Networked Markov Decision Processes with Delays

Sachin Adlakha, Sanjay Lall, and Andrea Goldsmith

IEEE Transactions on Automatic Control, Vol. 57, No. 4, pp. 1013–1018, April 2012

Abstract

We consider a networked control system, where each subsystem evolves as a Markov decision process with some extra inputs from other systems. Each subsystem is coupled to its neighbors via communication links over which the signals are delayed, but are otherwise transmitted noise-free. A centralized controller receives delayed state information from each subsystem. The control action applied to each subsystem takes effect after a certain delay rather than immediately. We give an explicit bound on the finite history of measurement and control that is required for the optimal control of such networked Markov decision processes. We also show that these bounds depend only on the underlying graph structure as well as the associated delays. Thus, the partially observed Markov decision process associated with a networked Markov decision process can be converted into an information state Markov decision process, whose state does not grow with time.

Keywords: Networked Systems, Markov Decision Processes, Delayed Systems.

I Introduction

We are interested in the control of an interconnected network of subsystems. Each subsystem is modeled as a Markov decision process (MDP), and the overall system is referred to as a *networked Markov decision process*, used to model a variety of control problems [1, 2]. This paper shows that for networked MDPs, the optimal controller is a function of a finite number of past observations.

We show that a networked MDP can be reduced to an MDP with a *sufficient information state* that does not grow with time. This sufficient information is a subset of the entire information state and it captures all relevant information required for the optimal control. This significantly reduces the computational complexity associated with obtaining an optimal controller for networked MDPs. We also give explicit tight bounds on the number of past observations required to compute an optimal controller. We show that for networked MDPs, the results depend only upon the network structure and the associated delays.

For example, consider the two coupled subsystems

$$\begin{aligned} x^1_{t+1} &= f^1\big(x^1_t, u_t, w^1_t, x^2_{t-2}\big) \\ x^2_{t+1} &= f^2\big(x^2_t, w^2_t, x^1_{t-2}\big) \end{aligned}$$

Here x^i is the state of subsystem i, and w^1, w^2 are IID random processes. We impose the constraint that at time t, the controller can measure x^1_t and x^2_{t-1}, so that it receives delayed measurements from the second subsystem. Notice also that the above dynamics has two additional delays, each of two time-steps, in the coupling between the subsystems. For this system, we show that there is an optimal controller of the form

$$u_t = \mu_t(x^1_t, x^1_{t-1}, x^1_{t-2}, x^1_{t-3}, x^2_{t-1}, x^2_{t-2})$$

Here μ_t is simply a function, and in particular it does not represent a system with state. This equation therefore tells us how much memory, or history dependence, the controller has. This result does not depend on the details of the functions f^i or the cost function, and in this paper we show that it holds for finite-state systems with discounted and average cost models.

This paper determines explicitly the history dependence of the controller, given the graph of delays according to which the systems are interconnected, the measurement delays, and the control action delays. The results of this paper also provide a unifying framework covering existing special cases, such as the delayed measurement problem, where the dynamics are

$$x_{t+1} = f\big(x_t, u_t, w_t\big)$$

and at time t the controller has access to x_{t-n}. Then there is an optimal controller with the form

$$u_t = \mu_t(u_{t-n}, \dots, u_{t-1}, x_{t-n})$$

which is the well-known result of [3].

In the framework of this paper, the subsystems are coupled to each other via communication links that are noise free, pure delay lines, without packet losses or noisy observations. The delays are fixed but may be different for each interconnection. We assume that each subsystem has a finite state space. A centralized controller receives delayed state measurements from each subsystem

S. Adlakha is with the Center for Mathematics of Information, California Institute of Technology, Pasadena, CA 91106 USA email: adlakha@caltech.edu

S. Lall is with the Department of Electrical Engineering and Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305 USA email: lall@stanford.edu

A. Goldsmith is with the Department of Electrical Engineering, Stanford University, Stanford, CA 94305 USA email: andrea@ee.stanford.edu

and computes an optimal control action to be applied to each subsystem, which takes effect after a certain delay. Although the controller receives state information from each subsystem, each of these states is delayed by different amounts, and so the current state of each subsystem is not available to the controller. This system can thus be represented as a partially observed Markov decision process (POMDP).

Optimal control design for MDPs and POMDPs has been studied extensively in the literature [4, 5, 6, 7]. There are two standard approaches to optimal control of POMDPs. The first approach generates a policy that is a function of the entire history of observation; this history is called an *information state* and it grows without bound as time increases. In the second approach, the controller is a function of the *belief state* which is the posterior distribution of the current state of the system conditioned on the entire observation history. For many problems, the set of belief states is high dimensional and, in general, the computation required to compute an optimal controller is prohibitively large. We are therefore motivated to find a representation of the belief state that is as small as possible.

MDPs with delays have also been studied in the literature. Altman and Nain [3] consider an MDP with delayed state availability and showed that an optimal controller is a function of the last observed state and the control actions since the last observed state. Bander and White [8] extended this result to the case where partial state observation is available after a delay. MDPs with control action delays are considered in [9]. In [10], the authors unified these results by considering an MDP with observation delays, action delays as well as cost delays. They also extended the result to the case of random delays. Optimal control for linear systems with control action delay was also considered in [11]. However, these works consider only a single system with delayed information to the controller.

Among the earliest works on distributed systems with delays is [12], where a separation structure for the *one-step delay sharing pattern* for a system with general nonlinear dynamics was obtained. Optimal control of linear systems with one-step delay sharing was also studied in [13] in an input-output framework. The above works considered systems with uniform delay patterns. That is, each part of the state is delayed by the same amount. In this work, we consider networked systems with generalized delay patterns where each part of the state can potentially be delayed by a different amount. Compared to our previous conference papers [14, 15], we consider the case where we allow for arbitrary, finite but fixed delays between subsystems, finite delays in receiving observations as well as finite delays in applying control inputs to subsystems.

