizing vector $(M_1^*, \ldots, M_K^*)$ after $M + K - 1$ evaluations/estimations of the cost function.

Proof: This lemma follows as a special case of a theory of integer optimization over separable convex functions (see [15]). A simplified and independent proof is given in [16]. □

VI. Time-Varying Server Rates

Here we consider the system of Fig. 1 when the constant server of rate μ is replaced by a time-varying server of rate $\mu(t)$. Sample path characteristics of the unfinished work $U_X(t)$ for time-varying servers are similar to those illustrated in Fig. 2 for constant server systems, with

[1] We note that convexity on a discrete set of points is equivalent to convexity of the piecewise-linear interpolation.

the exception that the $U_X(t)$ function decreases with a time-varying slope $-\mu(t)$.

Convexity analysis of $U_X(t)$ in this context is similar to the analysis for constant server rate systems. Indeed, defining the unfinished work blocking function $\beta_{X_1,X_2}(t)$ as before and literally repeating the same arguments of Section II, we can establish that the nonnegativity, symmetry, and monotonicity properties still hold for $\beta_{X_1,X_2}(t)$ in this time-varying context.

Likewise, we can define $N_X(t)$ as the (integer) number of packets in the system at time t, and define the occupancy blocking function $\alpha_{X_1,X_2}(t)$ as follows:

$$\alpha_{X_1,X_2}(t) \triangleq N_{X_1+X_2}(t) - N_{X_1}(t) - N_{X_2}(t).$$

While the occupancy blocking function may not satisfy the monotonicity property for general variable-length packets, it can be shown to satisfy nonnegativity, symmetry, and monotonicity in the special case when all packets have fixed lengths of L bits and service is nonpreemptive (see [16]).

Consequently, given a collection of K queues with background input processes $\{\theta_k(t)\}$ and server rate processes $\{\mu_k(t)\}$, together with a (finitely or infinitely distributable) input $X(t)$, we can establish the following.

Theorem 6: If the exchangeable components of $X(t)$ are exchangeable given $\{\theta_k(t)\}$ and $\{\mu_k(t)\}$, then $\sum \mathbb{E} f_k(U_k[\lambda_i])$ is convex in the rate vector $(\lambda_1, \ldots, \lambda_K)$. If all packets have a fixed length of L bits and service is nonpreemptive, then $\sum \mathbb{E} f_k(N_k[\lambda_k])$ is convex in the rate vector.

Recall from Little's theorem that if the expected waiting time $\mathbb{E}\{W(\lambda)\}$ is convex in λ, then so is the expected packet occupancy $\mathbb{E}\{N(\lambda)\}$. However, the converse implication does not follow. Indeed, below we provide a (counter) example which illustrates that—even for fixed-length packets under FIFO service—waiting times are not necessarily convex for time-varying servers.

(Counter) Example: Consider identical input processes X_1, X_2, X_3 which produce a single packet of length $L = 1$ periodically at times $\{0, 3, 6, 9, \ldots\}$. Let the server rate be periodic of period 3 with $\mu(t) = 1$ for $t \in [0, 2]$ and $\mu(t) = 100$ for $t \in (2, 3)$. Then $\mathbb{E}\{W_{X_1}\} = 1$, $\mathbb{E}\{W_{X_1+X_2}\} = 1.5$, and $\mathbb{E}\{W_{X_1+X_2+X_3}\} = 1.67$. Clearly, the increment in average waiting time when stream X_2 is added is larger than the successive increment when stream X_3 is added. Hence, waiting time is not convex in this time-varying server setting. □

Although waiting times are not necessarily convex, notice that minimizing $\overline{W}_{\rm tot}$ in a parallel queue configuration (Fig. 5) is accomplished by minimizing $\overline{N}_{\rm tot}$ (since $\overline{N}_{\rm tot} = \lambda_{\rm tot}\overline{W}_{\rm tot}$). For fixed-length packets, Theorem 6 ensures this is a convex optimization even for time-varying servers. Indeed, notice that expected occupancies $\mathbb{E}\{N_{X_1}\}$, $\mathbb{E}\{N_{X_1+X_2}\}$, and $\mathbb{E}\{N_{X_1+X_2+X_3}\}$ for the above example can be obtained by multiplying $\mathbb{E}\{W_{X_1}\}$, $\mathbb{E}\{W_{X_1+X_2}\}$, and $\mathbb{E}\{W_{X_1+X_2+X_3}\}$ by $\lambda = 1/3, 2/3$, and $3/3$, respectively, and the resulting values are convex. Indeed, the nonconvex values $1, 1.5$, and 1.67 become $\frac{1}{3}, 1$, and 1.67, which have increasing increments.

