Queueing Syst (2009) 63: 195–216
DOI 10.1007/s11134-009-9148-3

INVITED PAPER

Coding and control for communication networks

Wei Chen · Danail Traskov · Michael Heindlmaier · Muriel Médard · Sean Meyn · Asuman Ozdaglar

Received: 1 June 2009 / Revised: 20 October 2009 / Published online: 25 November 2009

Abstract The purpose of this paper is to survey techniques for constructing effective policies for controlling complex networks, and to extend these techniques to capture special features of wireless communication networks under different networking scenarios. Among the key questions addressed are:

(i) The relationship between static network equilibria, and dynamic network control.
(ii) The effect of coding on control and delay through rate regions.
(iii) Routing, scheduling, and admission control.

Through several examples, ranging from multiple-access systems to network coded multicast, we demonstrate that the rate region for a coded communication network may be approximated by a simple polyhedral subset of a Euclidean space. The polyhedral structure of the rate region, determined by the coding, enables a powerful *workload relaxation* method that is used for addressing complexity—the relaxation technique provides approximations of a highly complex network by a far simpler one.

These approximations are the basis of a specific formulation of an h-MaxWeight policy for network routing. Simulations show a 50% improvement in average delay performance as compared to methods used in current practice.

W. Chen · S. Meyn (✉)
Department of Electrical and Computer Engineering and the Coordinated Sciences Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL, USA
e-mail: meyn@control.csl.uiuc.edu

D. Traskov · M. Heindlmaier
Institute for Comm. Engineering, Technical University Munich, Munich, Germany

M. Médard · A. Ozdaglar
Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, USA

Keywords Routing · Scheduling · Networks · Coding · Information theory · Approximate dynamic programming

Mathematics Subject Classification (2000) 68M20 · 68M10 · 90B18 · 90B15 · 94A05

1 Introduction

One hundred years ago the Danish Scientist, Agner Krarup Erlang launched the field of queueing theory with his paper *The theory of probabilities and telephone conversations* [14]. In the 2009 workshop *100 Years of Queueing—The Erlang Centennial*, we recalled how Erlang helped to collect data to build models of telecommunication traffic, and that he was willing to pass through manhole covers for this purpose [5]. Based on these measurements, and ideas proposed by Johannsen in the 1907 manuscript [22], Erlang postulated probabilistic models of traffic statistics, described in [13, 14], which is the origin of the Poisson distribution typically assumed for arrival traffic in network models. The overall goal of his research was to obtain predictive models for the incredibly complex telephone exchange networks he was confronted with. Based on an M/D/1 model introduced in [13], he began the process of characterizing delay in queueing models based on a Markovian model, that was largely completed by Pollaczek and Khinchine in the 1930s, where average delay is expressed in their famous formula [24].

In this paper we consider the same issues of interest to Erlang one hundred years ago. We are concerned with finding characterizations or bounds to understand tradeoffs among delay, energy and network capacity. In addition, we want to use this insight to formulate control strategies for routing, scheduling, and admission control in telecommunication networks.

Following the style of Erlang's early papers, we identify a few key questions to be considered in this paper:

(A) What is the relationship between static network equilibria, and dynamic network control?
(B) What is the role of coding on control and delay analysis?
(C) How can we construct effective policies for controlling complex networks?

These questions have crisp answers when the answers are based on idealized models, and in the context of this paper, the answers are relevant even for refined network models that more accurately describe network behavior.

In this paper 'control' is restricted to the decision making processes required in routing packets in a communication network, admitting packets into the network, or determining priorities at a node in the network. A standing assumption is that a control solution should be based on a simple model that captures essential dynamics. If the control solution is sufficiently robust, then it will be effective even in the presence of inevitable modeling error.

Topics A and B are addressed in Sect. 2. A complete answer to A is provided, and we begin to address B by investigating the rate regions obtained under coding. This is

the first step for dynamic network control since our control policies are designed over the resulting rate regions. To address topic C we consider model reduction techniques to reduce complexity. The starting point is a relaxation technique introduced in the thesis of Neil Laws [27] to obtain lower bounds on achievable delay performance for stochastic networks.[1] An analogous relaxation can be used to construct a simple model for control design [34]. This however requires a sequence of steps:

1. We first construct a low-dimensional description of the network based on a 'workload relaxation' that is a generalization of Laws' relaxation.
2. A control solution is obtained for the relaxation, typically based on some formulation of optimality for the relaxation.
3. The solution for the relaxation is translated to obtain an effective policy for the original network.

The translation step might appear to be the most challenging. We borrow ideas from the MaxWeight policies of Tassiulas and Ephremides [16, 37, 39, 40] and state space collapse from heavy traffic theory [4, 23] to obtain a simple and effective approach to step 3. This approach to policy translation was introduced in [29], and developed further in [34]. The main ideas are summarized in Sect. 3, along with further background on MaxWeight policies.

2 Rate regions in network models

Until Sect. 3 we consider deterministic aspects of network control based on a fluid model.

2.1 Fluid models

In its most abstract form, a fluid model is a controlled differential equation with convex state and velocity constraints. The queue levels at time t are denoted $q_i(t)$, $1 \le i \le \ell$, or in vector form $q(t) \in \mathbb{R}_+^\ell$. There is a fixed arrival rate vector $\alpha \in \mathbb{R}_+^\ell$, and a convex set of achievable rates $\mathsf{R} \subset \mathbb{R}^\ell$ such that

$$\frac{d}{dt} q(t) = -r(t) + \alpha, \quad t \ge 0,$$

with $r(t) \in \mathsf{R}$ for each t. On letting V denote the velocity space $\mathsf{V} = \{-r + \alpha : r \in \mathsf{R}\}$, and $v(t) = -r(t) + \alpha \in \mathsf{V}$, we denote,

$$\frac{d}{dt} q(t) = v(t), \quad t \ge 0.$$

The following assumptions are imposed throughout the paper:

$$\begin{aligned} &\text{The rate region } \mathsf{R} \text{ is convex, with non-empty interior,} \\ &\text{This set contains the origin and the arrival-rate vector } \alpha. \end{aligned} \tag{1}$$

[1] See recent generalizations in [17, 36], the related LP approaches in [25, 26], and relaxation techniques from approximate dynamic programming, such as the work of Coffman and Mitrani [9] and its offspring [3].

The rate region depends on factors such as:

(i) Network topology, which may be a design choice.
(ii) Allowable routing decisions at each node.
(iii) The use of coding, and the type of coding strategy.

In particular, the use of coding can greatly expand the region R.

For the purposes of computation, the rate region is conceptualized in terms of achievable *equilibria* for the fluid model. Suppose that for a given $\alpha \in \mathbb{R}$, there is vector $r \in \mathsf{R}$ for which the system is at rest:

$$\frac{d}{dt}q(t) = -r + \alpha = 0.$$

Then we conclude that $\alpha = r \in \mathsf{R}$. The network load for a given α is defined by

$$\rho_\bullet(\alpha) = \min\bigl\{\rho : \rho^{-1}\alpha \in \mathsf{R}\bigr\}. \tag{2}$$

Given our assumption that R is convex and contains the origin, this representation can be inverted to give

$$\mathsf{R} = \bigl\{\alpha : \rho_\bullet(\alpha) \le 1\bigr\}. \tag{3}$$

In most applications we can work with a more structured fluid model defined as

$$\frac{d}{dt}q(t) = B\zeta(t) + \alpha, \quad t \ge 0 \tag{4}$$

where $\zeta(t) \in \mathbb{R}^{\ell_u}$ is a vector of allocation rates. In simple scheduling models we have $B = -[I - R^T]M$, where R is a routing matrix with binary entries and M is a diagonal matrix of maximal service rates. The allocation rates are restricted to a polyhedron of the form

$$\mathsf{U} := \bigl\{u \in \mathbb{R}^{\ell_m} : u \ge 0,\ Cu \le 1\bigr\}. \tag{5}$$

The *constituency matrix* C is an $\ell_m \times \ell_u$ matrix with binary entries. The rows of C correspond to *resources* $r = 1, \dots, \ell_m$. The rate region is thus given by

$$\mathsf{R} = \bigl\{\bigl[I - R^T\bigr]u : u \in \mathsf{U}\bigr\}.$$

In these models the rate region R and the velocity set V are polyhedral. The form of V is expressed in terms of generalized *workload vectors* $\{\xi^s : 1 \le s \le \ell_v\}$. On denoting $\rho_s = \alpha^T\xi^s$, there are binary elements $\{o_s : 1 \le s \le \ell_v\}$ such that the velocity space can be expressed as the intersection of half-spaces,

$$\mathsf{V} = \bigl\{v \in \mathbb{R}^\ell : v^T\xi^s \ge -(o_s - \rho_s),\ 1 \le s \le \ell_v\bigr\}. \tag{6}$$

In these models the network load can be expressed as the maximum of ρ_s over those s satisfying $o_s = 1$ (see [34, Theorem 6.1.1]).