The rest of the paper is organized as follows. In Section II, we briefly describe POMDPs and define the information state for POMDPs. In Section III, we describe networked MDPs without action delays. The information state for networked MDPs without action delays is derived in Section IV and this result is then extended to networked MDPs with action delays in subsection IV-B. Section V concludes the paper.

I-A Notation

We use x_t^i to denote the state of the subsystem i at time t, use y_t^i to denote the observation received from subsystem i, and u_t^i to denote the control input applied to subsystem i. If there is only one subsystem, we drop the superscript and use x_t, y_t and u_t to represent the state, the observation and the control input for that single subsystem. We also denote by z, s and a the realization of the state x, observation y and control action u. We define $x^i_{t_1:t_2} := \left(x^i_{t_1}, \dots, x^i_{t_2}\right)$ to be the list of states of subsystem i from time t_1 to t_2. If $t_2 < t_1$, this is an empty list. The notation $x_{0:t} = z_{0:t}$ is interpreted as an element-wise equality, meaning $x_0 = z_0, x_1 = z_1, \dots, x_t = z_t$. To denote the list of variables corresponding to all subsystems, we define $x_t := \left(x_t^1, \dots, x_t^n\right)$. Similarly, we define $u_t := \left(u_t^1, \dots, u_t^n\right)$ to be the control action applied to all subsystems at time t. We define $A^i_{0:t} := A^i_0 A^i_1 \dots A^i_t$ so that $A^i_{0:t}$ is a product of functions. For a set $\mathcal{X}$, the notation $\mathcal{X}^n$ has two meanings. In one case it denotes the n-fold Cartesian product of the set, and in the other we use the superscript to label distinct sets, so that $\mathcal{X}^i$ is the set of states of system i. We rely on context to distinguish these two uses. We write $\mathbb{Z}^+$ for the set of non-negative integers.

II Model and Definitions

II-A Partially Observed Markov Decision Processes

A POMDP is a generalization of an MDP and is used to model a variety of sequential decision processes. The goal of the controller is to choose a sequence of actions to optimize a predetermined criterion. We assume that the decisions are made at discrete times $t \in \mathbb{Z}^+$. Note that in a POMDP the state of the system is not fully observable [6, 16].

A POMDP is a tuple $\left(\mathcal{X}, \mathcal{Y}, \mathcal{U}, A_0(\cdot), \{A_t(\cdot,\cdot,\cdot),\ t \geq 1\}, C_t(\cdot,\cdot), g_t(\cdot,\cdot)\right)$, where $\mathcal{X}$ is the set of all possible states, $\mathcal{Y}$ is the set of all possible observations and $\mathcal{U}$ is the set of all possible actions taken by a player. Here $A_0(z_0) = \mathrm{Prob}(x_0 = z_0)$ is the probability mass function of the initial state of the system. For $t > 0$,

$$\begin{aligned} &A_t\left(z_t, z_{t-1}, a_{t-1}\right) \\ &\qquad = \mathrm{Prob}\left(x_t = z_t \mid x_{t-1} = z_{t-1}, u_{t-1} = a_{t-1}\right) \end{aligned} \quad (1)$$

is the conditional probability of state x_t given the previous state x_{t-1} and the applied input u_{t-1}. The sequence C_t is the observation kernel that gives the probability of receiving an observation by the controller at time t. That is, $C_t\big(s_t, z_t\big) = \text{Prob}\left(y_t = s_t \mid x_t = z_t\right)$. The sequence $g_t\left(x_t, u_t\right)$ represents the cost at time t and it depends on the current state x_t of the system as well as the action u_t taken at time t. We will assume that sets $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{U}$ are finite.

The decision in a POMDP is made based on the information available to the controller. We define h_t^{pomdp} to be the information available to the controller at time t, given by $h_t^{\text{pomdp}} = \big(u_{0:t-1}, y_{0:t}\big)$. Also, we use i_t^{pomdp} to denote a realization of h_t^{pomdp} as $i_t^{\text{pomdp}} = \big(a_{0:t-1}, s_{0:t}\big)$. For partially observed discrete time dynamic processes, the POMDP policy gives a probability distribution over possible actions or controls as a function of the information available to the decision-maker. That is $K_t(a_t, i_t) = \text{Prob}(u_t = a_t \mid h_t^{\text{pomdp}} = i_t)$.

For the remainder of the paper, we will suppress the state and action spaces in the notation and describe a POMDP by the tuple $\big(A, C, g\big)$, where we use $A = \{A_t,\ t \geq 0\}$, $C = \{C_t, t \geq 0\}$ and $g = \{g_t, t \geq 0\}$. We make the following assumption regarding the POMDP.

Assumption 1 *The POMDP $\big(A, C, g\big)$ satisfies the condition $A_0(z_0) > 0$ for all $z_0 \in \mathcal{X}$, $A_t\big(\cdot, z_{t-1}, a_{t-1}\big) > 0$ for all $z_{t-1} \in \mathcal{X}$, $a_{t-1} \in \mathcal{U}$ and for all $t \geq 1$. Furthermore, $C_t\big(\cdot, z_t\big) > 0$ for all $z_t \in \mathcal{X}$ and for all $t \geq 0$.*

II-B Information State for POMDPs

An information state for a POMDP summarizes information about the history of the POMDP to enable sufficient prediction of the future to make an optimal decision. A POMDP can be reformulated as an MDP using the information state. The information state consists of either a complete history of observations and actions or their corresponding sufficient statistics [6]. For the purpose of this paper, we define the term **sufficient information state** to mean a function of the past observations of the POMDP that is detailed enough to permit an optimal controller to use the history processed through this function as its only input. Using the sufficient information state, a POMDP can be converted into an MDP with observable state such that the optimal controller for this MDP also minimizes the cost function for the original POMDP. A precise statement of this result is given in Theorem 1.