VII. Conclusion

We have developed general convexity results for queues with arbitrary stochastic inputs. These convexity results establish important structural properties of queueing systems and lead to simple algorithms for optimal routing over parallel queues. Analysis was performed by introducing a new function of two input streams that we call the blocking function. Nonnegativity, symmetry, and monotonicity properties of the blocking function were established. These properties are valuable tools for proving convexity of unfinished work and waiting time moments in queues with both discrete and continuous input rates λ, and can likely be used to establish convexity in other contexts.

Appendix
Continuity of Unfinished Work

Here we show that for any particular time t^*, $\mathbb{E}f(U[\lambda])$ is a continuous, monotonically increasing function of λ (Theorem 3 of Section IV). We utilize the following facts about convex, nondecreasing functions.

Fact 1: If $f(u)$ is nondecreasing and convex, then for any fixed $a \geq 0$ there is a function $g(a,x)$ such that $f(a+x) = f(a) + g(a,x)$, where $g(a,x)$ is a convex, nondecreasing function of x for $x \geq 0$.

Fact 2: Any convex, nondecreasing function $g(x)$ with $g(0) = 0$ has the property that $g(x_1 + x_2) \geq g(x_1) + g(x_2)$ for any $x_1, x_2 \geq 0$.

Let $X(t)$ represent the base input of the controllable stream, which is infinitely splittable and has total rate $\lambda_{\max}$. Note that

$$\mathbb{E}f(U[\lambda]) \triangleq \mathbb{E}f(U_{\theta+X_\lambda}(t^*))$$

where $\theta(t)$ is a background input and $X_\lambda(t)$ is any substream of $X(t)$ with rate λ. This is a well-defined function of λ because, by the properties of infinitely splittable inputs, all substreams with the same rate are identically distributed. The fact that $\mathbb{E}f(U[\lambda])$ is monotonically increasing in λ follows as a simple consequence of the nonnegativity property of the blocking function. Indeed, consider a substream $X_\delta(t)$ of rate δ. We have

$$\begin{aligned}\mathbb{E}f(U[\lambda+\delta]) &\triangleq \mathbb{E}f(U_{\theta+X_\lambda+X_\delta}(t^*))\\ &\geq \mathbb{E}f(U_{\theta+X_\lambda}(t^*)) \triangleq \mathbb{E}f(U[\lambda])\end{aligned}$$

proving monotonicity. In what follows, we prove the continuity property.

Proof: (Continuity) Here we prove that the function $\mathbb{E}f(U[\lambda])$ is continuous from the right with respect to the λ parameter. Left continuity can be proven in a similar manner.

Take any λ in the set of achievable rates. We show that

$$\lim_{\delta - 0} \mathbb{E}f(U[\lambda+\delta]) = \mathbb{E}f(U[\lambda]) \tag{23}$$

where δ is the rate of a component process of $X(t)$ that we make arbitrarily small. By monotonicity, if δ decreases to zero, then $\mathbb{E}f(U[\lambda+\delta]) - \mathbb{E}f(U[\lambda])$ decreases toward some limit ϵ, where $\epsilon \geq 0$. Suppose now that this inequality is strict, so that $\epsilon > 0$. We reach a contradiction by showing that there is a collection of M substreams with total rate $M\delta$ such that $\lambda + M\delta < \lambda_{\max}$ but

$$\mathbb{E}f(U[\lambda + M\delta]) > \mathbb{E}f(U[\lambda_{\max}]).$$

Consider disjoint component streams $\{x_1, \ldots, x_M\}$, each x_i of rate δ, for some yet-to-be-determined δ and M. We assume that these M substreams are disjoint from another substream X_λ of rate λ, all of which are components of the entire process $X(t)$. Let $\phi(t) = \theta(t) + X_\lambda(t)$, and let $U_{\phi+x_1+\cdots+x_M}$ represent the unfinished work in the system at some particular time t^*, with input processes $\{\theta, X_\lambda, x_1, \ldots, x_M\}$. From the definition of the blocking function, we have