In communication models without routing, the region R is precisely the multiple-access rate region. In models found in wireless applications we find that a polyhedral

model may be appropriate, but complex in the presence of multiple-access interference combined with coding [10, 11]. In wireless models with fading, the region R may not polyhedral.

This is the starting point to addressing question A: *What is the relationship between static network equilibria, and dynamic network control*? The rate region is typically envisioned as a static property of the network. Here it is a first step towards building a dynamic model of the network for purposes of control design.

We say that the fluid model is *stabilizable* if for each $x \in \mathbb{R}_+^\ell$, there is a trajectory $(\boldsymbol{q}, \boldsymbol{r})$ that is feasible for the fluid model, and a time $T < \infty$ such that $q(t) = 0$ for $t \geq T$. For a stabilizable fluid model we consider the following optimality problem: We let $c\colon \mathbb{R}^\ell \to \mathbb{R}_+$ denote a norm (typically the ℓ_1-norm), and denote the value function,

$$J^*(x) = \inf \int_0^\infty c\big(q(t)\big)\, dt, \quad q(0) = x, \tag{7}$$

where the infimum is over all feasible $(\boldsymbol{q}, \boldsymbol{r})$. This choice of optimality criterion is motivated by the fact that the value function J^* approximates the relative value function for an associated average-cost optimization problem for a stochastic model [8, 30, 31].

However, computation of J^* is infeasible in all but the simplest models. To obtain an approximation to this optimal control problem we relax the constraints on $v(t)$ in the fluid model. Details of this approach can be found in [29, 32–34], and will be reviewed briefly in the following examples.

2.2 Interference constraints and combinatorial complexity

We first consider a system with routing only and no coding. This is the most common setting considered in network control in the literature. Shown in Fig. 1 is a network with 18 separate arrival processes, and 40 nodes. To model interference we adopt the *half-duplex constraint* model in which no node can send and receive simultaneously.

To construct a model of the form (4) we require a separate buffer for each flow at each node. This leads to fluid model in which $\ell = 40 \times 18 = 720$ (!). Shown on the right-hand side of Fig. 1 are three examples of two-dimensional slices of the region R (restricted to the positive orthant), computed using the formula (3). These results suggest that the rate region may be approximated by a polyhedron with a moderate number of faces. This is motivation for the workload relaxation introduced next for a simpler multiple-access model.

2.3 Multiple-access model

The constraint imposed in Sect. 2.2 is extremely pessimistic—by analogy, consider a party in which each guest feels that he or she can speak only if everyone else in the room is silent. Party guests can effectively filter out conversations of others while participating in their private conversations, and this efficiency can be replicated in communication systems through coding.

Consider the multiple-access system illustrated in Fig. 2, where two users transmit to a single receiver. The two users share a single channel which is corrupted by

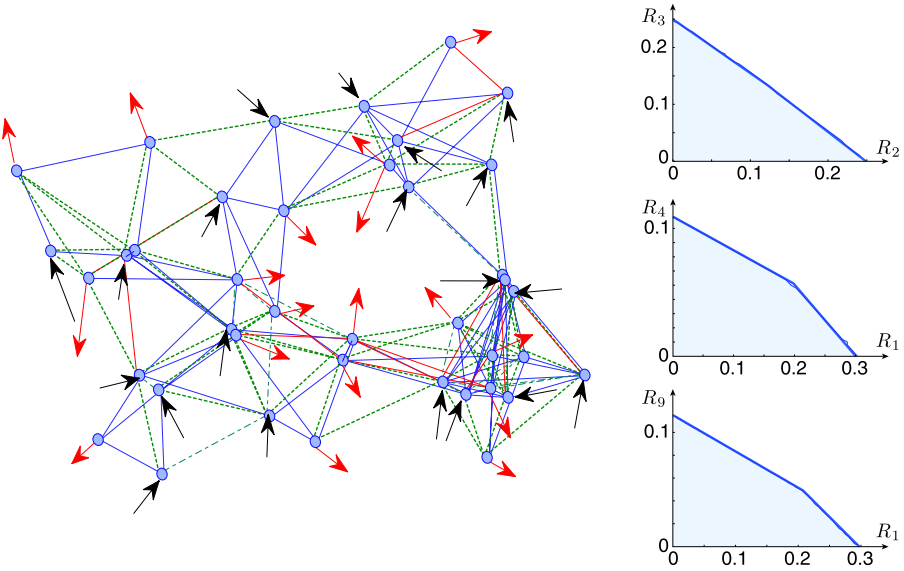

Fig. 1 Shown on *the left* is a network with 40 nodes, 18 flows, and 720 buffers. On *the right* is shown three examples of two-dimensional "slices" of the rate region R that is obtained under the half-duplex constraint. Even with many flows and hundreds of buffers, the rate region is not very complex

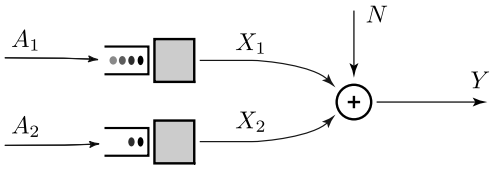

Fig. 2 Multiple-access communication system with two users and a single receiver

additive white Gaussian noise (AWGN), denoted by N in the figure. The queueing system has two buffers that receive arriving packets of data modeled as the i.i.d. sequences $A_i = (A_i(1), A_i(2), \ldots)$, with finite mean $\alpha_i = \mathsf{E}[A_i(t)]$, $i = 1, 2$. Data at queue i is stored in its respective queue until it is coded and sent to the receiver. The output of the system seen at the receiver is given by $Y(t) = X_1(t) + X_2(t) + N(t)$, where $\boldsymbol{N}$ is i.i.d. Gaussian with finite second moment σ_N^2, and independent of the two inputs $\{X_1, X_2\}$. It is assumed that user i is subject to the average power constraint $\mathsf{E}[X_i(t)^2] \le P_i$.

In this example and others, there are two steps required for network control: A possibly combinatorial problem at the lower level defines the set R of achievable rates. At a higher level, we have a queueing model where the rates in R are chosen as a function of time based on cost considerations on a longer time-horizon.

The set of all possible data rates is given by the Cover–Wyner region illustrated in Fig. 3 [11]. Any pair $(R_1, R_2) \in \mathsf{R}$ within this simple pentagonal region can be

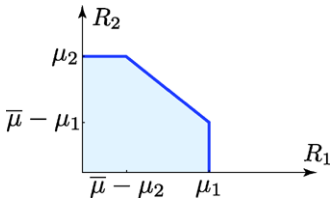

Fig. 3 Achievable rates in the multiple-access model

achieved through independent coding at the two buffers.[2] The rate region is a strict subset of R in the absence of multiple-access coding: the triangular region in the positive orthant consisting of points below the line connecting μ_1 and μ_2.