Definition 1 *Suppose (A, C, g) is a POMDP satisfying Assumption 1 and define a sequence of functions $\gamma_t : \mathcal{U}^t \times \mathcal{Y}^{t+1} \to \mathcal{Q}$. Define $\xi_t = \gamma_t\left(u_{0:t-1}, y_{0:t}\right)$ and let γ_t be such that for all $q \in \mathcal{Q}$, there exists some $s_{0:t} \in \mathcal{Y}^{t+1}$ and $a_{0:t-1} \in \mathcal{U}^t$ such that $\gamma_t\left(s_{0:t}, a_{0:t-1}\right) = q$. Then ξ_t is called a sufficient information state for the POMDP if there exists an MDP $(\tilde{A}, \tilde{g})$ over the state space $\mathcal{Q}$ and the action space $\mathcal{U}$ such that, for all POMDP policies K, we have*

1) $\tilde{A}$ is a sequence such that

$$\tilde{A}_{t+1}\left(q_{t+1}, q_t, a_t\right) = \\ \text{Prob}\left(\xi_{t+1} = q_{t+1} \mid \xi_{0:t} = q_{0:t}, u_{0:t} = a_{0:t}\right). \quad (2)$$

2) $\tilde{g}$ is a sequence $\tilde{g}_0,\ \tilde{g}_1 \ldots$ such that

$$\tilde{g}_t\left(q_t, a_t\right) = \mathbb{E}\left(g_t\left(x_t, a_t\right) \mid \xi_t = q_t, u_t = a_t\right). \quad (3)$$

3) For all $t \geq 0$, we have

$$\text{Prob}\Big(x_t = z_t \mid \xi_t = \gamma_t\left(s_{0:t}, a_{0:t-1}\right), \ldots, \\ \xi_0 = \gamma_0(s_0), u_{0:t-1} = a_{0:t-1}\Big) = \\ \text{Prob}\left(x_t = z_t \mid y_{0:t} = s_{0:t}, u_{0:t-1} = a_{0:t-1}\right). \quad (4)$$

Note that the random variables x_t and u_t depend on the chosen policy K, and for any policy K, the conditional probabilities in equation (2) and (4), and the conditional expectation in (3) are well defined. Also, note that $\tilde{A}$ in equation (2), $\tilde{g}_t$ in equation (3) and the conditional probability in equation (4) are independent of the POMDP policy K. Furthermore, equation (2) shows that given the action sequence or the policy the evolution of ξ_t is Markov. From the above definition, it is clear that associated with any POMDP is a sufficient information state MDP $(\tilde{A}, \tilde{g})$. Also note that the sequence $\tilde{A}$ is independent of time if the original POMDP is stationary.

Let $h_t^{\text{i-mdp}}$ be the history of the sufficient information state MDP at time t. Then we have $h_t^{\text{i-mdp}} = \big(u_{0:t-1}, \xi_{0:t}\big)$. We will use $i_t^{\text{i-mdp}}$ to denote a realization of $h_t^{\text{i-mdp}}$ as $i_t^{\text{i-mdp}} = \big(a_{0:t-1}, q_{0:t}\big)$. As before, we define a sufficient information state MDP policy as a mapping from the history of the information state MDP to an action at time t. Let $\tilde{K}_t$ be a sufficient information state MDP policy. As before, we can interpret $\tilde{K}_t$ as $\tilde{K}_t(a_t, i_t) = \text{Prob}\left(u_t = a_t \mid h_t^{\text{i-mdp}} = i_t\right)$. The following theorem shows that we can find an optimal POMDP policy by considering the associated MDP over the sufficient information state.

Theorem 1 *Consider the POMDP (A, C, g) and let $\mathcal{P}_{pomdp}$ be the set of all POMDP policies. Let $(\tilde{A}, \tilde{g})$ be the sufficient information state MDP associated with the given POMDP and let $\mathcal{P}_{i\text{-}mdp}$ be the set of all sufficient information state MDP policies. Then, for any T, we have*

$$\min_{\substack{K_0,\ldots,K_T \\ K_t \in \mathcal{P}_{pomdp}}} \sum_{t=0}^{T} \mathbb{E}\big[g_t(z_t, a_t)\big] = \min_{\substack{K_0,\ldots,K_T \\ K_t \in \mathcal{P}_{i\text{-}mdp}}} \sum_{t=0}^{T} \mathbb{E}\big[\tilde{g}_t(q_t, a_t)\big]$$

Proof The proof follows standard dynamic programming techniques. See for example Chapter 6 of [6]. ■

From the above theorem, it is clear that one can find an optimal policy for a POMDP by transforming it into a sufficient information state MDP. Given an optimal sufficient information state policy $\tilde{K}^{\text{opt}}$ one may immediately compute the optimal POMDP policy by using $\tilde{K}^{\text{opt}}$ with γ. The optimal sufficient information state policy $\tilde{K}^{\text{opt}}$ may be found using the standard dynamic programming recursion. From [4], we know that the optimal policy for an MDP is a function of its current state. In other words, the optimal policy for a POMDP is just a function of its sufficient information state ξ_t. One such sufficient information state is the entire history of the POMDP, where γ_t is an identity function [6]. As we show below, for networked MDPs, the sufficient information state includes only the finite past history of observations and control actions. Also note that the above theorem can be easily extended to the infinite horizon case (both average cost as well as discounted cost), as long as the limiting value of the sum of the costs is well defined [17]. For the discounted infinite horizon case, we can incorporate the discount factor in the time dependent cost function.

