Shannon meets Bellman: Feature based Markovian models for detection and optimization

Sean Meyn and George Mathew

***Abstract*— The goal of this paper is to develop modeling techniques for complex systems for the purposes of control, estimation, and inference:**

(i) A new class of hidden Markov models is introduced, called the optimal feature prediction (OFP) model. It is similar to the Gaussian mixture model in which the actual marginal distribution is used in place of a Gaussian distribution. This structure leads to simple learning algorithms to find an optimal model.

(ii) The OFP model provides a unification of other modeling approaches including the projective methods of Shannon, Mori and Zwanzig, and Chorin, as well as a version of the binning technique for Markov model reduction.

(iii) Several general applications are surveyed, including inference and optimal control. Computation of the spectrum, or solutions to dynamic programming equations are possible through a finite dimensional matrix calculation without knowledge of the underlying marginal distribution on which the model is based.

I. Introduction

These words are often attributed to Mark Shaney: *Oh yes, I was looking for. I'm so glad I remembered it. Yeah, what I have wondered if I had committed a crime. Don't eat with your assessment of Reagon and Mondale...* Mr. Shaney is in fact a fictional person. The true architect is Don P. Mitchell[1], but the inspiration comes from Claude Shannon.

Shannon introduced the idea of low dimensional Markov models to replicate features of English language. This appears as motivation for the notion of entropy in his famous 1948 paper *A mathematical theory of communication*, which is regarded as the birth of modern information theory [25].

There have been similar independent efforts in the physics community by Mori [22] and Zwanzig [29] to derive reduced order models to describe complex systems. In the Mori-Zwanzig formalism, a low-dimensional model for phase variables (what we call "features") is given by a generalized Langevin equation that has a Markovian element, a non-Markovian "memory" element and a random element. The so-called (first-order) *optimal prediction model* developed more recently by Chorin and co-workers [6], [7] coincides with the Mori-Zwanzig formalism when the non-Markovian dynamics are removed.

Financial support from the National Science Foundation (ECS-0523620) and DARPA DynaRUM is gratefully acknowledged.

S.M. is a professor at the Department of Electrical and Computer Engineering, and a Research Professor in the Coordinated Science Laboratory at the University of Illinois.

G.M. is a Postdoctoral fellow with the Department of Mechanical Engineering, University of California - Santa Barbara.

[1]Dewdney, A.K. (June 1989). "Computer Recreations". Scientific American 260 (6): p122-125

The optimal prediction model is described in Proposition 1.1. From the construction it can be seen that this is precisely the same as Shannon's Markovian model first introduced in [25].

Proposition 1.1: Suppose that $\boldsymbol{Z}$ is a stationary process on Z, let μ denote its marginal distribution, and let μ^2 denote the bivariate distribution,

$$\mu^2(dz_0, dz_1) = \mathsf{P}\{Z(0) \in dz_0,\ Z(1) \in dz_1\}.$$

Suppose that Radon-Nikodym derivative,

$$T(z_0, A) = \frac{\mu^2(dz_0, A)}{\mu(dz_0)}, \qquad z_0 \in \mathsf{Z}, \tag{1}$$

exists for each $z_0 \in \mathsf{Z}$ and $A \in \mathcal{B}(\mathsf{Z})$ (the Borel sigma field on Z). Assume moreover that $T(\,\cdot\,, A)$ is measurable for each A, $T(z_0, \cdot\,)$ is a probability measure on $\mathcal{B}(\mathsf{Z})$ for each z_0. Then T defines a transition kernel on $\mathsf{Z}\times\mathcal{B}(\mathsf{Z})$ with invariant measure μ. □

In this paper we survey a range of new applications and new formulations of feature-based Markovian models. Of particular interest in current research are applications to spectral theory, hypothesis testing, and to machine learning. In this paper emphasis is focused on applications to machine learning.

The paper is organized as follows. In the following section we introduce the optimal feature prediction (OFP) model. This is a generalization of the optimal prediction model designed to combine the statistical flexibility of Shannon's model with the computational features of finite state space hidden Markov models. Among the most compelling applications of this technique is to decentralized control of complex networked systems. Optimal solutions are in general intractable since even a Markov model with finite state space gives rise to an infinite dimensional Markov decision process [9], [4], [3]. In Section III we illustrate with a single network example the application of Markovian modeling to decentralized control. Section IV contains conclusions and some unanswered questions.

II. Markov modeling

Proposition 1.1 is a trivial consequence of the definitions, yet its implications are surprisingly rich. A roadblock to its application is that the transition kernel T is not known. Moreover, in general it remains an infinite dimensional object, in which case learning the entire transition kernel is not feasible.

Here we introduce a simplified class of models for which learning the transition kernel amounts to a finite-dimensional

optimization problem. The model class retains the important feature of the optimal-prediciton model that certain steady-state statistics are captured exactly.

To define the OFP model for a stationary process $\boldsymbol{Z}$ we begin with the following structural assumption. Let μ denote the marginal distribution of $\boldsymbol{Z}$, and μ^2 the bivariate distribution defined in Proposition 1.1. Hence μ is a probability measure on $\mathcal{B}(\mathsf{Z})$, and μ^2 is a probability measure on $\mathcal{B}(\mathsf{Z}^2)$. It is assumed throughout the paper that μ^2 possesses a density with respect to the product distribution,

$$\mu^2(dz_0, dz_1) = p(z_0, z_1)\mu(dz_0)\mu(dz_1) \tag{2}$$

where $p\colon \mathsf{Z}^2 \to \mathbb{R}_+$ is measurable. We also use the more compact form $\mu^2 = p\mu \otimes \mu$ where for two probability measures on $\mathcal{B}(\mathsf{Z})$ and two functions s, r on X the outer products are defined by,

$$\mu\otimes\mu\,(dz_0, dz_1){:=}\mu(dz_0)\mu(dz_1),\;\; s\otimes r\,(z_0, z_1){:=}s(z_0)r(z_1).$$

The existence of a density in (2) is guaranteed when the state space Z is countable. An example of a model for which this fails is the Markov process defined by the n-dimensional Ornstein Uhlenbeck process $X(t{+}1) = AX(t){+}BW(t{+}1)$. Suppose that $\boldsymbol{W}$ is i.i.d. $N(0, I)$ and (A, B) is controllable. In this case μ is a full-rank Gaussian distribution, and hence possesses a density with respect to Lebesgue measure. The condition (2) holds with $\boldsymbol{Z} = \boldsymbol{X}$ if and only if the matrix B has rank n. However, if this rank condition is relaxed then (2) does hold with $Z(t) := X(nt)$, $t \geq 0$ (see [21], where these results are a consequence of the irreducibility structure of the linear model).

