Predicting the Evolution of Controlled Systems Modeled by Finite Markov Processes

Shuo Li and Matteo Pozzi

***Abstract*—The operation and maintenance of infrastructure components and systems can be modeled as a Markov process, partially or fully observable. Information about the current condition can be summarized by the "inner" state of a finite state controller. When a control policy is assigned, the stochastic evolution of the system is completely described by a Markov transition function. This article applies finite state Markov chain analyses to identify relevant features of the time evolution of a controlled system. We focus on assessing if some critical conditions are reachable (or if some actions will ever be taken), in identifying the probability of these critical events occurring within a time period, their expected time of occurrence, their long-term frequency, and the probability that some events occur before others. We present analytical methods based on linear algebra to address these questions, discuss their computational complexity and the structure of the solution. The analyses can be performed after a policy is selected for a Markov decision process (MDP) or a partially observable MDP. Their outcomes depend on the selected policy and examining these outcomes can provide the decision makers with deeper understanding of the consequences of following that policy, and may also suggest revising it.**

***Index Terms*—Partially observable (PO) Markov decision process (MDP), finite state controllers, finite state Markov chains, first passage time.**

I. Introduction

WE ADOPT a Markov chain analysis to predict relevant features of the condition evolution of controlled systems, with application to infrastructure components and systems controlled by policies for operation and maintenance (O&M). Finite state Markov chains (FSMC) are the most classic discrete-time, discrete-state stochastic process (when successive states of the process are not independent). Their theory and applications are reviewed in many textbooks [1]–[3], including classical textbooks of risk analysis for engineers [4], [5]. The analysis of FSMCs, using linear algebra methods, allows for identifying the long-term asymptotic properties of the process, and its short-term transient behavior, e.g., computing the mean and variance of the "first passage time," i.e., the number of steps to move from one state to another [6]–[8].

The problem of selecting decisions to control a process can be formulated as a Markov decision process (MDP) or as a partially observable MDP (POMDP), depending on the observability of the process state [9]. For MDPs, the optimal policy (i.e., that minimizing the expected long-term management cost) can be identified by dynamic programming [9]; for POMDPs, approximate optimal policies can be identified by point-based value iteration methods [10], or by methods based on finite state controllers (FSCs). FSCs assume that the policy is a function of a state variable defined on a finite discrete domain. This policy is described by a "policy graph," where the nodes represent the states, and a link between two nodes indicates that, after receiving a specific observation, the state evolves from one to the other [11]. The optimal policy of an MDP belongs to the family of FSCs, and many works [12]–[14] have developed methods for optimizing FSCs for POMDPs (e.g., imposing constraints on the number of nodes in the graph).

Markov models can describe the deterioration of infrastructure systems, and also their evolution under a O&M control policy, when maintenance actions are selected depending on the collected observations. They have been extensively applied to O&M problems, e.g., of bridges [15], of electric-power systems [16], [17], of road pavements [18]–[20], of structural systems [21], of coastal protection [22], and of wind mills [23], [24]. Both MPDs [18]–[20], [22], [25] and POMDPs [21], [23], [24], [26]–[29] have been adopted to optimize decisions.

This article does not focus on the identification of the optimal policy. We assume that a control policy has been selected for a MDP or a POMDP, and we focus on predicting the consequences of following it. In O&M problems, the failure of some components is usually associated to a high penalty cost, and the optimal policy finds the best tradeoff between maintenance costs and frequency of critical events. However, even when the controller adopts that optimal policy, it is not trivial to predict how the condition of the system evolves, in the short and in the long term, because the policy, which is a function mapping "states" into actions, is not directly informative about what to expect in the future. For example, we may ask "Can a failure event occur? And, if so, how frequently?" "What is the probability that a critical condition occurs within a specific time horizon? And that it occurs before a maintenance action is taken?" These questions can be approximately answered by analyzing Monte Carlo simulations (MCSs), but that approach may require high computational cost for investigating rare critical events with good accuracy.