To complete the construction of the fluid model, we construct the matrices B and C and the vector α consistently with the rate region shown in Fig. 3. We let 'μ' denote a maximal processing rate: $\mu_1 = \mathcal{C}(P_1)$ and $\mu_2 = \mathcal{C}(P_2)$, where $\mathcal{C}(P) = \frac{1}{2}\log(1 + P/\sigma_N^2)$ for any $P \geq 0$. The dynamics of the two-dimensional fluid model are expressed as

$$\frac{d}{dt}q_1(t) = -\mu_1\zeta_1(t) + \alpha_1,$$
$$\frac{d}{dt}q_2(t) = -\mu_2\zeta_2(t) + \alpha_2.$$

The matrix B and the constituency matrix C are given by

$$B = -\begin{bmatrix} \mu_1 & 0 \\ 0 & \mu_2 \end{bmatrix}, \qquad C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \mu_1/\overline{\mu} & \mu_2/\overline{\mu} \end{bmatrix}, \tag{8}$$

where $\overline{\mu} = \mathcal{C}(P_1 + P_2)$. The constituency matrix C has three rows, corresponding to the three faces of the rate region R.

There are several representations for the rate region. We have $\mathsf{R} = \{-Bu : u \in \mathsf{U}\}$ where U is given in (5). We can also obtain an expression for the rate region in terms of workload. Recall that $\mathsf{R} = -\mathsf{V}_0$, where $\mathsf{V}_0 = \{v \in \mathbb{R}^\ell : v^T\xi^s \geq -o_s,\ 1 \leq s \leq \ell_v\}$ is the velocity space for the arrival-free model (with $\alpha = 0$). The vector ξ^s is called a workload vector if $o_s = 1$. There are three workload vectors in this model:

$$\xi^1 = (1/\mu_1, 0)^T, \qquad \xi^2 = (0, 1/\mu_2)^T, \qquad \xi^3 = (1/\overline{\mu}, 1/\overline{\mu})^T. \tag{9}$$

The velocity space for the fluid model is obtained by shifting V_0 by the mean arrival-rate vector α. That is, $\mathsf{V} = \{V_0 + \alpha\} := \{v + \alpha, v \in V_0\}$. A typical case is illustrated in Fig. 4.

It is assumed that a cost function $c\colon \mathbb{R}_+^2 \to \mathbb{R}_+$ is given that is convex, and vanishes only at the origin. For simplicity we shall concentrate on a linear cost function

[2]The Cover–Wyner region is constructed under the assumption that each queue has an infinite supply of bits for coding. Strict limits on delay will result in a smaller rate region.

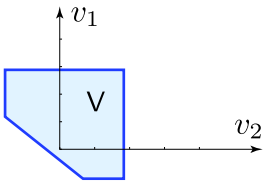

Fig. 4 Velocity space for the multiple-access model

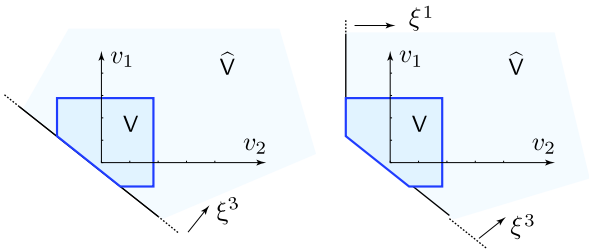

Fig. 5 Two instances of the relaxed velocity space for the multiple-access model

of the form $c(x) = c_1x_1 + c_2x_2$, with $c_i > 0$ for each i. Our interest is in minimizing the average cost over all policies.

We obtain an approximate solution using a workload relaxation. Shown on the left-hand side of Fig. 5 is the half-space $\widehat{\mathsf{V}}$ that defines a one-dimensional workload relaxation using the workload vector ξ^3, and on the right is a two-dimensional relaxation that maintains the two constraints defined by ξ^1 and ξ^3. In either case, the fluid model is defined to be the controlled model in continuous time, whose velocity is constrained to the region $\widehat{\mathsf{V}}$,

$$\frac{d}{dt}\hat{q}(t) = \hat{v}(t), \quad \hat{v}(t) \in \widehat{\mathsf{V}},\ t \geq 0.$$

A relaxation for the stochastic model is defined similarly [34].

There are many results establishing solidarity between a relaxation and the original stochastic model. The strongest such result is obtained for a general class of queueing network models in [29]: If in the one-dimensional relaxation the workload vector is chosen corresponding to the highest load, then under general conditions a policy for the stochastic model based on the optimal policy for the relaxation will be approximately optimal, with logarithmic regret as the load tends to unity.

We now focus on the one-dimensional relaxation for the fluid model, based on the workload vector ξ^3. There is a policy that is *pathwise optimal* in this case. To see why, first note that the movement of $\hat{q}(t)$ is unconstrained in directions orthogonal to ξ^3 (the velocity space for $\hat{q}(t)$ is a half-space defined by ξ^3). Hence, in an optimal solution, the cost $c(\hat{q}^*(t))$ will be minimal among all values $c(x)$ for which ${\xi^3}^T x = {\xi^3}^T \hat{q}^*(t)$.

This motivates the *effective cost*, defined for $w \in \mathbb{R}_+$ by

$$\overline{c}(w) = \min\bigl\{c(x) : {\xi^3}^T x = w,\ x \in \mathbb{R}_+^2\bigr\}. \tag{10}$$

The basic feasible solutions of the linear program (10) are

$$x^{*1} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \overline{\mu} w, \qquad x^{*2} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \overline{\mu} w,$$

and the effective cost is thus $\overline{c}(w) = \overline{\mu} w \min(c_1, c_2)$.

In an optimal solution we have $c(\hat{q}^*(t)) = \overline{c}(\hat{w}^*(t))$ for all $t > 0$, where $\hat{w}^*(t) := \xi^{3T} \hat{q}^*(t)$.

Supposing further that $c_1 < c_2$, the optimal solution will satisfy $\hat{q}_2^*(t) = 0$ and $\hat{q}_1^*(t) = \overline{\mu} \hat{w}(t) = \overline{\mu} \hat{w}(0) - \overline{\mu}(1 - \rho_3)t$ for $0 < t \le \hat{w}(0)/(1 - \rho_3)$, where $\rho_3 = \xi^{3T} \alpha$. The resulting infinite-horizon cost is a quadratic function of the initial workload:

$$\hat{J}^*(x) = \int_0^\infty c\big(\hat{q}^*(t)\big)\, dt = \frac{1}{2} \overline{\mu} c_1 \frac{w^2}{1 - \rho_3}, \quad \hat{q}^*(0) = x,\ w = \xi^{3T} x,\ x \in \mathbb{R}_+^2. \tag{11}$$

We consider next the two-dimensional relaxation based on the workload vectors ξ^1 and ξ^3. For linear cost, the effective cost is in the solution to the linear program,

$$\begin{aligned} \overline{c}(w) = \min \quad & c_1 x_1 + c_2 x_2 \\ \text{s.t.} \quad & \xi^{1T} x = w_1, \\ & \xi^{3T} x = w_3 \quad x \ge 0. \end{aligned} \tag{12}$$

We can invert the constraint equations to obtain x as a function of w. On letting Ξ denote the 2×2 matrix with rows ξ^{1T} and ξ^{3T}, we have

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \Xi^{-1} \begin{pmatrix} w_1 \\ w_3 \end{pmatrix} = \begin{pmatrix} \mu_1 w_1 \\ \overline{\mu} w_3 - \mu_1 w_1 \end{pmatrix},$$

provided w is feasible, which requires $w \in \mathsf{W} = \{(w_1, w_3) \in \mathbb{R}_+^2 : w_1 \ge 0,\ w_3 \ge \mu_1 w_1 / \overline{\mu}\}$. The effective cost is defined for feasible w by

$$\overline{c}(w) = c^T \Xi^{-1} x = (c_1 - c_2)\mu_1 w_1 + c_2 \overline{\mu} w_3.$$

The monotone region W^+ is defined to be the set of all $w \in \mathsf{W}$ such that $\overline{c}(w') \ge \overline{c}(w)$ whenever $w' \ge w$ (component-wise). The effective cost is called monotone if $\mathsf{W}^+ = \mathsf{W}$. In this example, the effective cost is monotone if and only if $c_1 \ge c_2$. In this case there exists a pathwise optimal solution for the relaxation in which the components of the workload process are minimal.