III Networked Markov Decision Processes

III-A Definitions and Example

Let $\mathcal{G}$ be a weighted directed graph $(\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \dots, n\}$ is a finite set of vertices and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is a set of edges. Each vertex $i \in \mathcal{V}$ represents a Markov decision process. An edge $(i, j) \in \mathcal{E}$ if the MDP at vertex i directly affects the MDP at vertex j. Associated with each edge $(i, j) \in \mathcal{E}$ is a non-negative integer weight, M_{ij}, which specifies the delay for the dynamics of vertex i to propagate to vertex j. We assume without loss of generality that $(i, i) \notin \mathcal{E}$.

Associated with each $j \in \mathcal{V}$, let Pa^j be the parent vertices $\text{Pa}^j = \{\, i \in \mathcal{V} \mid (i, j) \in \mathcal{E} \,\}$ and let Ch^j be the child vertices $\text{Ch}^j = \{\, i \in \mathcal{V} \mid (j, i) \in \mathcal{E} \,\}$. At each time t, the state of the MDP at vertex i belongs to a finite set $\mathcal{X}^i$. The control action taken at vertex i is drawn out of a finite set $\mathcal{U}^i$.

In the remainder of the paper, we denote $\mathcal{X}^{-i} = \prod_{j \in \text{Pa}^i} \mathcal{X}^j$. We also denote $\mathcal{X}^{(n)} = \prod_{i=1}^n \mathcal{X}^i$ as the Cartesian product of the state space corresponding to all vertices. Similarly, we define $\mathcal{U}^{(n)} = \prod_{i=1}^n \mathcal{U}^i$.

Definition 2 *A networked Markov decision process is a tuple (A, g) where*

1) A is a set of transition matrices $\{A_t^i,\ t \geq 0 \mid i \in \mathcal{V}\}$ with $A_0^i : \mathcal{X}^i \to [0, 1]$ for all $i \in \mathcal{V}$, such that for all $z_0 \in \mathcal{X}^i$, we have $A_0^i\,(z_0) \geq 0$ and $\sum_{z_0} A_0^i\,(z_0) = 1$. For $t > 0$, we have $A_t^i : \mathcal{X}^i \times \mathcal{X}^i \times \mathcal{X}^{-i} \times \mathcal{U}^i \to [0, 1]$ such that, for all $i \in \mathcal{V}$ and for all $a_t \in \mathcal{U}^i$ and $\tilde{z}_t \in \mathcal{X}^{-i}$ we have $A_t^i(z_t', z_t, \tilde{z}_t, a_t) \geq 0$ for all $z_t', z_t \in \mathcal{X}^i$, and $\sum_{z_t'} A_t^i(z_t', z_t, \tilde{z}_t, a_t) = 1$ for all $z_t \in \mathcal{X}^i$.

2) g is a sequence $g_0, g_1, \dots$ with $g_t : \mathcal{X}^{(n)} \times \mathcal{U}^{(n)} \to [0, 1]$.

We make the following assumption regarding the networked MDP.

Assumption 2 *The networked MDP $\big(A, g\big)$ is such that $A_0^i\,(z_0) > 0$ for all $z_0 \in \mathcal{X}^i$ and for all $i \in \mathcal{V}$. Furthermore, $A_t^i(\cdot, z_t, \tilde{z}_t, a_t) > 0$ for all $i \in \mathcal{V}$, for all $z_t \in \mathcal{X}^i$, for all $a_t \in \mathcal{U}^i$, for all $\tilde{z}_t \in \mathcal{X}^{-i}$ and for all $t \geq 1$.*

In a networked MDP, the controller needs to choose a control action corresponding to each vertex $i \in \mathcal{V}$. Associated with each vertex $i \in \mathcal{V}$ of a networked MDP, we have a non-negative integer N_i which specifies the delay in receiving the state measurement from system i. We define $h_t^{\text{n-mdp}}$ to be the information available to the decision-maker at time t, given by $h_t^{\text{n-mdp}} = \big(x^1_{0:t-N_1}, u^1_{0:t-1}, \dots, x^n_{0:t-N_n}, u^n_{0:t-1}\big)$. Also define $i_t^{\text{n-mdp}}$ to be a realization of $h_t^{\text{n-mdp}}$ as $i_t^{\text{n-mdp}} = \big(z^1_{0:t-N_1}, a^1_{0:t-1}, \dots, z^n_{0:t-N_n}, a^n_{0:t-1}\big)$.

Thus, the observations received by the decision-maker at time t consist of the state of the subsystem i delayed by N_i time steps. A networked MDP policy specifies the decisions taken at time t.

Definition 3 (Networked MDP Policy) *K is called a networked MDP policy if $K = (K_0, K_1, \dots)$ where $K_0 : \mathcal{U}^{(n)} \times \prod_{i=1}^n \big(\mathcal{X}^i\big)^{1-N_i} \to [0, 1]$ and $K_t : \mathcal{U}^{(n)} \times \prod_{i=1}^n \big(\mathcal{X}^i\big)^{t+1-N_i} \times \prod_{i=1}^n \big(\mathcal{U}^i\big)^t \to [0, 1]$, for all $t \geq 1$ such that for all $z_0 \in \prod_{i=1}^n \big(\mathcal{X}^i\big)^{1-N_i}$ we have $K_0(a_0, z_0) \geq 0$ for all $a_0 \in \mathcal{U}^{(n)}$ and $\sum_{a_0} K_0(a_0, z_0) = 1$. Also, for all $t \geq 1$, $a_t' \in \mathcal{U}^{(n)}$, $z \in \prod_{i=1}^n \big(\mathcal{X}^i\big)^{t+1-N_i}$ and $a_t \in \prod_{i=1}^n \big(\mathcal{U}^i\big)^t$ we have $K_t\,(a_t', z, a_t) \geq 0$ and $\sum_{a_t'} K_t\,(a_t', z, a_t) = 1$.*

Note that for all times t, the product $\prod_{i=1}^n \big(\mathcal{X}^i\big)^{t+1-N_i}$ in the above definition is taken over those i for which $t+1-N_i$ is strictly positive. For the networked systems, a general mixed control policy is defined as a sequence of transition matrices $K_t,\ t \geq 0$ given by $K_t(a_t, i_t) = \text{Prob}(u_t = a_t \mid h_t^{\text{n-mdp}} = i_t)$.