The OFP model is obtained using an approximation to the density p. Let $\{r_i : 1 \leq i \leq N\}$ denote measurable, real-valued functions on Z, and define for given parameters $\{\Theta_{ij} : 1 \leq i, j \leq N\}$,

$$\mu^2_\Theta(dz_0, dz_1) := \sum_{i,j=1}^{N} \theta_{i,j} r_i(z_0) r_j(z_1)\mu(dz_0)\mu(dz_1). \tag{3}$$

The transition kernel T_Θ is then defined using (1):

$$T_\Theta(z, A) = \frac{\mu^2_\Theta(dz, A)}{\mu_{\Theta\,1}(dz)}, \qquad z \in \mathsf{Z},\;\; A \in \mathcal{B}(\mathsf{Z}).$$

where $\mu_{\Theta\,1}(dz_0) := \mu^2_\Theta(dz_0, \mathsf{Z})$ is the first marginal.

The choice of basis $\{r_i : 1 \leq i \leq N\}$ and the parameter Θ will depend on which features we wish to capture in the Markov model. We will see that the steady-state first and second order statistics of an appropriate function class can be captured precisely in the finite rank model.

We first explain how this model class is related to hidden Markov models.

A. Finite rank models and HMMs

The transition kernel has finite rank, in the sense that there are functions $\{s_i\}$ and probability measures $\{\mu_i\}$ satisfying,

$$T_\Theta(z, A) = \sum_{i=1}^{N} s_i(z)\mu_i(A), \qquad z \in \mathsf{Z},\;\; A \in \mathcal{B}(\mathsf{Z}). \tag{4}$$

Some properties of finite-rank transition laws are summarized in the following:

Proposition 2.1: The Markov chain $\widehat{\boldsymbol{Z}}$ with finite-rank transition kernel (4) has the following properties:

(i) $\widehat{\boldsymbol{Z}}$ is, of course, a Markov chain on the state space Z.

(ii) The N-dimensional stochastic process defined by $(s_1(\widehat{Z}(t)), \ldots, s_N(\widehat{Z}(t)))^{\intercal}$, $t \geq 0$, is a Markov chain on $\mathbb{R}^N$.

(iii) $\widehat{\boldsymbol{Z}}$ is also a hidden Markov model: There is a finite state space Markov chain $\boldsymbol{I}$ on the finite set $\{1, \ldots, N\}$, an i.i.d. process $\boldsymbol{W}$ on $\mathbb{R}$, and a function $\varphi\colon \{1, \ldots, N\} \times \mathbb{R} \to \mathsf{Z}$ such that,

$$\widehat{Z}(t+1) = \varphi(I(t), W(t+1)), \qquad t \geq 0.$$

□

The parameter Θ is chosen so that $\mu^2_\Theta \sim \mu^2$. In the following subsections we introduce several optimization criteria and describe their properties. Section II-E briefly describes how an optimal parameter can be computed using Monte-Carlo techniques.

The simplicity of computation of an optimal parameter is remarkable, given the difficulties associated with model construction for general HMMs. One reason for the simplicity is that there is less to be learned: Never do we attempt to estimate μ. We shall see that in many applications this full information is not needed. For example, solutions to dynamic programming equations can be obtained using finite-dimensional statistics.

B. L_2 optimal model

The L_2-mismatch-criterion is defined for any Θ by,

$$\mathcal{E}(\Theta) = \tfrac{1}{2}\int (p_\Theta(z_0, z_1) - p(z_0, z_1))^2 \mu(dz_0)\mu(dz_1) \tag{5}$$

Computing the gradient of $\mathcal{E}$ with respect to Θ and setting this equal to zero gives the minimizer. The form of the solution and other conclusions are summarized in the following:

Proposition 2.2: Suppose that $\{r_i\}$ are linearly independent in $L_2(\mu)$. Then, the vector Θ^* minimizes $\mathcal{E}$ if and only if the optimal-prediction constraints hold for each i and j:

$$\mathsf{E}_{\Theta^*}[r_i(\widehat{Z}(t))r_j(\widehat{Z}(t+1))] = \mathsf{E}[r_i(Z(t))r_j(Z(t+1))], \tag{6}$$

where the expectations are in steady-state. The unique solution is expressed,

$$\Theta^* = \{\Theta^*_{ij}\} = [R^r(0)]^{-1} R^r(1) [R^r(0)]^{-1} \tag{7}$$

where $R^r_{i,j}(0) = \mu(r_i r_j)$ and $R^r_{i,j}(1) = \mu^2(r_i \otimes r_j)$.

C. Positivity and optimal prediction

What is left out in Proposition 2.2 is the constraint that a transition kernel must be non-negative valued, with $T_\Theta(z, \mathsf{Z}) \equiv 1$. The latter constraint is automatic under the assumptions of Proposition 2.2 provided the constant function 1 lies in the span of the $\{r_i\}$. One approach to

guarantee non-negativity is through barrier function methods. For example, define the augmented cost function,

$$\mathcal{E}_B(\Theta,\varepsilon) = \mathcal{E}(\Theta) + \varepsilon \int \log\big(p_\Theta(z_0,z_1)\big)\mu(dz_0)\mu(dz_1)$$

Minimization of $\mathcal{E}$ can be cast as a convex program.