Manuscript received May 1, 2020; revised November 12, 2020; accepted March 9, 2021. This work was supported by the National Science Foundation project under Grant CMMI #1663479, titled "From Future Learning to Current Action: Long-Term Sequential Infrastructure Planning under Uncertainty." Associate Editor: R. Pan. *(Corresponding author: Matteo Pozzi.)*

The authors are with the Department of Civil, and Environmental Engineering, Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: shuol2@andrew.cmu.edu; mpozzi@cmu.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TR.2021.3067595.

Digital Object Identifier 10.1109/TR.2021.3067595

Moreover, MCSs also require specific initial conditions, and their results are conditional to those conditions. Instead, we focus on exact methods for global analysis of all conditions, to predict the system evolution. We illustrate how to apply FSMC analysis, based on linear algebra, to FSCs, in fully observable environments, as MPDs, and in partially observable ones, as POMDPs. We discuss the properties of the solution and its computational complexity. Outcomes of these analyses can provide the decision makers with deeper understanding of the system evolution, and may even suggest revising the control policy.

The rest of this article is organized as follows. In Section II, we introduce fundamental properties of the FSMCs and graphs describing the process, we formulate the general problem in Section III and propose methods to predict the evolution features in FSMCs in Section IV. In Section V, we present some examples. Finally, Section VI concludes this article.

II. Markov Evolution Models

A. Evolution of the External State and of Observations

We assume that a decision maker, who we refer to as "agent," interacts with a stationary system, observing its state and taking actions at a sequence of time instants $\{t_0, t_1, \ldots, t_N\}$, where the interval between consecutive instants ($\Delta t = t_k - t_{k-1}$) is uniform, and N is infinity (unless specified below). At every instant t_k along the sequence, the system is in state $s_k \in S$, that we call the "external state" (to stress that it belongs to the physical world outside the controller), and the agent takes decision $a_k \in A$, where S and A are the domains of external states and of actions, respectively. Interacting with the system, for every $k \geq 1$, the agent receives observation $y_k \in Y$ at time t_k, where Y is the observation domain. We assume that the external state is such that its evolution is Markovian when the controlling actions are given. Such evolution can be defined by a stationary transition model $T(s, a, s') = \mathbb{P}[s_{k+1} = s' \mid s_k = s, a_k = a]$, and the relation between state and observation by an stationary emission model $E(s, a, y) = \mathbb{P}[y_k = y \mid s_k = s, a_{k-1} = a]$ (these definitions are appropriate if S and Y are discrete domains, as we will assume later, while these models need to be defined as density functions on continuous domains).

B. Finite State Controller

We focus on deterministic stationary policies, as those optimal in the infinite horizon case for both MDPs and POMPDs. The controller takes actions, as a reaction to the collected observations. It can be modeled by a "conditional plan," that is a set of functions $\{f_0, f_1, \ldots, f_N\}$ mapping a finite sequence of observations into an action: for every k, function $f_k : Y^k \to A$ maps the sequence of the first k observations into an action (initial function f_0 has no argument, and it directly assigns initial action a_0). Equivalently, the agent's current action can be modeled as depending on a current "inner state" of the controller, which evolves depending on observations. Such inner state identifies the agent's current knowledge about the system, summarizing past and current observations. At time t_k, we indicate this state as $h_k \in H$, where H is the domain of inner states. Now, the agent's behavior is defined by two functions: a policy function $a_k = \pi(h_k) : H \to A$, relating current inner state to current action, and an updating function $h_k = \eta(h_{k-1}, y_k) : H \times Y \to H$, relating previous inner state and current observation to current inner state. Functions π and η can be formulated as depending on time index k but, because we focus on stationary behaviors, we drop this dependence. The inner state framework can be related to the conditional plan, for example by assuming that the inner state h_k is just the collection of all observations $\{y_1, y_2, \ldots, y_k\}$, collected up the present time. Note that, in general, the sequence of inner states $\{h_0, h_1, \ldots, h_N\}$ is not Markovian, as the next inner state depends on the next observation, that is not independent of past inner states and actions (even conditional to the current ones). If the inner state domain H is a finite set, the agent is adopting a FSC.