III-B The Networked MDP as a POMDP

In networked MDPs, although the controller receives state information from the subsystems, these states are delayed by different amounts. Thus, a networked MDP can be written as a POMDP. Consider a networked MDP as given in Definition 2. Let us define a new state $\hat{x}_t = \big\{x^i_{t-b':t} \mid i \in \mathcal{V}\big\}$, where we choose $b' = \max_{i,j \in \mathcal{V}} M_{ij} + \max_{i \in \mathcal{V}} N_i$. The state $\hat{x}$ is chosen such that in the resulting system the observation at time t is only a function $\hat{h}$ of the current state at time t. It is easy to check that there exists a function $\hat{f}$ such that $\hat{x}_{t+1} = \hat{f}\,(\hat{x}_t, u_t, w_t)$, where w_t is some noise process. Associated with this function is a transition probability mass function $\hat{A}_t\,(\hat{z}_{t+1}, \hat{z}_t, a_t)$, where $\hat{z}_t$ is the realization of the state $\hat{x}_t$. The observation at any time t is given as $\hat{y}_t = \hat{h}(\hat{x}_t)$. Corresponding to this observation process is a probability mass function $\hat{C}_t\left(\hat{s}_t, \hat{z}_t\right)$, where $\hat{s}_t$ is the realization of the observation $\hat{y}_t$ and is given as $\hat{s}_t = \left\{z^i_{t-N_i} \mid i \in \mathcal{V}\right\}$. The cost function is given as

$$\hat{g}_t\left(\hat{x}_t, u_t\right) = g_t\left(x_t, u_t\right) \tag{5}$$

It is easy to check that the tuple $\left(\hat{A}, \hat{C}, \hat{g}\right)$ is a POMDP as defined in Subsection II-A.

IV Information State for Networked Markov Decision Processes

Before we present the main result of the paper, we make the following definitions.

Definition 4 *Let*

$$d_i = \max\{N_i, \max_{k \in Pa^i}(N_k - M_{ki} - 1)\} \tag{6}$$

and define the integers b_i by

$$b_i = \max\{d_i, \max_{k \in Ch^i}(d_k + M_{ik})\} - N_i \tag{7}$$

In the remainder of the paper, we use the following additional notation. We define for each $t \geq 0$ the product function $P_t = A^1_{0:t}A^2_{0:t}\dots A^n_{0:t}$. Define $\alpha_t = \{z^i_{0:t-N_i}, a^i_{0:t-1} \mid i \in \mathcal{V}\}$. For any function f, we use the notation $\sum_{z \notin \alpha_t} f(z^1_0, z^1_1, \dots)$ to indicate a summation over $z^i_{t-N_i+1:t} \in \mathcal{X}^i$ for each $i \in \mathcal{V}$. We also define the set $\beta_t = \{z^i_{t-N_i-b_i:t-N_i}, a^i_{t-d_i:t} \mid i \in \mathcal{V}\}$ with a similar corresponding summation notation. Here we abuse set notation to refer to the set of named variables and not their values.

The following theorem is the main result of this paper. It specifies a sufficient information state for a networked MDP. It shows that a networked MDP can be converted into an MDP with a state that is bounded and does not grow with time.

Theorem 2 *Consider a networked MDP satisfying Assumption 2. Then,*

$$\xi_t = \left\{u^i_{t-d_i:t-1}, x^i_{t-N_i-b_i:t-N_i} \mid i \in \mathcal{V}\right\} \tag{8}$$

is a sufficient information state for the networked MDP.

IV-A Proofs

To prove the theorem, we check the conditions for a sufficient information state as given in Definition 1. Note that for a networked MDP, γ_t is a truncation function. From the definition of ξ_t it is clear that for all possible information states, there exists a sequence of states and actions that maps to a given information state. The following lemma shows that ξ_t as defined in (8) satisfies the first condition for a sufficient information state in (2).

Lemma 1 *Consider a networked MDP (A, g) and a networked MDP policy K. Define*

$$\tilde{A}_{t+1}(q_{t+1}, q_t, a_t) \triangleq \text{Prob}\left(\xi_{t+1} = q_{t+1} \mid \xi_t = q_t, u_t = a_t\right)$$

Then ξ_t satisfies the Markov property

$$\begin{aligned}&\tilde{A}_{t+1}(q_{t+1}, q_t, a_t)\\ &\qquad = \text{Prob}\left(\xi_{t+1} = q_{t+1} \mid \xi_{0:t} = q_{0:t}, u_{0:t} = a_{0:t}\right)\end{aligned}$$

and $\tilde{A}$ is independent of the policy K.