Alternatively, a parameter can be chosen by directly imposing the optimal prediction property on a subspace: For a given collection of functions $\{\phi_i\}$ in $L_2(\mu)$ we can choose $\Theta = \Theta^*$ to guarantee, for each i, j,

$$\mathsf{E}_{\Theta^*}[\phi_i(\widehat{Z}(t))\phi_j(\widehat{Z}(t+1))] = \mathsf{E}[\phi_i(Z(t))\phi_j(Z(t+1))] \quad (8)$$

This may not be possible while still respecting positivity, but we can construct a convex cost function to capture an approximate fit subject to positivity.

D. Relative entropy metric

The long-run entropy rate $n^{-1}D(\mu^n\|\mu^n_\Theta)$ converges under general conditions to the following function of Θ:

$$\mathcal{E}_D(\Theta) = \langle \mu^2, \log(p^2_\Theta) - \log(p_{\Theta 1})\rangle + b(\mu^2) \qquad (9)$$

where the inner-product notation denotes integration, and $b(\mu^2)$ is independent of Θ. This is known as the Donsker-Varadhan rate function that appears in the generalization of Sanov's Theorem for Markov chains [8], [14].

The rate function is known to be convex, and hence minimizing $\mathcal{E}_D$ can be cast as a convex optimization problem. This can be refined to include optimal prediction constraints of the form (8).

E. On-line computation

Computation of Θ^* based on the form given in Proposition 2.2 is possible by naive Monte-Carlo given observations of $\boldsymbol{Z}$ in steady-state.

For either the convex, non-quadratic cost functions $\mathcal{E}_B(\Theta,\varepsilon)$ or $\mathcal{E}_D(\Theta)$ the gradient and Hessian have simple forms that facilitate the application of stochastic gradient or stochastic Newton-Rapshon techniques that are convergent to the unique optimizer.

Note that the Baum-Welch and EM algorithms are designed to achieve the same computational goals for HMMs. These algorithms are only known to converge to a local optimum in general.

III. Optimization

Optimal prediction models have clear applications to policy improvement or approximate optimization in controlled stochastic systems. In this section we describe the extension to controlled Markov models (or MDPs).

The basis approach utilized in the construction of the OFP MDP model is similar in spirit to the use of bases to approximate value functions or policies in machine learning [26], [2], [28], [23], [18]. The contribution of this paper is the new class of models, as well as novel application. In particular, in Section III-D we show how these ideas can be used to construct decentralized policies based on local, distributed MDP models.

We begin with Markov model construction for non-Markovian models.

A. MDP models

Suppose that $(\boldsymbol{Z},\boldsymbol{U})$ are a state and control process. It is assumed that $\boldsymbol{U}$ is defined by a stationary, perhaps randomized policy defined for some feedback law ϕ via,

$$\begin{aligned}\mathsf{P}\{U(t)=u \mid Z^t_{-\infty}; Z(t)=z\} &= \mathsf{P}\{U(t)=u \mid Z(t)=z\}\\ &= \phi(u\mid z)\end{aligned}$$

We can construct a Markov model for $(\boldsymbol{Z},\boldsymbol{U})$ using Proposition 1.1. This defines the transition law T on $(\mathsf{Z}\times\mathsf{U})\times(\mathsf{Z}\times\mathsf{U})$ by,

$$T\big(\tbinom{z_0}{u_0},\tbinom{z_1}{u_1}\big) = \mathsf{P}\big\{\tbinom{Z(t+1)}{U(t+1)} = \tbinom{z_1}{u_1} \mid \tbinom{Z(t)}{U(t)} = \tbinom{z_0}{u_0}\big\} \quad (10)$$

where the expectation is in steady-state. A controlled transition law is then defined for each triple u, z_0, z_1 by,

$$T_u(z_0,z_1) = \textstyle\sum_{u_1} T\big(\tbinom{z_0}{u},\tbinom{z_1}{u_1}\big) \qquad (11)$$

Alternatively, suppose that we are given a basis $\{r_i : 1 \le i \le N\}$ of functions on $\mathsf{Z}\times\mathsf{U}$. We can then construct an approximation to the bivariate distribution of $\big(\tbinom{Z(t)}{U(t)},\tbinom{Z(t+1)}{U(t+1)}\big)$ in steady state, and then a transition law of the form,

$$\begin{aligned}&T\big(\tbinom{z_0}{u_0},\tbinom{z_1}{u_1}\big)\\ &\qquad = \sum_{i,j=1}^N \Theta^*_{ij} s_i(z_0,u_0) r_j(z_1,u_1)\mu(z_1)\phi(u_1\mid z_1)\end{aligned}$$

with Θ^* obtained using (7) based on the joint process $(\boldsymbol{Z},\boldsymbol{U})$. With (10) obtained using a basis, the MDP model is again obtained using (11). The resulting controlled transition law is expressed,

$$T_u(z_0,z_1) = \textstyle\sum_{i,j=1}^N \Theta^*_{ij} s_i(z_0,u_0) r_j(z_1)\mu(z_1) \qquad (12)$$

where with a slight abuse of notation we define $r_j(z_1) = \sum_u r_j(z_1,u)\phi(u\mid z_1)$, and μ is the marginal distribution of $\boldsymbol{Z}$.