If $c_1 < c_2$ then W^+ is a strict subset of W. In such cases an optimal solution minimizing the infinite horizon cost $\hat{J}^*$ will be defined by a cone W^* satisfying $\mathsf{W}^+ \subset \mathsf{W}^* \subset \mathsf{W}$, with $\hat{w}^*(t) \in \mathsf{W}^*$ for all $t > 0$, where $\hat{w}^*(t) = \Xi \hat{q}^*(t)$ [34].

2.4 Single-flow communication network

To see how relaxations may be constructed for networks we consider the simplest formulation of the routing model, in which coding among different connections is

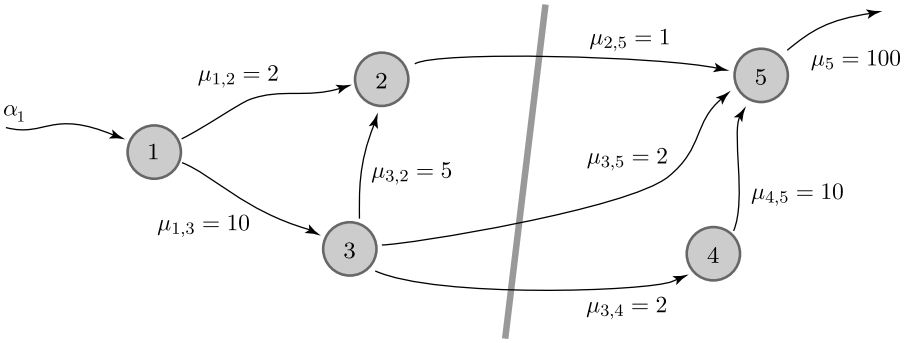

Fig. 6 Single-user routing model with min-cut $A = \{1, 2, 3\}$

disallowed. The impact of coding in a network setting will be addressed in the following subsection.

In a fluid model, the ℓ-dimensional vector of queue-lengths evolves according to the differential equation,

$$\frac{d^+}{dt} q_i(t) = \alpha_i + \sum_{k=1}^{\ell} \zeta_{ki} \mu_{ki} - \sum_{j=1}^{\ell} \zeta_{ij} \mu_{ij}, \tag{13}$$

where μ_{ki} is the maximal rate on the link joining nodes k and i, and the rate constraints are decoupled,

$$\mathsf{U} := \{0 \le \zeta_{ij} \le 1,\ 1 \le i, j \le \ell\}.$$

Consider the single-user routing model shown in Fig. 6. The capacities on each of seven links are as indicated. The links are unidirectional so, for example, we have $\mu_{5,4} = 0$. The imposition of the rate $\mu_5 = 100$ at node 5 is for the purposes of constructing a linear fluid model of the form (4).

We first look at this model from the point of view of the Min-Cut Max-Flow Theorem. Based on this result we can conclude that the maximal rate α_1^* that the network can support in equilibrium is the sum of the rates on the minimal cut shown:

$$\alpha_1^* = \mu\big(A, A^c\big) = 5.$$

The fluid model is stabilizable for any arrival rate satisfying $\alpha_1 < 5$, and the network load is given by $\rho_\bullet = \alpha_1/5$.

The conclusion $\rho_\bullet = \alpha_1/5$ can also be reached by constructing the workload vectors that define V. To construct a fluid model we first order the vector of activity rates as follows:

$$\zeta = (\zeta_{1,2},\ \zeta_{1,3},\ \zeta_{2,5},\ \zeta_{3,2},\ \zeta_{3,4},\ \zeta_{3,5},\ \zeta_{4,5},\ \zeta_5)^T$$

so that (13) is of the form (4) with $\alpha = (\alpha_1, 0, 0, 0, 0)^T$, and

$$B = \begin{bmatrix} -2 & -10 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & -1 & 5 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & -5 & -2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & -10 & 0 \\ 0 & 0 & 1 & 0 & 0 & 2 & 10 & -100 \end{bmatrix}.$$

The workload vector $\xi = (1/5, 1/5, 1/5, 0, 0)^T$ consistently expresses the load by $\rho_\bullet = \alpha^T \xi = \alpha_1/5$. For given $x \in \mathbb{R}^5_+$, the workload $w = \xi^T x$ is given by

$$w = \frac{1}{\mu(A, A^c)} \times \text{Total fluid above network cut}$$

where $\mu(A, A^c) = 5$ is the maximal rate that fluid can flow across the cut.

Just as in the multiple-access example we can construct a one-dimensional workload relaxation, and compute the associated optimal policy.

2.5 Multicast and network coding

In this section we consider coding to allow several receivers to share a connection. A common application in networks is multicasting, i.e. the transmission of information from a source node s to a subset of network nodes $\mathcal{T}$. Let $(s, \mathcal{T}, R)$ denote a multicast connection with rate R. In general, *network coding* is necessary to achieve capacity [1]; practical capacity achieving schemes have been proposed in [19–21]. Note that, for a single point-to-point connection, a necessary and sufficient condition for a flow to be established between a sender and a receiver is given by the min-cut max-flow conditions. Such a condition trivially remains necessary when we consider a single sender and several receivers. However, when using routing schemes to establish connections, one must use combinations of trees. In such trees, the min-cut condition may not be sufficient, since in effect trees may compete with each other for capacity. This competition for capacity among trees will render the min-cut condition non-sufficient when trees serving different receiver nodes compete for capacity along a link. Under network coding, however, different receivers in a multicast connection do not compete with each other for resources. Thus, under network coding conditions, the maintenance of min-cut conditions between the set of senders and every receiver individually is not only necessary, but also a sufficient condition to achieve capacity.

Although network coding obviates competition for resources among receivers in a multicast connection in wireless models, interference from simultaneous transmissions has to be taken into account. In effect, interference changes the underlying network according to the presence or absence of connections. A conservative approach for managing interference is to assign to each node an orthogonal channel and therefore avoid interference at the price of reduced bandwidth efficiency. The approach taken in [41], on the other hand, is a scheduling technique designed for network coded multicast traffic. Its basic idea is, in short, to activate subsets of neighbors in a manner that takes advantage of packets overheard by neighbors (the so-called broadcast

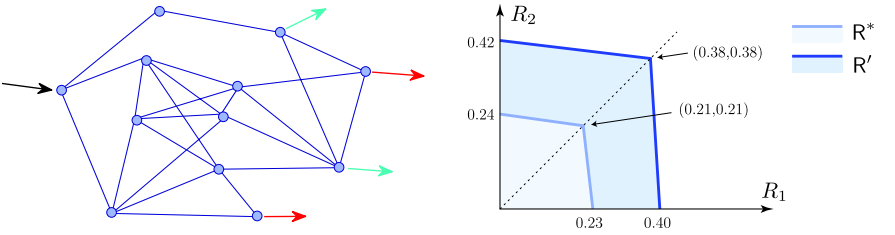

Fig. 7 Shown on *the left-hand side* is a wireless network with 12 nodes, one source, and two multicast groups, each consisting of two nodes. Shown on *the right-hand sid*e are two possible rate regions: the smaller region R^* is obtained using orthogonal scheduling, and the larger region is obtained using a more efficient coding scheme [41]

advantage) as well as of frequency reuse in the network due to efficient scheduling of non-conflicting transmissions. Moreover, if we have two different multicast connections then, while there is no competition for capacity among receivers within a multicast connection, there may be competition for resources among different multicast connections. Network coding may alleviate this competition for resources, as well as reduce the resources used for each multicast connection individually.

To consider the interaction among users, let $\mathcal{T}_1$ denote one group or receivers, and $\mathcal{T}_2$ another set of receivers. Consider, to begin with, orthogonal scheduling and the multicast connections $(s, \mathcal{T}_1, R_1^*)$, $(s, \mathcal{T}_2, R_2^*)$, and $(s, \mathcal{T}_1 \cup \mathcal{T}_2, R_0^*)$, where for each connection the rate is taken to be the maximal rate that can be supported by the network if only this particular connection is present. We have plotted the rate points $(R_1^*, 0) = (0.23, 0)$, $(0, R_2^*) = (0, 0.24)$, and $(R_0^*, R_0^*) = (0.21, 0.21)$ in Fig. 7. The extreme point (R_0^*, R_0^*) corresponds to transmission to each multicast group simultaneously at rate R_0^*. Interference among users is captured by the fact that R_0^* is in general strictly smaller than either R_1^* or R_2^*. Without coding we obtain the half-duplex constraint model in which the minimum cut from the source to the users in $\mathcal{T}_1 \cup \mathcal{T}_2$ is $\min(R_1^*, R_2^*)$.