Proof By definition

$$\begin{aligned}L &= \text{Prob}\left(\xi_{t+1} = q_{t+1} \mid \xi_{0:t} = q_{0:t}, u_{0:t} = a_{0:t}\right)\\ &= \frac{\text{Prob}\left(\xi_{0:t+1} = q_{0:t+1}, u_{0:t} = a_{0:t}\right)}{\text{Prob}\left(\xi_{0:t}, = q_{0:t}, u_{0:t} = a_{0:t}\right)}.\end{aligned} \tag{9}$$

The sequence $\xi_{0:t}$ consists of $\{x^i_{0:t-N_i}, u^i_{0:t-1} \mid i \in \mathcal{V}\}$ and $q_{0:t}$ is the sequence $\{z^i_{0:t-N_i}, a^i_{0:t-1} \mid i \in \mathcal{V}\}$. Thus, we have

$$\begin{aligned}&\text{Prob}\left(\xi_{0:t} = q_{0:t}, u_{0:t} = a_{0:t}\right)\\ &\qquad = \text{Prob}\left(x^i_{0:t-N_i} = z^i_{0:t-N_i}, u^i_{0:t} = a^i_{0:t} \mid i \in \mathcal{V}\right).\end{aligned}$$

Furthermore, using the notation $P_t = A^1_{0:t}\dots A^n_{0:t}$, we have $P_tK_{0:t} = \text{Prob}\left(x^i_{0:t} = z^i_{0:t}, u^i_{0:t} = a^i_{0:t} \mid i \in \mathcal{V}\right)$. Let us denote the denominator of equation (9) by L_{den}. Then we have

$$\begin{aligned}L_{\text{den}} &= \text{Prob}\left(x^i_{0:t-N_i} = z^i_{0:t-N_i}, u^i_{0:t} = a^i_{0:t} \mid i \in \mathcal{V}\right)\\ &= \sum_{z \notin \alpha_t} P_tK_{0:t},\end{aligned} \tag{10}$$

Note that the arguments of the transition kernel A^i_t are $z^i_t, z^i_{t-1}, a^i_{t-1}, \{z^k_{t-1-M_{ki}} \mid k \in \text{Pa}^i\}$. We first show that some of the A^i_t's are independent of the variables being summed over. Consider an arbitrary $s \geq 0$, and suppose A^i_{t-s} depends upon at least one of $z^1_{t-N_1+1:t}, \dots, z^n_{t-N_n+1:t}$. Then we must have either $t - N_i + 1 \leq t - s$, or $t - N_i + 1 \leq t - s - 1$, or $t - N_k + 1 \leq t - s - 1 - M_{ki}$ for some $k \in \text{Pa}^i$, where each inequality arises from the corresponding argument of A^i_{t-s}. This implies that either $s \leq N_i - 1$ or $s \leq \max\{N_k - 1 - M_{ki} \mid k \in \text{Pa}^i\} - 1$. Hence for each i, the largest such s is exactly equal to $d_i - 1$ where d_i is defined by equation (6). Thus if $s \geq d_i$ then A^i_{t-s} does not depend on any of $z^1_{t-N_1+1:t}, \dots, z^n_{t-N_n+1:t}$. In other words, $A^i_{0:t-d_i}$ are independent of all the variables of summation. Furthermore, note that $K_{0:t}$ depends only on the variables in α_t, and hence is independent of the variables of the summation. Thus, the denominator of equation (9) is

$$L_{\text{den}} = \left(\prod_{i=1}^n A^i_{0:t-d_i}\right) K_{0:t} \sum_{z \notin \alpha_t} \prod_{i=1}^n A^i_{t-d_i+1:t} \tag{11}$$

Let us denote the numerator of equation (9) as L_{num}. Then,

$$L_{\text{num}} = \sum_{z \notin \alpha_{t+1}} P_{t+1} K_{0:t}. \tag{12}$$

Following the same argument as above, it is easy to verify that if $s \geq d_i - 1$, then A^i_{t-s} does not depend on any of $z^1_{t-N_1+2:t+1}, \dots, z^n_{t-N_n+2:t+1}$. Thus, $A^i_{0:t-d_i+1}$ are independent of the variables of the summation of L_{num}. We can thus write L_{num} as

$$L_{\text{num}} = \left(\prod_{i=1}^{n} A^i_{0:t-d_i} \right) K_{0:t} \sum_{z \notin \alpha_{t+1}} \prod_{i=1}^{n} A^i_{t-d_i+1:t+1}$$

Canceling the common factors from the numerator and denominator gives

$$L = \frac{\sum_{z \notin \alpha_{t+1}} \prod_{i=1}^{n} A^i_{t-d_i+1:t+1}}{\sum_{z \notin \alpha_t} \prod_{i=1}^{n} A^i_{t-d_i+1:t}}. \tag{13}$$

Using the definition of conditional probability, we can write

$$\begin{aligned} R &= \text{Prob}\left(\xi_{t+1} = q_{t+1} \mid \xi_t = q_t, u_t = a_t\right) \\ &= \frac{\text{Prob}\left(\xi_{t+1} = q_{t+1}, \xi_t = q_t, u_t = a_t\right)}{\text{Prob}\left(\xi_t = q_t, u_t = a_t\right)}. \end{aligned} \tag{14}$$

Let R_{den} denote the denominator of equation (14). Using the definition of ξ_t, we can write the denominator as $R_{\text{den}} = \sum_{a \notin \beta_t} \sum_{z \notin \beta_t} \sum_{z \notin \alpha_t} P_t K_{0:t}$, where this equality holds because of a similar argument as given for L_{den}. As before $A^i_{t-d_i}$ and $K_{0:t}$ are independent of the variables of summation $\{z \notin \alpha_t\}$ and hence we can write R_{den} as

$$R_{\text{den}} = \sum_{a \notin \beta_t} \sum_{z \notin \beta_t} \left(\prod_{i=1}^{n} A^i_{0:t-d_i} \right) K_{0:t} \underbrace{\sum_{z \notin \alpha_t} \prod_{i=1}^{n} A^i_{t-d_i+1:t}}_{\hat{R}_{\text{den}}}$$