B. Q Learning

Given a Markov model, of the form (11) or (12), and given a cost function $c\colon \mathsf{Z}\times\mathsf{U}\to\mathbb{R}_+$, the average-cost dynamic programming equation is expressed,

$$\min_u \big\{c(z,u) + T_u h^*(z)\big\} = h^*(z) + \eta_*$$

where η_* is the optimal average cost, typically independent of the initial condition z, and $h^*\colon \mathsf{Z}\to\mathbb{R}$ is the *relative value function*. Watkin's approach is based on the substitution of h^* by the so-called "Q-values",

$$H^*(z,u) = c(z,u) + T_u h^*(z), \qquad z\in\mathsf{Z},\ u\in\mathsf{U}.$$

Letting $\underline{H}^*(z) = \min_u H^*(z,u)$, we find that H^* satisfies the fixed point equation,

$$H^*(z,u) + \eta_* = c(z,u) + T_u\underline{H}^*(z), \qquad z\in\mathsf{Z},\ u\in\mathsf{U} \quad (13)$$

If $\boldsymbol{U}$ is defined by a randomized policy that assigns positive probability to each feasible (z,u), then we can estimate H^* using a simple Monte-Carlo recursion. Stability of the

algorithm is simplified using the ODE method for stability of stochastic approximation introduced in [5].

Estimation of H^* is facilitated when the MDP model is of finite rank, of the form (12). In this case we conclude from (13) that, for some vector α^*,

$$H^*(z,u)+\eta_*-c(z,u) = \sum \alpha_i^* s_i(z,u), \qquad (z,u) \in \mathsf{Z}\times\mathsf{U}.$$

Computation of α^* is straightforward. We thus arrive at an alternative approach to finite-dimensionally parameterized Q-learning. The only other such approach, introduced recently in [18], is known to be convergent only under strong conditions on the basis that parameterizes H^* [18], [17].

Once H^* is obtained, the optimal policy for the MDP model is given by the minimizer,

$$\phi^*(z) = \arg\min_u H^*(z,u), \qquad z \in \mathsf{Z}.$$

Conditions on T_u and on $(\boldsymbol{Z},\boldsymbol{U})$ will be required to ensure that the resulting policy will be optimal, or even stabilizing for $\boldsymbol{Z}$ if this process is not Markovian.

C. Sensitivity

Schweitzer's approach for sensitivity analysis in Markov models [24] can be extended to optimal-prediction models for non-Markovian processes. Suppose we have a family of processes $\boldsymbol{Z}^\alpha$, indexed by a parameter α that lies in a convex set. For simplicity it is assumed that the parameter is scalar, and that the common state space Z is finite. Let T_α denote the optimal prediction model, and let μ_α denote the invariant measure for T_α, interpreted as a row vector. Assume that a cost function c on Z is given, and let $\eta_\alpha = \sum c(z)\mu_\alpha(z)$ denote the steady-state cost.

Let $1\otimes\mu_\alpha$ denote the rank-one matrix with all rows equal to μ_α. The inverse $U_\alpha = (I - T_\alpha + 1\otimes\mu_\alpha)^{-1}$ is known as the *fundamental matrix* [21]. The function $\hat{c}_\alpha := U_\alpha c$ solves Poisson's equation, $T_\alpha\hat{c}_\alpha = \hat{c}_\alpha - c + \eta_\alpha$.

The formula (14) is well-known for Markov chains [24]. We believe that this formula can be used to construct algorithms for policy improvement based on steepest descent, following [12], [13].

Proposition 3.1: Suppose that T_α is irreducible and that the derivative with respect to α exists for each α. Then the marginal μ_α is differentiable, and its derivative is given by

$$\mu'_\alpha = \mu_\alpha T'_\alpha U_\alpha \tag{14}$$

In particular, for any function c, the sensitivity of the mean is given by

$$\eta'_\alpha = \sum_{z_0,z_1} \mu_\alpha(z_0)T'_\alpha(z_0,z_1)\hat{c}_\alpha(z_1)$$

□

D. Local Markov Models and Distributed Control

We now show how the OFP MDP model can be used to construct decentralized control laws in complex systems. For simplicity we restrict to a special case consisting of two locations, each subject to local control.

Suppose that $\boldsymbol{X}$ is a state process that can be decomposed as a pair of processes $X(t) = (X^1(t), X^2(t))$, $t \ge 0$. In the numerical experiments described below the process $\boldsymbol{X}$ is assumed to be defined by an MDP model, but this is not necessary. We view $\boldsymbol{Z} = \boldsymbol{X}^1$ as the feature variable: It together with its local control process $\boldsymbol{U}$ will be used to construct a local OFP MDP model. A *second model* is obtained based on $\boldsymbol{X}^2$.

Suppose that a cost function has been defined with respect to the full state $\boldsymbol{X}$. To complete the construction of an MDP model for $\boldsymbol{Z}$ it is necessary to define a cost function on this feature variable. For a fixed policy, consider the following,

$$\overline{c}(z,u) := \mathsf{E}[c(X(t)) \mid Z(t) = z,\ U(t) = u] \tag{15}$$

where the conditional expectation is taken in steady-state. For the optimal prediction model this is consistent: By the smoothing property of the conditional expectation,

$$\mathsf{E}[\overline{c}(Z(t),U(t)) = \mathsf{E}[c(X(t))] \tag{16}$$

In this way we have projected the cost onto the local process $(\boldsymbol{Z},\boldsymbol{U})$.

Standard MDP methodology suggests several approaches to policy improvement by adapting value iteration or policy iteration [11], [1], [10], [26], [2]. A direct approach based on policy iteration is described as follows: We begin with a randomized policy $\phi^{(0)}$ that assigns positive probability to any feasible control value — The motivation for this initialization is exactly as for simulated annealing or actor-critic methods [26], [2]. Based on this initial condition we generate a sequence Markov models $\{T^{(n)} : n \ge 0\}$, and a sequence of policies $\{\phi^{(n)} : n \ge 0\}$ inductively: For $n \ge 0$, given the nth policy,

(i) Obtain the optimal-prediction MDP model $T_u^{(n)}$

(ii) Obtain the conditional cost (15),

$$\overline{c}^{(n)}(z,u) := \mathsf{E}^{\phi^{(n)}}[c(X(t)) \mid Z(t) = z,\ U(t) = u]$$

The right hand side is the expectation in steady-state, based on the policy $\phi^{(n)}$.