The potential gain in capacity is illustrated in the plot on the right-hand side of Fig. 7. The rate region R^* is obtained using orthogonal coding, and the region R' is obtained using the coding approach of [41]. Observe that the rate regions are almost rectangular. This means that a 4-multicast with random terminals can support almost the same rate as a 2-multicast and indicates that network coded multicasting scales well with the number of receivers. However, the main message is that attention to coding can significantly expand the achievable rate region, as seen in this figure.

3 Relaxations and control

In this section we show how the concepts surveyed in the previous sections can be used to obtain control solutions for a stochastic network model, and for the physical network. We assume that rate regions have been determined through a combination of coding and local scheduling as discussed in Sect. 2. This is the basis of dynamic network models over which we design our policies.

We now introduce a stochastic model for the network, defined as a discrete time analog of the fluid model. Suppose $(\boldsymbol{A}, \boldsymbol{B})$ is an i.i.d. process with a finite second moment. For each t, the vector $A(t)$ has non-negative entries with mean α, the entries of $B(t)$ take on values in $\{-1, 0, 1\}$, and the mean of $B(t)$ is B. The queue-length process is denoted $\boldsymbol{Q}$, and evolves in discrete time through the recursion,

$$Q(t+1) = Q(t) + B(t+1)U(t) + A(t+1), \quad t \ge 0, \ Q(0) = x. \tag{14}$$

The input $U(t)$ is subject to the same constraints as $\zeta(t)$, which include the constraint that the resulting process $\boldsymbol{Q}$ remains in the positive orthant $\mathbb{R}^{\ell}_{+}$. We ignore lattice constraints on the queue-length process, so that $Q_i(t)$ is only restricted to be non-negative for each i and t. In all of the numerical experiments described here the entries of $A(t)$ and $-B(t)$ are taken to be Bernoulli, with means denoted $\mathsf{E}[A_i(t)] = \alpha_i$ and $\mathsf{E}[B_{ij}(t)] = -\mu_{ij}$. The general model was developed in [18, 34] and is now called the controlled random walk (CRW) network model.

Models of this form were introduced by Lippman in [28] for networks with Poisson interarrival times and service times—assumptions justified by the work of Erlang.

3.1 Policy translation

The approach to policy translation advocated in [29] is inspired by the dynamic programming equations associated with the fluid and stochastic models. In particular, suppose that the function J^* defined in (7) is a smooth function of x, and let ∇J^* denote its gradient. Then, the total cost optimality equation (TCOE) is given by

$$\min_{v \in \mathsf{V}} v^T \nabla J^*(x) = -c(x), \quad x \in \mathsf{X}. \tag{15}$$

Moreover, the minimizing v defines an optimal state feedback law $v^*(t) = \phi^*(q(t))$, with

$$\phi^*(x) := \arg\min_{v \in \mathsf{V}} v^T \nabla J^*(x), \quad x \in \mathsf{X}. \tag{16}$$

Similar equations define optimal policies for the CRW model with respect to a discount or average-cost optimality criteria [2, 34]. Generally, if $h\colon \mathbb{R}^{\ell} \to \mathbb{R}_{+}$ is a C^1 function then we define the *h-myopic policy* via

$$\phi(x) = \arg\min_{v \in \mathsf{V}} v^T \nabla h(x), \quad x \in \mathsf{X},$$

so that $\phi = \phi^*$ if $h = J^*$. It is assumed throughout that h is convex, monotone, and vanishes only at the origin. Under these assumptions it is known that the h-myopic policy is stabilizing for the fluid model when $\rho_\bullet < 1$ [34].

It would be tempting to attempt to apply an h-myopic policy to the CRW model, with h an approximation to the solution to an optimality equation for the CRW model, such as the fluid value function. However, this fails because the policy might not be feasible for the stochastic model. The problem is that in the fluid model it is possible to have service at an empty buffer, which is impossible in a discrete-time CRW model.

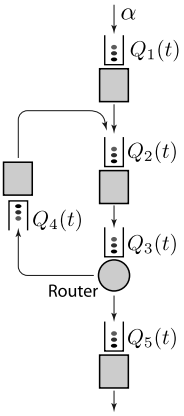

Fig. 8 Packets will be routed from buffer three to buffer four under the MaxWeight policy, for some values of $Q(t)$

However, the situation is not so dire—there are many examples of functions h for which the h-myopic policy is feasible for the CRW model.

The MaxWeight policy provides one example. If $h(x) = \frac{1}{2}\|x\|^2$, then the resulting h-myopic policy coincides with the MaxWeight policy for the models considered in [16, 37, 39, 40]. The MaxWeight policy has been popular for scheduling and routing in view of its stability properties. Recent work has developed decentralized implementations of these policies using consensus-type algorithms (see [35]) and distributed spanning tree constructions (see [15]).

Many generalizations have been proposed, including a more general quadratic $h(x) = \frac{1}{2}x^T Dx$ with $D > 0$ diagonal, or the generalization

$$h(x) = \sum_i d_i x_i^{1+\delta} \tag{17}$$

with $\delta > 0$ and $d_i > 0$ for each i. These generalizations and more recent refinements introduced in [6, 16, 42] are used to improve the performance of the policy with respect to delay. Results from experiments show that significant gains are possible by including additional global information, such as information regarding shortest paths to desired destinations.

Indeed, it has been observed that delay can be large when using the MaxWeight policy. An explanation was provided in [38] using the network model shown in Fig. 8 as an example. A centralized routing algorithm intended to minimize delay would never route any packets to buffer 4. Though stabilizing, the MaxWeight algorithm will route packets to buffer 4 for certain values of the queue-length vector. This will definitely increase the average delay.

To obtain a broader class of functions for which the h-myopic policy is feasible for the CRW model, and also stabilizing, consider again the function defined in (17). Under general conditions, the h-MaxWeight policy is feasible for the CRW model when $\delta > 0$, while feasibility typically fails when $\delta = 0$ (the case in which h is linear). The explanation given in [29] is that the function h satisfies the following boundary

conditions when δ is strictly positive:

$$\frac{\partial}{\partial x_i} h(x) = 0 \quad \text{whenever } x_i = 0. \tag{18}$$

Under these assumptions on h, including the boundary condition (18), the resulting h-myopic policy is called the h-MaxWeight policy in [29, 34]. This boundary condition is interpreted as zero 'marginal disutility' at an empty buffer, which ensures that there is a disincentive to work on an empty queue. This property is the key for stability because starvation of resources is avoided.

The condition (18) is easy to arrange. Suppose that $h_0 \colon \mathbb{R}^\ell \to \mathbb{R}_+$ is any function satisfying the assumptions imposed above: h_0 is convex, monotone, and vanishes only at the origin. We can then perturb this function to obtain a function satisfying (18), while maintaining the other desirable properties. One class of perturbations is of the form $h(x) = h_0(\tilde{x})$ where $\tilde{x} = (\tilde{x}_1, \ldots, \tilde{x}_\ell)^T \in \mathbb{R}^\ell_+$, and each $\tilde{x}_i(x)$ is convex, monotone, and vanishes only at the origin. Two examples are the exponential and logarithmic perturbations: For a given parameter $\theta > 0$ these are defined by, respectively,

$$\tilde{x}_i := x_i + \theta\big(e^{-x_i/\theta} - 1\big), \tag{19}$$

$$\tilde{x}_i := x_i \log(1 + x_i/\theta). \tag{20}$$

Feasibility of the h-MaxWeight policy requires some assumptions on the velocity set V. One set of sufficient conditions is given in Proposition 3.1.