$$\begin{aligned}U_{\phi+x_1+\cdots+x_M} &= U_{\phi+x_1+\cdots+x_{M-1}} + U_{x_M} + \beta_{\phi+x_1+\cdots+x_{M-1},x_M}\\ &\geq U_{\phi+x_1+\cdots+x_{M-1}} + U_{x_M} + \beta_{\phi,x_M}\end{aligned} \tag{24}$$

where (24) follows by the monotonicity property of the blocking function. By recursively iterating (24), we find

$$U_{\phi+x_1+\cdots+x_M} \geq U_\phi + \sum_{i=1}^{M}[U_{x_i} + \beta_{\phi,x_i}]. \tag{25}$$

Now applying the monotonically increasing, convex function $f(u)$ to both sides of (25) and writing $f(U_\phi + x) = f(U_\phi) + g(U_\phi, x)$ (from Fact 1), we have

$$\begin{aligned} f(U_{\phi+x_1+\cdots+x_M}) &\geq f(U_\phi) + g\left(U_\phi, \sum_{i=1}^{m}[U_{x_i} + \beta_{\phi,x_i}]\right) \\ &\geq f(U_\phi) + \sum_{i=1}^{M} g\left(U_\phi, [U_{x_i} + \beta_{\phi,x_i}]\right). \end{aligned} \tag{26}$$

Inequality (26) holds by application of Fact 2, as $g(U, x)$ is a convex function of x that is zero at $x = 0$. Now notice that

$$\begin{aligned} \mathbb{E}f\left(U[\lambda+\delta]\right) - \mathbb{E}f\left(U[\lambda]\right) &\triangleq \mathbb{E}f\left(U_{\phi+x_i}\right) - \mathbb{E}f\left(U_\phi\right) \\ &= \mathbb{E}f\left(U_\phi + U_{x_i} + \beta_{\phi,x_i}\right) - \mathbb{E}f\left(U_\phi\right) \\ &= \mathbb{E}\left\{g(U_\phi, U_{x_i} + \beta_{\phi,x_i})\right\} \end{aligned}$$

for any substream x_i of rate δ, where x_i and ϕ are disjoint. Hence, by assumption

$$\mathbb{E}\left\{g(U_\phi, U_{x_i} + \beta_{\phi,x_i})\right\} \geq \epsilon > 0. \tag{27}$$

Taking expectations of (26) and using (27), we find

$$\mathbb{E}f\left(U_{\phi+x_1+\cdots+x_M}\right) \geq \mathbb{E}f\left(U_\phi\right) + M\epsilon. \tag{28}$$

Inequality (28) above holds whenever $X_\lambda + x_1 + \cdots + x_M$ is a substream of the entire, infinitely splittable process $X(t)$. We now choose M large enough so that $M\epsilon$ is greater than the expectation of $f(U_{\theta+X})$ when the entire input $X(t)$ is applied, i.e., $M\epsilon > \mathbb{E}f\left(U_{\theta+X}\right)$. However, we choose a rate δ for each of the x_i substreams that is small enough to ensure $X_\lambda + x_1 + \cdots + x_M$ is a component process of $X(t)$. By monotonicity of $\mathbb{E}f\left(U[\lambda]\right)$, we have that

$$\mathbb{E}f\left(U_{\phi+x_1+\cdots+x_M}\right) \leq \mathbb{E}f\left(U_{\theta+X}\right) < M\epsilon.$$

But this contradicts (28), proving the theorem. □

References

[1] M. J. Neely and E. Modiano, "Convexity and optimal load distributions in work conserving $*/*/1$ queues," in *Proc. IEEE INFOCOM*, Anchorage, AK, Apr. 2001, pp. 1055–1064.

[2] D. P. Bertsekas and R. Gallager, *Data Networks*. Englewood Cliffs, NJ: Prentice-Hall, 1992.

[3] S. L. Spitler and D. C. Lee, "Optimization of call admission control for a statistical multiplexer allocating link bandwidth," *IEEE Trans. Autom. Control*, vol. 48, no. 10, pp. 1830–1836, Oct. 2003.

[4] K. Kumaran and M. Mandjes, "The buffer-bandwidth trade-off curve is convex," *Queueing Syst.*, vol. 38, pp. 471–483, 2001.

[5] K. Kumaran, M. Mandjes, and A. Stolyar, "Convexity properties of loss and overflow functions," *Oper. Res. Lett.*, vol. 31, pp. 95–100, 2003.