(iii) Obtain an optimal policy $\phi^{(n*)}$ for the MDP model with controlled transition law $T^{(n)}$ and cost function $\overline{c}^{(n)}$.

(iv) Define a new randomized policy via,

$$\phi^{(n+1)} = \phi^{(n)} + \gamma_n(\phi^{(n*)} - \phi^{(n)}) \tag{17}$$

where $\{\gamma_n\} \subset (0,1)$ is a non-negative gain sequence.

The reason for using $\phi^{(n+1)}$ and not $\phi^{(n*)}$ in (iv) is that the latter is typically a deterministic policy — For each state $Z(t) = z$, there is a unique optimal control value $U(t) = u^*$ defined by $\phi^{(n*)}(z)$. A deterministic policy is undesirable since some state-control pairs are never visited, and hence learning is inhibited.

The application of a basis can be used to streamline estimation of the projected cost (15).

Let $\{\psi_i^c : 1 \le i \le \ell_c\}$ denote a collection of functions of (z,u), and for $\beta \in \mathbb{R}^{\ell_c}$ define,

$$\overline{c}_\beta(z,u) = \sum \beta_i\psi_i^c(z,u), \qquad z \in \mathsf{Z},\ u \in \mathsf{U}.$$

To approximate the projected cost, recall that the conditional expectation in (15) is nothing but a *projection*: The L_2 projection of the random variable $c(X(t))$ onto the subspace of all random variables that can be expressed as a function of the pair $(Z(t), U(t))$. Consequently, the conditional expectation satisfies

$$\mathsf{E}\big[\big(c(X(t)) - \overline{c}(Z(t),U(t))\big)g(Z(t),U(t))\big] = 0$$

for every function g for which $\mathsf{E}[(g(Z(t),U(t))^2] < \infty$. We relax this requirement, and instead project onto the finite-dimensional space of random variables spanned by $\{\psi_i^c(Z(t),U(t)) : 1 \le i \le \ell_c\}$. The value β^* that achieves the projection satisfies for each i,

$$\mathsf{E}\big[\big(c(X(t)) - \overline{c}_\beta(Z(t),U(t))\big)\psi_i^c(Z(t),U(t))\big] = 0$$

We thereby obtain the explicit representation,

$$\begin{aligned}\beta^* &= \Sigma_c^{-1}\mathsf{E}[c(X(t))\psi^c(Z(t),U(t))]\\ \Sigma_c &:= \mathsf{E}[\psi^c(Z(t),U(t))\psi^c(Z(t),U(t))^{\intercal}]\,,\end{aligned} \tag{18}$$

with $\psi^c = (\psi_1^c, \dots, \psi_{\ell_c}^c)^{\intercal}$. This representation is similar to the expression for Θ^* given in (7), and in fact the derivation is analogous. It is clear that β^* can be estimated using Monte-Carlo, just as Θ^* is estimated.

E. Example: Completely decentralized control of a network

Figure 1 shows a network example to illustrate the construction of a decentralized policy. This network consists of two stations, four buffers, with two exongenous arrival processes. We take a controlled random walk (CRW) model of the form,

$$\begin{aligned}Q_1(t+1) &= Q_1(t) - S_1(t+1)U_1(t) + A_1(t+1)\\ Q_4(t+1) &= Q_4(t) - S_4(t+1)U_4(t) + S_3(t+1)U_3(t)\end{aligned} \tag{19}$$

where the dynamics at Station 2 are defined analogously. The sequence $\boldsymbol{S}_i$ is taken Bernoulli with parameter μ_i. The two arrival processes $(\boldsymbol{A}_1, \boldsymbol{A}_3)$ are i.i.d., taking values in the positive integers. In the numerical results that follow they are scaled Bernoulli: For a fixed integer $\kappa \ge 1$, the distribution of $\kappa^{-1}\boldsymbol{A}_i$ is Bernoulli for $i = 1, 3$.

We let $c\colon \mathsf{X} \to \mathbb{R}_+$ denote a cost function on the state space $\mathsf{X} = \mathbb{Z}_+^4$ of buffer levels. In the numerical results that follow we take $c(x) = \|x\|_1$, the ℓ_1 norm.

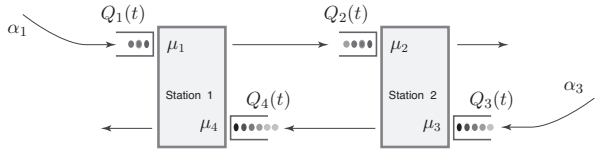

Fig. 1. A two-station network

We restrict to decentralized Markov policies, possibly randomized. Hence, for some feedback law $\phi\colon \mathsf{X} \to [0,1]^4$,

$$\mathsf{P}\{U_i(t) = 1 \mid Q_0^t, U_0^{t-1}\} = \phi_i(x), \qquad Q(t) = x,\ t \ge 0.$$

By decentralized we mean that the functions $\phi_1(x)$, $\phi_4(x)$ should only depend on $\binom{x_1}{x_4}$, and $\phi_2(x)$, $\phi_3(x)$ should only depend on $\binom{x_2}{x_3}$. The goal is to build an approximate model described as an MDP model at Station 1.

This setting is *optimistic* since control is based on virtually no information. It is far more restrictive then the setting of the MaxWeight policy, which assumes knowledge of buffer levels at down-stream nodes of one-hop distance [27], [20].

For the purposes of model and policy construction we set $Z(t) = X^1(t) = (Q_1(t), Q_4(t))$ and $X^2(t) = (Q_2(t), Q_3(t))$. The local control is the pair $(U_1(t), U_4(t))$, subject to the constraint that $U_1(t) + U_4(t) \le 1$. We assume that the policy is non-idling, meaning that $U_1(t)+U_4(t) = 1$ whenever $Q_1(t) + Q_4(t) \ge 1$.