Proposition 3.1 *Suppose that the following hold for the general fluid model*:

(i) $\rho_\bullet < 1$.

(ii) *For any* $v \in \mathsf{V}$ *and any* $i \in \{1, \ldots, \ell\}$, *if* $v_i < 0$ *then there is a vector* $v^+ \in \mathsf{V}$ *satisfying*

$$v_i^+ = 0 \quad \textit{and} \quad v_j^+ \le v_j \quad \textit{for } j \ne i.$$

Then, without loss of generality, the h*-myopic policy can be constructed so that for any* $x \in \mathbb{R}^\ell_+$, *and any* i, *we have* $\phi_i(x) \ge 0$ *when* $x_i = 0$.

Proof By definition we have

$$\phi(x) \in \arg\min_{v \in \mathsf{V}} \sum_{i=1}^{\ell} v_i \frac{\partial}{\partial x_i} h(x), \quad x \in \mathsf{X}.$$

If v^0 achieves the minimum, then there exists v^+ satisfying $v_i^+ = 0$ whenever $x_i = 0$ and $v_i < 0$, and $v_j^+ \le v_j$ otherwise. Monotonicity of h implies that ${v^0}^T \nabla h(x) \ge {v^+}^T \nabla h(x)$. □

To see how this applies to policy translation, we show how to translate an optimal policy for the fluid model relaxation to the CRW model. The translation is performed

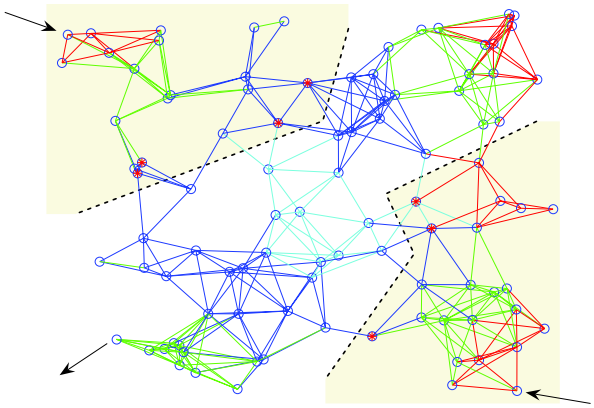

Fig. 9 A network with 100 nodes. There are four classes of links, differentiated by their respective capacities 0.5, 0.25, 0.2 and 0.15, and indicated in the figure by the four different styles shown. The two arrivals and one exit are indicated by *the arrows*

in two steps. In the first step we modify the value function for the relaxation. For a one-dimensional relaxation, and with linear cost, the value function $\hat{J}^*$ is a quadratic function of workload $w = \xi^T x$. To attempt to faithfully track the relaxation, we introduce a penalty term that introduces a large cost when $c(x) \gg \overline{c}(w)$:

$$h_0(x) := \hat{J}^*(w) + \frac{b}{2}\big(c(x) - \overline{c}(w)\big)^2, \quad w = \xi^T x,\ x \in \mathbb{R}^\ell_+, \tag{21}$$

where $b > 0$ is a constant. In the second step we perturb h_0 to obtain h satisfying (18).

It is shown in [29] that the resulting h-MaxWeight policy is approximately optimal under general conditions. While the arguments are general, this result is proven for a version of the scheduling model described below (4). Note however that to obtain such exact performance guarantees it is necessary to demand far greater information than when using the standard MaxWeight policy. In practice trade-offs must be made between information and performance. Moreover, network structure may change with time, in which case the policy must adapt to these changes.

3.2 h-MaxWeight for dynamic routing

In this final section we describe how these techniques apply to networks found in telecommunication applications.

Figure 9 shows a network with 100 nodes, two arrival streams, and one node from which the packets from these two sources exit the network. The network was constructed by first selecting at random the positions of nodes. A link between two nodes was created whenever the distance was less than a threshold. These links were chosen to be unidirectional, and each direction was also randomly selected. The capacity on a link between nodes i and j is denoted μ_{ij}. Its value was set to one of four possible values: 0.5, 0.25, 0.2 or 0.15. Generally, smaller rates were assigned for links in the central region of the network.

The two dashed lines shown in Fig. 9 represent a single cut between the two inflows, and the single outflow. The total number of packets in the shaded region coincide with the workload (in units of packets) corresponding to this cut.

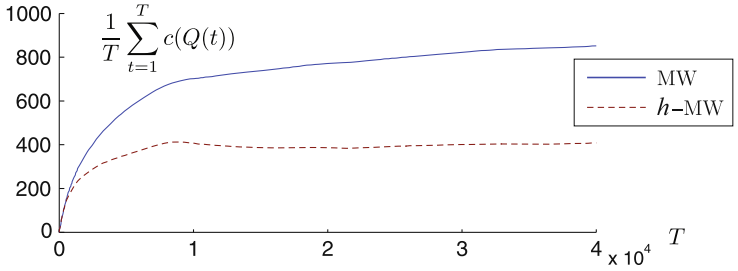

Fig. 10 Comparison of the average cost using the h-MaxWeight with "shortest path squared" component shown in (26) and the ordinary MaxWeight policy. The rates for the two arrivals are 0.283 and 0.33 respectively. The network load $\rho_\bullet$ is 0.95. In the simulation, n is set to be 1

In all of the numerical experiments described here, the arrival process $\boldsymbol{A} = \{A(t) : t \geq 1\}$ was taken to be an i.i.d. sequence with distribution,

$$P\big(A_i(t) = n\big) = \alpha_i/n, \qquad P\big(A_i(t) = 0\big) = 1 - \alpha_i/n, \quad 1 \leq i \leq 2, \tag{22}$$

where n is a constant used to capture variability. The mean and variance of $A_i(t)$ are given by, respectively,

$$m_{A_i(t)} = \alpha_i, \qquad \sigma^2_{A_i(t)} = n\alpha_i - \alpha_i^2. \tag{23}$$

Due to lack of space, the plots shown here are based on the model using $n = 1$. The entries of $-B(t)$ were taken to be Bernoulli with distribution

$$P\big(-B(t)_{ij} = 1\big) = \mu_{ij}, \qquad P\big(-B(t)_{ij} = 0\big) = 1 - \mu_{ij}, \quad 1 \leq i, j \leq \ell. \tag{24}$$

The queue-length process $\boldsymbol{Q}$ was constrained to evolve in the positive lattice $\mathbb{Z}_+^\ell$.

Our goal is to find a policy that approximately minimizes the average cost: the cost function is taken to be the total population, $c(x) = \sum_{i=1}^{\ell} x_i$. The average cost is simply the running average of $c(Q(t))$. Figure 10 compares the average cost using two policies: the h-MaxWeight policy for a particular function h, and the ordinary MaxWeight policy. The average cost is reduced by more than 50% with the h-MaxWeight policy in this experiment. The function h_0 is described in (26) that follows, and h was obtained via $h(x) = h_0(\tilde{x})$, with $\tilde{x}$ the exponential perturbation (19). Histograms of delay are shown in Fig. 11 for this example under various load conditions. Further details are provided at the end of this section.