[6] D. Stoyan, *Comparison Methods for Queues and Other Stochastic Models*. Chichester, U.K.: Wiley, 1983.

[7] S. Ross, *Stochastic Processes*. New York: Wiley, 1996.

[8] C.-S. Chang, X. L. Chao, and M. Pinedo, "Monotonicity results for queues with doubly stochastic Poisson arrivals: Ross's conjecture," *Adv. Appl. Probab.*, vol. 23, pp. 210–228, 1991.

[9] F. Baccelli and P. Brémaud, *Elements of Queueing Theory*, 2nd ed. Berlin, Germany: Springer-Verlag, 2003.

[10] L. Gün, A. Jean-Marie, A. M. Makowski, and Tedijanto, "Convexity Results for Parallel Queues with Bernoulli Routing," Univ. Maryland, College Park, ISR Tech. Rep., 1990.

[11] C. S. Chang, X. Chao, and M. Pinedo, "A note on queues with Bernoulli routing," in *Proc. IEEE 29th Conf. Decision and Control*, 1990.

[12] A. Jean-Marie and L. Gün, "Parallel queues with resequencing," *J. ACM*, vol. 40, no. 5, Nov. 1993.

[13] F. Bonomi and A. Kumar, "Adaptive optimal load balancing in a nonhomogeneous multiserver system with a central job scheduler," *IEEE Trans. Comput.*, vol. 39, no. 10, pp. 1232–1250, Oct. 1990.

[14] G. Koole, "On the pathwise optimal Bernoulli routing policy for homogeneous parallel servers," *Math. Oper. Res.*, vol. 21, pp. 469–476, 1996.

[15] D. P. Bertsekas, A. Nedic, and A. E. Ozdaglar, *Convex Analysis and Optimization*. Boston, MA: Athena Scientific, 2003.

[16] M. J. Neely, "Dynamic power allocation and routing for satellite and wireless networks with time varying channels," Ph.D. dissertation, MIT, LIDS, Cambridge, MA, 2003.

Explicit Constructions of Algebraic-Geometric Codes

Hiren Maharaj, *Member, IEEE*

***Abstract*—We propose a simple construction of algebraic–geometric codes which are subcodes of Goppa codes and which coincide with Goppa codes in many cases. The codes we construct have the advantage that for an explicitly given extension of the rational function field, one can easily obtain explicit bases and therefore an exact formula for the dimension. Furthermore, we show that in many cases good upper and lower bounds for the minimum distance can be obtained.**

***Index Terms*—Algebraic–geometric codes, explicit constructions, Goppa codes.**

I. Introduction

Let F be an algebraic function field of a single variable with the finite field $\mathbb{F}_q$ as the full field of constants. Given n distinct rational places $P_1, P_2, \ldots, P_n$ of F and a divisor G of F with support disjoint from $\{P_1, P_2, \ldots, P_n\}$, the Goppa code $C_{\mathcal{L}}(P_1, P_2, \ldots, P_n; G) \subseteq \mathbb{F}_q^n$ is defined by

$$C_{\mathcal{L}}(P_1, P_2, \ldots, P_n; G) = \{(f(P_1), \ldots, f(P_n)) | f \in \mathcal{L}(G)\}.$$

Goppa codes have great utility because their parameters have the following estimates: if $g(F) < \deg G < n$ then $C_{\mathcal{L}}(P_1, P_2, \ldots, P_n; G)$ is an $[n, k, d]$ code where $k \geq \deg G - g(F) + 1$ and $d \geq n - \deg G$. For actual use of a Goppa code it is also desirable to have an explicit basis for the Riemann–Roch space $\mathcal{L}(G)$ from which one can obtain a basis for the code. However, in practice, obtaining such a basis is not always easy and in many cases it is a challenging problem (see, for example, [1], [13], [14]). There are useful algorithms for finding bases of Riemann–Roch spaces in general [3], but for even relatively small degree extensions of the rational function field, computation can be very slow.

Manuscript received January 22, 2004.

The author is with the Department of Mathematical Sciences, Clemson University, Clemson, SC 29634 USA (e-mail: hmahara@clemson.edu).

Communicated by A. E. Ashikhmin, Associate Editor for Coding Theory.
Digital Object Identifier 10.1109/TIT.2004.840896