We follow the four steps outlined above: Given the nth policy,

(i) Obtain the optimal-prediction MDP model $T^{(n)}_{u_1,u_4}\big(\binom{x_1}{x_4}, \binom{y_1}{y_4}\big) =$

$$\mathsf{P}\Big\{\tbinom{Q_1(t+1)=y_1}{Q_4(t+1)=y_4}\Big|\tbinom{Q_1(t)=x_1}{Q_4(t)=x_4}, \tbinom{U_1(t)=u_1}{U_4(t)=u_4}\Big\}$$

where the conditional probability is taken in steady-state.

(ii) Obtain the conditional cost,

$$\overline{c}\big(\tbinom{x_1}{x_4}, \tbinom{u_1}{u_4}\big) = \mathsf{E}\Big[c(Q_1(t+1))\Big|\tbinom{Q_1(t)=x_1}{Q_4(t)=x_4}, \tbinom{U_1(t)=u_1}{U_4(t)=u_4}\Big]$$

again taken in steady-state.

(iii) Obtain the optimal policy $\phi^{(n*)}$ for the MDP model with transition law $T^{(n)}$ and cost $\overline{c}$. For the average-cost optimality criterion this is obtained by solving the dynamic programming equation,

$h^{(n*)}\binom{x_1}{x_4} = -\eta^{(n*)} +$

$$\min\big\{\overline{c}\big(\tbinom{x_1}{x_4}, \tbinom{u_1}{u_4}\big) + \textstyle\sum_y T^{(n)}_{\binom{u_1}{u_4}}\big(\tbinom{x_1}{x_4}, \tbinom{y_1}{y_4}\big)h^{(n*)}\tbinom{y_1}{y_4}\big\} \tag{20}$$

where $\eta^{(n*)}$ is a constant — equal to the average cost under the optimal policy for the nth model. For each $\binom{x_1}{x_4}$, the value $\phi^{(n*)}(x_1, x_4) \in \{0,1\} \times \{0,1\}$ is taken as any minimizer in (20), where the minimum is over all admissible controls.

(iv) Define a new randomized policy via (17).

A sequence of policies is obtained at Station 2 following the symmetric procedure.

In the numerical results below the following parameter values were used: $\mu_1 = \mu_3$, $\mu_2 = \mu_4 = 3\mu_1$, and $\alpha_1 = \alpha_3 = \frac{1}{4}\mu_2\rho$, where ρ is the network load, taken to be $\rho = 9/10$. The burstiness parameter for the arrival process was taken to be $\kappa = 2$. The model was constructed so that only one event can occur at a time: For each t, $i \ne j$, and each k,

$$S_i(t)S_j(t) = S_i(t)A_k(t) = 0$$

This model is of the form obtained via *uniformization* [15], [20].

Observe that the system is completely symmetric, and hence local models at the two stations can be assumed identical. Details of the implementation of the four estimation-modeling-control steps are described as follows.

(i) The optimal-prediction MDP model $T^{(n)}$ was obtained after 10^6 samples. Only the conditional statistics of arrivals to buffers 4 and 2 are required to specify this model. The

conditional probability is defined for $a = 0, 1$ by,
$p_{A_4^I}(a \mid \binom{x_1}{x_4}) =$

$$\mathsf{P}\{S_3(t+1)U_3(t) = 1 \mid Q_1(t) = x_1,\, Q_4(t) = x_4\}$$

Estimates were obtained via Monte-Carlo, exploiting symmetry of the model. Given $Q_1(t) = x_1$, $Q_4(t) = x_4$, and $Q_3(t) = x_3$, $Q_2(t) = x_2$, updates of the entries $p_{A_4^I}(1 \mid \binom{x_1}{x_4})$ and $p_{A_4^I}(1 \mid \binom{x_3}{x_2})$ were obtained at time t via the Monte-Carlo recursions,

$$p_{A_4^I}(1 \mid \tbinom{x_1}{x_4}) \longleftarrow p_{A_4^I}(1 \mid \tbinom{x_1}{x_4}) + (-p_{A_4^I}(1 \mid \tbinom{x_1}{x_4}) + \mu_3 U_3(t))/(t+1)$$

$$p_{A_4^I}(1 \mid \tbinom{x_3}{x_2}) \longleftarrow p_{A_4^I}(1 \mid \tbinom{x_3}{x_2}) + (-p_{A_4^I}(1 \mid \tbinom{x_3}{x_2}) + \mu_1 U_1(t))/(t+1)$$

(ii) The conditional cost was obtained after 10^6 samples. Again exploiting symmetry, the Monte-Carlo recursions at time t, given $Q(t) = x = (x_1, x_2, x_3, x_4)$, are expressed

$$\overline{c}(\tbinom{x_1}{x_4}, \tbinom{u_1}{u_4}) \longleftarrow \overline{c}(\tbinom{x_1}{x_4}, \tbinom{u_1}{u_4}) + (-\overline{c}(\tbinom{x_1}{x_4}, \tbinom{u_1}{u_4}) + c(Q(t)))/(t+1)$$

$$\overline{c}(\tbinom{x_3}{x_2}, \tbinom{u_3}{u_2}) \longleftarrow \overline{c}(\tbinom{x_3}{x_2}, \tbinom{u_3}{u_2}) + (-\overline{c}(\tbinom{x_3}{x_2}, \tbinom{u_3}{u_2}) + c(Q(t)))/(t+1)$$

(iii) The optimal policy $\phi^{(n*)}$ for the MDP model with transition law $T^{(n)}$ was approximated via $5,000$ steps of value iteration [20].

(iv) The new randomized policy was obtained using the update rule (17) with $\gamma_n = 1/(2\sqrt{n}+1)$.