We first consider the approach introduced in [29] using the function h_0 defined in (21),

$$h_0(x) := \hat{J}^*(w) + \frac{b}{2}\big(c(x) - \overline{c}(w)\big)^2, \quad w = \xi^T x,\ x \in \mathbb{R}_+^{100}, \tag{25}$$

where $\hat{J}^*$ is a quadratic function of workload, as in (11). With this version of the h-MaxWeight policy, the average workload is reduced significantly when compared to the ordinary MaxWeight policy. Results from one simulation are shown in Fig. 12. The explanation for this is that the h-MaxWeight policy results in far fewer packets

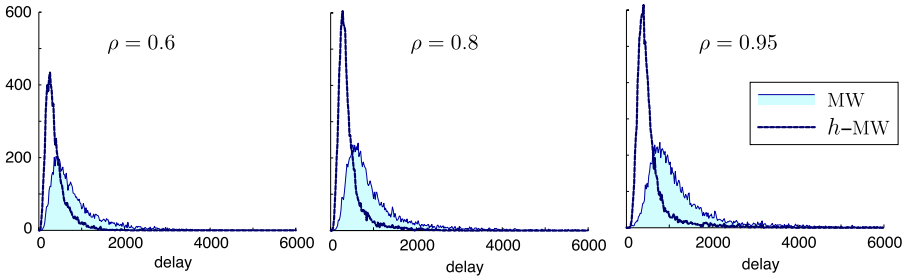

Fig. 11 Histogram of delays experienced by packets for different loads. The arrival rates for the load 0.6, 0.8, 0.95 are (0.179, 0.208), (0.238, 0.278) and (0.283, 0.33), respectively. The function h in the h-MW policy was taken to be the exponential perturbation of (26)

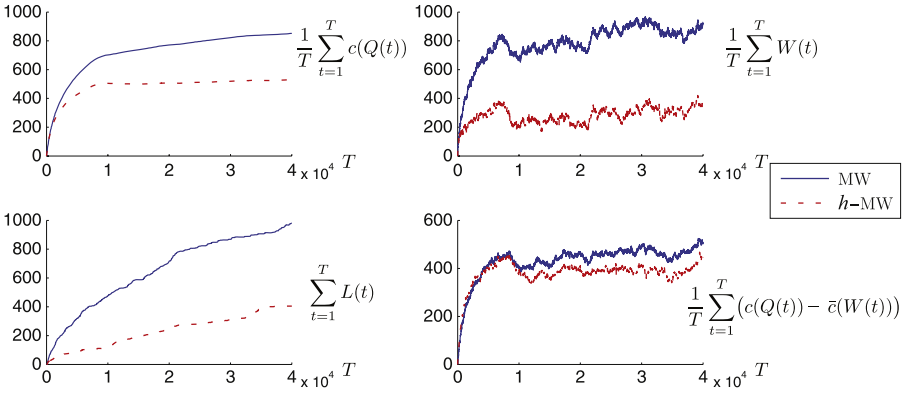

Fig. 12 Simulation results for the h-MaxWeight policy using the exponential perturbation of the function h_0 given in (25)

that double-back across the network cut. Denote by $L(t)$ the number of packets that cross the cut in the upstream direction at time $t \geq 0$. The cumulative sum of this quantity was obtained for the two policies, and the results are shown in Fig. 12. It is seen that the number of "loopy packets" is reduced by approximately one half when compared to the MaxWeight policy.

However, there is one aspect of this policy that is not satisfactory. Recall that an optimal solution for the workload relaxation requires $c(Q(t)) - \overline{c}(W(t)) \equiv 0$ for $t > 0$. The error $c(Q(t)) - \overline{c}(W(t))$ is proportional to the number of packets downstream of the network cut. Hence, in an optimal solution for the relaxation, all nodes downstream of the network cut are free of packets. The running average of $c(Q(t)) - \overline{c}(W(t))$ was obtained for the two policies, and the results are also displayed in Fig. 12. We see that the result is similar for either policy, and the average value is approximately half of the total average cost.

To better approximate the idealization $c(Q(t)) - \overline{c}(W(t)) \equiv 0$, we now introduce an additional penalty term in h_0 to more aggressively move traffic towards the exit node. The idea is to introduce information regarding the shortest path to the exit, following a similar modification of the MaxWeight policy introduced in [42]. For

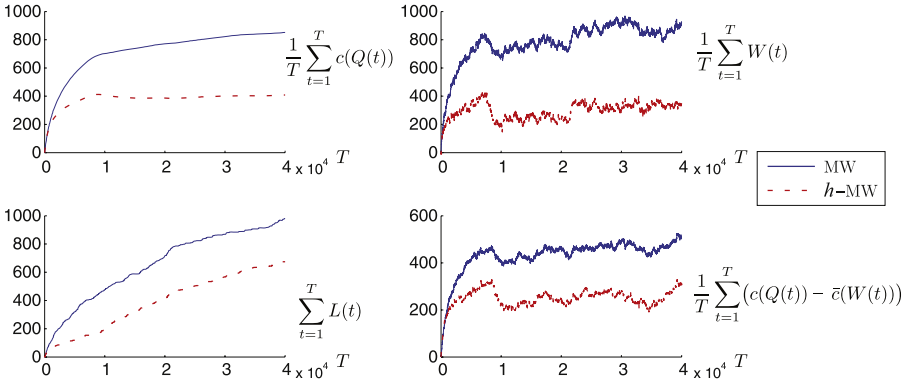

Fig. 13 Simulation results for the h-MaxWeight policy with shortest path component. *The solid line* shows result for the MaxWeight policy, while *the dashed line* shows the result for the h-MaxWeight with "shortest path squared" component shown in (26)

each i, denote by $s(i)$ the length (in hops) of the shortest path from node i to the exit node. For a given $p > 0$, the corresponding vector of powers of $\{s(i)\}$ is denoted by $d^{sp} = [s(1)^p, s(2)^p, \ldots, s(\ell)^p]^T$. The definition of h_0 in (21) is then modified to include the linear function of x obtained as the dot product with d^{sp}:

$$h_0(x) := \hat{J}^*(w) + \frac{b}{2}\big(c(x) - \overline{c}(w)\big)^2 + b_{sp} d^{sp\,T} x, \quad w = \xi^T x,\ x \in \mathbb{R}^{\ell}_{+}, \tag{26}$$

where b_{sp} is a constant.

Figure 13 shows results obtained under the same conditions as described following Fig. 10, but using this function in the definition of the h-MaxWeight policy, with $p = 2$ (so that $d_i^{sp} = s(i)^2$). The average value of $c(Q(t)) - \overline{c}(W(t))$ is reduced by nearly one half, as compared to the previous version of the h-MaxWeight policy.

In each simulation the delay for each packet entering the network was also recorded. This is the total time from arrival to the network, to the time of exit from the network. Histograms of delay are shown in Fig. 11. This includes the network under the same statistical setting as in the prior experiments, with network load $\rho_\bullet = 0.95$. Two other experiments were conducted with reduced arrival rates, resulting in loads $\rho_\bullet = 0.6$ and 0.8. The average delay, and the variability of delay are reduced dramatically using this policy, as compared to ordinary MaxWeight.

4 Conclusions

We have seen that a careful look at deterministic aspects of a communication network can provide insight regarding network control, as well as practical algorithms. As illustrated by examples, it is possible to develop models and solutions for broad classes of networks, and to take into account many network issues, including those related to multiple-access interference and coding.

There are many promising directions for future research; we highlight just two: the interplay between network control and information theory, and the decentralized implementation of network algorithms.

Information theory characterizes the fundamental gains and limits of coding, while network control is concerned primarily with policies and performance bounds. Information theory is an indispensable tool to guide the design of network algorithms; however, techniques from information theory have only recently begun to have impact on communication network design. This tension between the two disciplines (the *unconsummated union* [12]), is yet to be resolved in a satisfactory way. Working from both directions to create a more cohesive bridge is a very promising avenue of further research. The applications described here to multiple-access communication and network coded multicasting are just two examples.

The second direction concerns the decentralized implementation of network control algorithms. Centralized coordination is limited in most wireless networks owing to physical limitations and choice of architecture. There have been two recent approaches for designing decentralized network control algorithms. The first approach uses insights from game theory to design dynamic update mechanisms among users competing for network resources (see for example [7]). While being fully decentralized and flexible in modeling heterogeneous user metrics, this approach may lead to inefficiencies in the overall network performance due to strategic interactions among users. In environments where there are no strategic interactions, a more direct approach can be used that relies on optimization decomposition methods and consensus algorithms. Some of the techniques described in this paper, in particular network coding for multicasting, can be naturally decentralized using this approach. The construction of decentralized implementations of the h-MaxWeight policy is also possible using consensus algorithms, but the need for high reliability and efficiency will drive further research in this direction.

Acknowledgements This research was partially supported by NSF under grant ECS-0523620 and by the DARPA ITMANET program. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF or DARPA.