Let us determine explicitly which variables $\hat{R}_{\text{den}}$ depends on. For notational convenience, denote

$$T = A^1_{t-d_1+1:t} \dots A^n_{t-d_n+1:t}$$

If T depends on z^i_s then we must either have $t - d_i \leq s$ or $t - d_k - M_{ik} \leq s$ for some $k \in \text{Ch}^i$. The first inequality holds if z^i_s occurs in $A^i_{t-d_i+1:t}$ and the second holds if it occurs in $A^k_{t-d_k+1:t}$. If $\hat{R}_{\text{den}}$ depends on $z^i_{t-N_i-r}$ then either $t - d_i \leq t - N_i - r$ or $t - d_k - M_{ik} \leq t - N_i - r$ for some $k \in \text{Ch}^i$, and these conditions imply that either $r \leq d_i - N_i$ or $r \leq \max\{d_k + M_{ik} \mid k \in \text{Ch}^i\} - N_i$. Using the definition of b_i in equation (7), these two inequalities imply that $r \leq b_i$. Thus $\hat{R}_{\text{den}}$ depends on $\{a^i_{t-d_i:t-1} \mid \in \mathcal{V}\}$ and $\{z^i_{t-N_i-b_i:t-N_i} \mid i \in \mathcal{V}\}$ and hence is independent of variables $\{a \notin \beta_t\}$ and $\{z \notin \beta_t\}$. Thus, we can write

$$R_{\text{den}} = \left(\sum_{a \notin \beta_t} \sum_{z \notin \beta_t} \prod_{i=1}^{n} A^i_{0:t-d_i} K_{0:t} \right) \sum_{z \notin \alpha_t} \prod_{i=1}^{n} A^i_{t-d_i+1:t} \tag{15}$$

Let R_{num} denote the numerator of equation (14). Then, $R_{\text{num}} = \sum_{a \notin \beta_t} \sum_{z \notin \beta_t} \sum_{z \notin \alpha_{t+1}} P_{t+1} K_{0:t}$. Using the same argument as for R_{den} we can write the numerator as

$$R_{\text{num}} = \left(\sum_{\substack{a \notin \beta_t \\ z \notin \beta_t}} \prod_{i=1}^{n} A^i_{0:t-d_i} K_{0:t} \right) \sum_{z \notin \alpha_{t+1}} \prod_{i=1}^{n} A^i_{t-d_i+1:t+1} \tag{16}$$

From equation (15) and equation (16) we have

$$R = \frac{\sum_{z \notin \alpha_{t+1}} \prod_{i=1}^{n} A^i_{t-d_i+1:t+1}}{\sum_{z \notin \alpha_t} \prod_{i=1}^{n} A^i_{t-d_i+1:t}}. \tag{17}$$

The result follows from equations (13) and (17). ■

The next lemma evaluates the cost function $\tilde{g}_t$ for the induced MDP and shows that it is independent of the POMDP policy.

Lemma 2 *The cost function as defined in equation* (3) *is independent of the POMDP policy K.*

Proof The proof follows from the definition of $\tilde{g}_t$ and uses an argument similar to that in Lemma 1. The proof is omitted for space constraints. ■

The following lemma shows that the conditional probability density function for the state at time t is same for the induced MDP and the original POMDP.

Lemma 3 *For all $t \geq 0$, we have*

$$\begin{aligned} &\text{Prob}\left(\hat{x}_t = \hat{z}_t \mid \xi_{0:t} = q_{0:t}, u_{0:t-1} = a_{0:t-1}\right) \\ &\qquad = \text{Prob}\left(\hat{x}_t = \hat{z}_t \mid \hat{y}_{0:t} = \hat{s}_{0:t}, u_{0:t-1} = a_{0:t-1}\right), \end{aligned}$$

where we have used the notation $\gamma_t\left(s_{0:t}, a_{0:t-1}\right) = q_t$.

Proof The proof trivially follows from the definition of the sequence $\xi_{0:t}$ and the sequence $\hat{y}_t$. ■

Proof of Theorem 2 From Lemmas 1, 2, and 3, we have that ξ_t as defined in equation (8) is a sufficient information state for a networked MDP. ■

IV-B Networked MDP with Action Delays

In this section, we extend our result to the case where the control action does not take effect immediately. Consider a networked Markov decision process with action delays. The system dynamics are $x^i_{t+1} = f^i\big(x^i_t, \{x^j_{t-M_{ji}} \mid j \in \text{Pa}^i\}, u^i_{t-P_i}, w^i_t\big)$, for all $i \in \mathcal{V}$. Here $u^i_{t-P_i}$ is the control action applied to subsystem i at time $t - P_i$.

To obtain a sufficient information state for a networked MDP with action delays, we convert this system into a networked MDP with no action delays. To do this, let us define a new state $\hat{x}^i_t = (x^i_t, u^i_{t-P_i:t-1})$ for all $i \in \mathcal{V}$. As before, if any $P_i = 0$, we interpret the list $u^i_{t-P_i:t-1}$ as empty and thus $\hat{x}^i_t = x^i_t$. This new state is chosen such that the state evolution of each subsystem at time $t+1$

depends on the current state and action at time t. Thus, a networked MDP with action delays can be reformulated as a networked MDP with no action delays and with system dynamics given as $\hat{x}^i_{t+1} = \hat{f}^i\big(\hat{x}^i_t, \{\hat{x}^j_{t-M_{ji}} \mid j \in \mathrm{Pa}^i\}, u^i_t, w^i_t\big)$ for all $i \in \mathcal{V}$. Using Theorem 2, we know that a sufficient information state for this new system consists of past states $\hat{x}^i_{t-b_i-N_i:t-N_i}$ and past control actions $u^i_{t-d_i:t-1}$ for all $i \in \mathcal{V}$. Let us define $\hat{d}_i$ by

$$\hat{d}_i = \begin{cases} d_i & \text{if } P_i = 0 \\ b_i + N_i + P_i & \text{otherwise} \end{cases} \tag{18}$$

Using this definition, it is easy to check that a sufficient information state for a networked MDP with action delays consists of past states $x^i_{t-b_i-N_i:t-N_i}$ and past control actions $u^i_{t-\hat{d}_i:t-1}$ for all $i \in \mathcal{V}$. This gives us the following theorem.