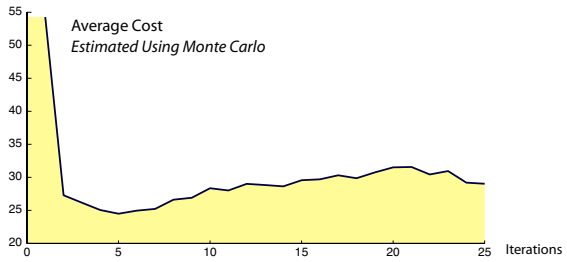

Fig. 2. Average cost for the nth policy, $n = 1, 2, \ldots 25$, estimated from two million observations.

Figure 2 shows the average cost, estimated using two million Monte-Carlo steps, for each of the 25 policies obtained. The initial policy was a perturbation of *serve the longest queue*:

$$\mathsf{P}\{U_1(t) = 1\} = \begin{cases} 0.85 & \text{if } Q_1(t) \geq Q_4(t); \\ 0.15 & \text{if } Q_1(t) < Q_4(t) \text{ and } Q_1(t) \geq 1. \end{cases} \tag{21}$$

Note that the average cost shown in Figure 2 is not monotone in the number of iterations. The cost drops quickly, and then increases slightly. Similar behavior was seen in all experiments.

The four-step algorithm is intended to mimic the policy improvement algorithm (PIA) for which it is known that the average cost from successive policies is monotone decreasing [19]. The lack of monotonicity seen here may be a product of the fact that the model $T^{(n)}_{u_1,u_4}(\binom{x_1}{x_4}, \binom{y_1}{y_4})$ changes with n. Another factor that impairs performance is the imposition of randomization in Step (iv) of the procedure.

Shown on the right in Figure 3 is an illustration of the fifth policy obtained using this algorithm. As in the "serve the longest queue" policy shown on the left, Buffer 1 receives higher priority if its contents are larger. However, the policy shown on the right in the figure is more similar to a threshold policy of the form "Serve buffer 4 whenever $Q_1 \leq \bar{x}_1$ and $Q_4 \geq 1$", where $\bar{x}_1 \sim 10$.

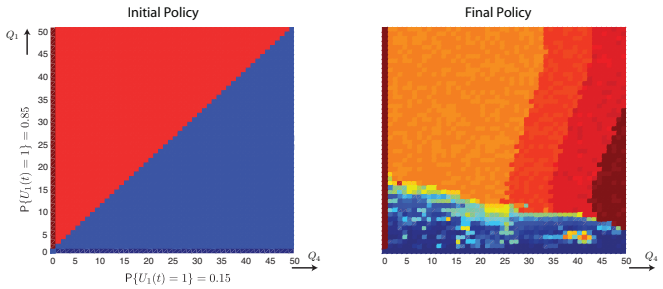

Fig. 3. Plot illustrating the initial policy and the fifth policy. Each is randomized — The color indicates the probability that $U_1(t)$ is equal to one (which is one minus the probability that $U_4(t)$ is equal to one, provided $Q_1(t) + Q_4(t) \geq 1$). The dark blue indicates a value of approximately 0.1, and dark red approximately 0.9.

The average cost for the decentralized policy illustrated in Figure 3 is approximately 25. We consider two classes of policies for comparison: Versions of the MaxWeight policy (MW) and logarithmic safety-stock policies (LogSS). See Section 4.8 of [20] for an introduction to the MW policy.

Each of the policies considered is non-idling. Hence $u_1 = 1$ whenever $x_4 = 0$ and $x_1 \geq 1$. If $x_4 > 0$ then, provided $x_1 \geq 1$, the policies are specified by the following decision regions:

$$\text{MW}: \quad u_1 = \mathbb{I}\{\mu_1(x_1 - x_2) > \tfrac{1}{5} n \mu_4 x_4\}$$

$$\text{LogSS}: \quad u_1 = \mathbb{I}\{x_2 + x_3 < 2n \log\big(1 + |x|/(2n)\big)\}$$

where $|x|$ denotes the ℓ_1 norm. The parameter n was varied from 0 to 10, where for $n = 0$ we take $0 \log(\infty) = 0$.

The first equation describes the MW policy defined by the diagonal matrix $D = \text{diag}\,(1, n/5, 1, n/5)$. The standard MW policy is obtained with $n = 5$. The LogSS policy is intended to approximately minimize the workload process, while minimizing $c(Q(t))$ subject to the current workload value for each time t — see [20] for further discussion (in particular, Example 6.7.2).

Results from simulations using these policies are shown in Figure 4. The performance of the decentralized policy is approximately equal to that of the standard MW policy. The LogSS policy with $n \in \{2, \ldots, 10\}$ is clearly the best of the three policies in terms of performance, but this policy requires complete global information.

Once again, the quantity of information utilized by the decentralized policy is much lower than would be expected

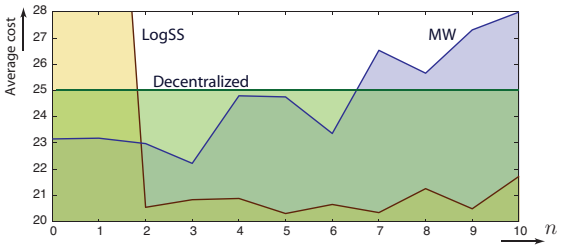

Fig. 4. Average cost performance of the decentralized policy compared to MaxWeight and LogSS.

in applications. In the cooperative setting considered here it is not unrealistic to assume sharing of global information, such as generalized workload in the routing model, or a global cost variable in other models.

IV. Conclusions

The optimal-prediction method is a standard work-horse in many areas. Its application in machine learning is either unrecognized, or taken for granted - nobody believes the real world is Markovian! By recalling the optimal prediction properties of Shannon's construction we have identified generalizations and new applications.

In particular, the standard approach to decentralized control of MDP models is through the introduction of a belief state to transform the partially observed optimal control problem to one with full observations. While this approach can in principle lead to an optimal policy, the complexity is severe. We have demonstrated an alternative approach to decentralized control through the construction of multiple *local* Markovian models. Further details and other examples are described in the working paper [16].