References

1. Ahlswede, R., Cai, N., Li, S.-Y.R., Yeung, R.W.: Network information flow. IEEE Trans. Inf. Theory **46**, 1024–1016 (2000)
2. Bertsekas, D.P.: Dynamic Programming and Optimal Control, 3rd edn. Athena Scientific, Nashua (2007)
3. Bertsimas, D., Niño-Mora, J.: Restless bandits, linear programming relaxations, and a primal-dual index heuristic. Oper. Res. **48**(1), 80–90 (2000)
4. Bramson, M., Williams, R.J.: Two workload properties for Brownian networks. Queueing Syst. Theory Appl. **45**(3), 191–221 (2003)
5. Brockmeyer, E., Halstrøm, H.L., Jensen, A.: The Life and Works of A.K. Erlang. The Copenhagen Telephone Company, Copenhagen (1948)
6. Bui, L., Srikant, R., Stolyar, A.: Novel architectures and algorithms for delay reduction in back-pressure scheduling and routing. arXiv:0901.1312. A short version of this paper is accepted to the INFOCOM 2009 Mini-Conference, Jan 2009
7. Candogan, U.O., Menache, I., Ozdaglar, A., Parrilo, P.A.: Competitive scheduling in wireless collision channels with correlated channel state. In: Proceedings of the International Conference on Game Theory for Networks (GameNets), 13–15 May 2009

8. Chen, W., Huang, D., Kulkarni, A., Unnikrishnan, J., Zhu, Q., Mehta, P., Meyn, S., Wierman, A.: Approximate dynamic programming using fluid and diffusion approximations with applications to power management. In: Proceedings of the 48th IEEE Conference on Decision and Control, 16–18 December 2009 (to appear)
9. Coffman, E.G. Jr., Mitrani, I.: A characterization of waiting times realizable by single server queues. Oper. Res. **28**, 810–821 (1980)
10. Cover, T.M.: Comments on broadcast channels. IEEE Trans. Inf. Theory **44**(6), 2524–2530 (1998)
11. Cover, T.M., Thomas, J.A.: Elements of Information Theory. Wiley, New York (1991)
12. Ephremides, A., Hajek, B.E.: Information theory and communication networks: an unconsummated union. IEEE Trans. Inf. Theory **44**(6), 2416–2434 (1998)
13. Erlang, A.K.: Solution of some problems in the theory of probabilities of significance in automatic telephone exchanges. In: Brockmeyer, E., Halstrøm, H.L., Jensen, A. (eds.) The Life and Works of A.K. Erlang, p. 189. The Copenhagen Telephone Company, Copenhagen (1948). Originally published in Danish in Elektrotkeknikeren, vol. 13 (1917)
14. Erlang, A.K.: The theory of probabilities and telephone conversations. In: Brockmeyer, E., Halstrøm, H.L., Jensen, A. (eds.) The Life and Works of A.K. Erlang, p. 131. The Copenhagen Telephone Company, Copenhagen (1948). Originally published in Danish in Nyt Tidsskrift for Matematik B (1909)
15. Eryilmaz, A., Ozdaglar, A., Modiano, E.: Polynomial complexity algorithms for full utilization of multihop wireless networks. In: Proceedings of IEEE INFOCOM (2007)
16. Georgiadis, L., Neely, M., Tassiulas, L.: Resource Allocation and Cross Layer Control in Wireless Networks. Foundations and Trends in Networking, vol. 1(1). Now Publishers, Hanover (2006)
17. Gupta, G.R., Shroff, N.B.: Delay analysis for multi-hop wireless networks. In: Proceedings of IEEE Infocom, Rio de Janeiro, Brazil (2009). Presentation given at ITA-Workshop 2009, UCSD
18. Henderson, S.G., Meyn, S.P., Tadić, V.B.: Performance evaluation and policy selection in multiclass networks. Discrete Event Dyn. Syst. Theory Appl. **13**(1–2), 149–189 (2003). Special issue on learning, optimization and decision making (invited)
19. Ho, T., Médard, M., Effros, M., Karger, D.: On randomized network coding. In: Proc. 41st Allerton Annual Conference on Communication, Control and Computing, October 2003
20. Ho, T., Medard, M., Koetter, R., Karger, D.R., Effros, M., Shi, J., Leong, B.: A random linear network coding approach to multicast. IEEE Trans. Inf. Theory **52**(10), 4413–4430 (2006)
21. Jaggi, S., Sanders, P., Chou, P.A., Effros, M., Egner, S., Jain, K., Tolhuizen, L.M.G.M.: Polynomial time algorithms for multicast network code construction. IEEE Trans. Inf. Theory **51**(6), 1973–1982 (2005)
22. Johannsen, F.W.: Waiting times and number of calls. P.O. Electr. Eng. J. (1907)
23. Kelly, F.P., Laws, C.N.: Dynamic routing in open queueing networks: Brownian models, cut constraints and resource pooling. Queueing Syst. Theory Appl. **13**, 47–86 (1993)
24. Kleinrock, L.: Queueing Systems. Vol. 1: Theory. Wiley, New York (1975)
25. Kumar, S., Kumar, P.R.: Performance bounds for queueing networks and scheduling policies. IEEE Trans. Automat. Contr. **AC-39**, 1600–1611 (1994)
26. Kumar, P.R., Meyn, S.P.: Duality and linear programs for stability and performance analysis queueing networks and scheduling policies. IEEE Trans. Automat. Contr. **41**(1), 4–17 (1996)
27. Laws, N.: Dynamic routing in queueing networks. Ph.D. Thesis, Cambridge University, Cambridge, UK (1990)
28. Lippman, S.: Applying a new device in the optimization of exponential queueing systems. Oper. Res. **23**, 687–710 (1975)
29. Meyn, S.: Stability and asymptotic optimality of generalized MaxWeight policies. SIAM J. Control Optim. **47**(6), 3259–3294 (2009)
30. Meyn, S.P.: The policy iteration algorithm for average reward Markov decision processes with general state space. IEEE Trans. Automat. Contr. **42**(12), 1663–1680 (1997)
31. Meyn, S.P.: Stability and optimization of queueing networks and their fluid models. In: Mathematics of Stochastic Manufacturing Systems, Williamsburg, VA, 1996, pp. 175–199. Am. Math. Soc., Providence (1997)
32. Meyn, S.P.: Sequencing and routing in multiclass queueing networks. Part II: Workload relaxations. SIAM J. Control Optim. **42**(1), 178–217 (2003)
33. Meyn, S.P.: Dynamic safety-stocks for asymptotic optimality in stochastic networks. Queueing Syst. Theory Appl. **50**, 255–297 (2005)
34. Meyn, S.P.: Control Techniques for Complex Networks. Cambridge University Press, Cambridge (2007)

35. Modiano, E., Shah, D., Zussman, G.: Maximizing throughput in wireless networks via gossiping. In: Proceedings of ACM Sigmetrics/IFIP Performance (2006)
36. Shah, D., Wischik, D.: Lower bound and optimality in switched networks. In: Proceedings of the 46th Annual Allerton Conference on Communication, Control, and Computing, pp. 1262–1269, Sept. 2008
37. Srikant, R.: The Mathematics of Internet Congestion Control. Systems & Control: Foundations & Applications. Birkhäuser Boston, Boston (2004)
38. Subramanian, V., Leith, D.: Draining time based scheduling algorithm. In: Proceedings of the 46th IEEE Conf. on Decision and Control, pp. 1162–1167 (2007)
39. Tassiulas, L., Ephremides, A.: Jointly optimal routing and scheduling in packet radio networks. IEEE Trans. Inf. Theory **38**(1), 165–168 (1992)
40. Tassiulas, L., Ephremides, A.: Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. IEEE Trans. Automat. Contr. **37**(12), 1936–1948 (1992)
41. Traskov, D., Heindlmaier, M., Medard, M., Koetter, R., Lun, D.S.: Scheduling for network coded multicast: a conflict graph formulation. In: Proceedings of the IEEE GLOBECOM Workshop, pp. 1–5 (2008)
42. Ying, L., Shakkottai, S., Reddy, A.: On combining shortest-path and back-pressure routing over multihop wireless networks. In: Proceedings of IEEE Infocom, Rio de Janeiro, Brazil, April 2009