Theorem 3 *For a networked Markov decision process with action delays, the set*

$$\xi_t = \Big\{u^i_{t-\hat{d}_i:t-1}, x^i_{t-N_i-b_i:t-N_i} \mid i \in \mathcal{V}\Big\}$$

is a sufficient information state.

IV-C Discussion

From Theorem 2, we note that every networked MDP has a sufficient information state ξ_t given by equation (8), which depends only on the finite past history of the states and control actions. Thus, from Definition 1 we have that associated with every networked MDP is a tuple $(\tilde{A}, \tilde{g})$ where $\tilde{A}_t$ is the transition matrix given by $\tilde{A}_{t+1}(q_{t+1}, q_t, a_t) = \mathrm{Prob}\big(\xi_{t+1} = q_{t+1} \mid \xi_t = q_t, u_t = a_t\big)$ and $\tilde{g}_t$ is the cost function associated with this new MDP. The cost function is given by equation (3). From Theorem 1 we note that an optimal controller for the original POMDP can be found by considering the associated sufficient information state MDP. An optimal controller can be found using dynamic programming [4, 18] over the state space $\mathcal{Q}$ generated by ξ_t. This holds for both finite horizon as well as infinite horizon average cost and discounted cost models. In our earlier conference paper [19], we showed that the constants b_i and d_i relate to the Markov blanket of the Bayesian network associated with the networked MDP. We also provided a numerical example in [14], where we showed that for a certain class of linear systems, these information requirements are also necessary for the optimal control.

V Conclusions

We studied networked MDPs with delays between subsystems. Each subsystem transmits its state to a centralized controller via a link with an associated delay. The control action applied to each subsystem takes effect after a certain delay. Since the controller does not have access to the current state of the system, these systems are a special case of POMDPs. We show that for this special class of POMDPs, a sufficient information state is a function of a finite number of past system states and the past controller inputs. The number of past states as well as past inputs depends only on the underlying graph structure of the networked MDP as well as the associated delays. We also give explicit bounds on the number of past states and inputs required to compute an optimal control action for networked MDPs with delays. This result shows that the controller synthesis can be achieved at substantially lower computational cost. A dynamic programming algorithm based on the finite information state can be used to compute the optimal controller for such systems.

References

[1] J. Kuri and A. Kumar, "Optimal control of arrivals to queues with delayed queue length information," *IEEE Transactions on Automatic Control*, vol. 40, no. 8, pp. 1444–1450, 1995.

[2] E. Altman and S. Stidham, "Optimality of monotonic policies for two-action Markovian decision processes, with applications to control of queues with delayed information," *Queueing Systems*, vol. 21, no. 3, pp. 267–291, 1995.

[3] E. Altman and P. Nain, "Closed-loop control with delayed information," *Performance Evaluation Review*, vol. 20, pp. 193–204, 1992.

[4] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming.* Wiley, 1994.

[5] K. J. Astrom, "Optimal control of Markov processes with incomplete state estimation," *Journal of Mathematical Analysis and Applications*, vol. 10, pp. 174–205, 1965.

[6] P. R. Kumar and P. Varaiya, *Stochastic Systems: Estimation, Identification and Adaptive Control.* Prentice Hall, 1986.

[7] R. D. Smallwood and E. J. Sondik, "The optimal control of partially observable Markov processes over a finite horizon," *Operations Research*, vol. 21, no. 5, pp. 1071–1088, 1973.

[8] J. L. Bander and C. C. White III, "Markov decision processes with noise-corrupted and delayed state observations," *Journal of Operational Research Society*, vol. 50, pp. 660–668, 1999.

[9] E. Altman, T. Basar, and R. Srikant, "Congestion control as a stochastic control problem with action delays," *Automatica*, vol. 12, pp. 1937–1950, 1999.

[10] K. V. Katsikopoulos and S. E. Engelbrecht, "Markov decision processes with delays and asynchronous cost collection," *IEEE Transactions on Automatic Control*, vol. 48, no. 4, pp. 568–574, 2003.

[11] M. Basin, J. Rodriguez-Gonzalez, and R. Martinez-Zuniga, “Optimal control for linear systems with time delay in control input based on the duality principle,” *Proceedings of the American Control Conference*, pp. 2144–2148, 2003.

[12] P. Varaiya and J. Walrand, “On delayed sharing patterns,” *IEEE Transactions on Automatic Control*, vol. 23, pp. 443–445, 1978.

[13] P. G. Voulgaris, “Optimal control of systems with delayed observation sharing patterns via input-output methods,” *Proceedings of the IEEE Conference on Decision and Control*, pp. 2311–2316, 2000.

[14] S. Adlakha, R. Madan, S. Lall, and A. Goldsmith, “Optimal control of distributed Markov decision processes with network delays,” *Proceedings of the IEEE Conference on Decision and Control*, pp. 3308–3314, 2007.

[15] S. Adlakha, S. Lall, and A. Goldsmith, “Information state for Markov decision processes with network delays,” *Proceedings of the IEEE Conference on Decision and Control*, pp. 3840–3847, 2008.

[16] G. E. Monahan, “A survey of partially observable Markov decision processes: Theory, models, and algorithms,” *Management Science*, vol. 28, no. 1, pp. 1–16, 1982.

[17] D. Bertsekas and S. Shreve, *Stochastic optimal control: The discrete time case.* Academic Press, 1978.

[18] D. Bertsekas, *Dynamic Programming and optimal control, volume 1.* Athena Scientific, 1995.

[19] S. Adlakha, S. Lall, and A. Goldsmith, “A Bayesian network approach to control of networked Markov decision processes,” *Proceedings of the Allerton Conference on Communication, Control, and Computing*, pp. 446–451, 2008.