Among the other applications considered in current research are,

(i) Applications in other decision making domains such as finance.

(ii) Hypothesis testing and change detection.

(iii) Control variates for simulation variance reduction

The open problems are too long to list. Among the most interesting is the question of how to construct suitable local variables for application in decentralized control.

Acknowledgment

Research supported in part by DARPA DSO through contract FA9550-07-C-0024, NSF CCF 07-29031 *Robust Inference and Communication: Theory Algorithms and Performance Analysis*, and ITMANET DARPA RK 2006-07284.

The authors acknowledge numerous helpful comments from our colleagues at UTRC and UIUC. In particular, input from Jose Miguel Pasini and Prashant Mehta is very much appreciated.

References

[1] E. Altman. *Constrained Markov decision processes*. Stochastic Modeling. Chapman & Hall/CRC, Boca Raton, FL, 1999.

[2] D.P. Bertsekas and J. N. Tsitsiklis. *Neuro-Dynamic Programming*. Atena Scientific, Cambridge, Mass, 1996.

[3] V. S. Borkar. Average cost dynamic programming equations for controlled Markov chains with partial observations. *SIAM J. Control Optim.*, 39(3):673–681 (electronic), 2000.

[4] V. S. Borkar. Dynamic programming for ergodic control with partial observations. *Stoch. Proc. Applns.*, 103(2):293–310, 2003.

[5] V. S. Borkar and S. P. Meyn. The O.D.E. method for convergence of stochastic approximation and reinforcement learning. *SIAM J. Control Optim.*, 38(2):447–469, 2000. (also presented at the *IEEE CDC*, December, 1998).

[6] A. Chorin, O. Hald, and R. Kupferman. Optimal prediction with memory. *Physica D*, 166:239–257, 2002.

[7] A. J. Chorin, O. H. Hald, and R. Kupferman. Non-Markovian optimal prediction, 2001.

[8] M.D. Donsker and S.R.S. Varadhan. Asymptotic evaluation of certain Markov process expectations for large time. I. II. *Comm. Pure Appl. Math.*, 28:1–47; ibid. **28** (1975), 279–301, 1975.

[9] R. J. Elliott, L. Aggoun, and J. B. Moore. *Hidden Markov models*, volume 29 of *Applications of Mathematics (New York)*. Springer-Verlag, New York, 1995. Estimation and control.

[10] E. A. Feinberg and A. Shwartz, editors. *Handbook of Markov decision processes*. International Series in Operations Research & Management Science, 40. Kluwer Academic Publishers, Boston, MA, 2002. Methods and applications.

[11] R. A. Howard. *Dynamic Programming and Markov Processes*. John Wiley and Sons/MIT Press, New York, NY, 1960.

[12] V. R. Konda and V. S. Borkar. Actor-critic-type learning algorithms for Markov decision processes. *SIAM J. Control Optim.*, 38(1):94–123 (electronic), 1999.

[13] V. R. Konda and J. N. Tsitsiklis. On actor-critic algorithms. *SIAM J. Control Optim.*, 42(4):1143–1166 (electronic), 2003.

[14] I. Kontoyiannis and S. P. Meyn. Spectral theory and limit theorems for geometrically ergodic Markov processes. *Ann. Appl. Probab.*, 13:304–362, 2003. Presented at the INFORMS Applied Probability Conference, NYC, July, 2001.

[15] S. Lippman. Applying a new device in the optimization of exponential queueing systems. *Operations Res.*, 23:687–710, 1975.

[16] G. Mathew and S. Meyn. Learning macroscopic dynamics for optimal prediction. Submitted to 2008 IEEE Conf. on Dec. and Control. Preliminary version presented at Info. Thy. & Appl. at ITA, UCSD 2008.

[17] F. S. Melo, S. Meyn, and M. Isabel Ribeiro. An analysis of reinforcement learning with function approximation. In *Proceedings of ICML*, pages 664–671, 2008.

[18] Francisco S. Melo and M. Isabel Ribeiro. Convergence of Q-learning with linear function approximation. In *Proceedings of the 2007 European Control Conference*, pages 2671–2678, 2007.

[19] S. P. Meyn. The policy iteration algorithm for average reward Markov decision processes with general state space. *IEEE Trans. Automat. Control*, 42(12):1663–1680, 1997.

[20] S. P. Meyn. *Control techniques for complex networks*. Cambridge University Press, Cambridge, 2007.

[21] S. P. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability*. Springer-Verlag, London, second edition, 1993. 2008 Edition to appear, Cambride University Press, Cambridge Mathematical Library. 1993 edition online: `http://black.csl.uiuc.edu/~meyn/pages/book.html`.

[22] H. Mori. Transport, collective motion, and brownian motion. *Progress of Theoretical Physics*, 33:423–455, 1965.

[23] Dirk Ormoneit and Śaunak Sen. Kernel-based reinforcement learning. *Mach. Learn.*, 49(2-3):161–178, 2002.

[24] P. J. Schweitzer. Perturbation theory and finite Markov chains. *J. Appl. Prob.*, 5:401–403, 1968.

[25] C.E. Shannon. A mathematical theory of communication. *Bell System Tech. J.*, 27:379–423, 623–656, 1948.

[26] R.S. Sutton and A.G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, on-line edition at http://www.cs.ualberta.ca/~sutton/book/the-book.html edition, 1998.

[27] L. Tassiulas and A. Ephremides. Jointly optimal routing and scheduling in packet radio networks. *IEEE Trans. Inform. Theory*, 38(1):165–168, 1992.

[28] J. N. Tsitsiklis and B. Van Roy. Feature-based methods for large scale dynamic programming. *Mach. Learn.*, 22(1-3):59–94, 1996.

[29] R. Zwanzig. *Nonequilibrium Statistical Mechanics*. Oxford University Press, Oxford, England, 2001.