C. Joint (Inner and External) States

While the evolution of the external states is Markovian when the control actions are given, it is not necessarily Markovian when the system is coupled with a controller who selects actions online, as the FSC does. The current action a_k, in facts, depends on the current inner state h_k, which is not independent of the past external state s_{k-1}, even conditional to the current external state s_k. However, we can define a joint state $w_k = \{s_k, h_k\} \in W = S \times H$, merging inner and external states, whose evolution is Markovian. To understand that, suffice is to note that, once current inner and external states are jointly assigned, current action a_k is given, next external state s_{k+1} and observation y_{k+1} are independent of the past, and so is the evolution of the inner states. Hence, sequence $\{w_0, w_1, \ldots, w_N\}$ is Markovian, and its evolution can be studied by tools developed for FSMCs.

D. Control Under Full Observability: Finite MDPs

A special case of the previous framework is when the external state is fully observable, because the initial external state s_0 is known to the agent, and the current observation is identical to the current external state: for every $k \geq 1$, $y_k = s_k$ (that is, for any action a, $E(s, a, y) = \mathbb{I}[s = y]$, where $\mathbb{I}[\cdot]$ is the indicator function). In this case, the external state is a sufficient statistic of the process, and the agent's optimal behavior is a function of that. This is the MDP case, reviewed in Appendix A. Hence, both inner and augmented state can be taken as identical to the external one ($w_k = h_k = s_k$), which we simply refer to as "the state," and the policy can be based on that: $a_k = \pi(s_k) : S \to A$. The state evolution is modeled by a Markovian transition function $T(s, s') = T(s, \pi(s), s') = \mathbb{P}[s_{k+1} = s' \mid s_k = s]$. For a finite process, the finite state set S is $\{1, 2, \ldots, |S|\}$, where $|S|$ is the cardinality of S. In that case, the transition function is described by $|S| \times |S|$ stochastic matrix $\mathbf{T}$, and the process follows a FSMC. Fig. 1(a) reports the influence diagram of the fully observable process.

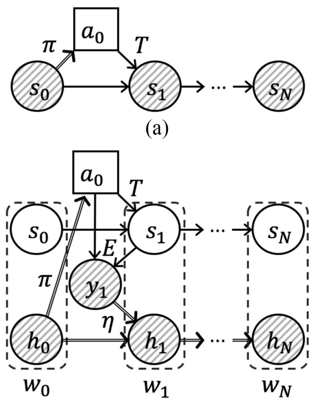

Fig. 1. Influence diagram for the fully observable process (a) and for the partially observable process (b). Shaded nodes refer to variables observable at current time.

E. Control Under Partial Observability: POMDPs

When observations are only partial or noisy, the agent cannot make her inner state identical to the external one (simply because the latter is unknown). This is the partially observable Markov decision process (POMDP) case, reviewed in Appendix VIII-B. When this is the case, if policy and updating functions for the inner state are given, the evolution of the joint state follows the transition function:

$$\begin{aligned} T(w, w') &= \mathbb{P}[w_{k+1} = w' \mid w_k = w] \\ &= T(s, \pi(h), s') \sum_y E(s,' \pi(h), y) \mathbb{I}[h' = \eta(h, y)] \end{aligned} \tag{1}$$

where $w = \{s, h\}$ and $w' = \{s,' h'\}$ are the current and the next joint state, respectively, which merge external and inner states. If H is a discrete set, so is W, of cardinality $|W| = |S||H|$, and the transition function can be described by a $|W| \times |W|$ stochastic matrix of a FSMC. Fig. 1(b) shows the influence diagram for the evolution of the controlled systems.

A key question is how to define the inner state under partial observations, and model its evolution. For a POMDP, the agent's optimal behavior is a function of the "belief," which is, at any time t_k, $|S|$-dimension stochastic (column) vector $\mathbf{b}_k \in \Omega_B$, where $\Omega_B = \{\mathbf{b} \in [0,1]^{|S|} : \mathbf{1}^\top \mathbf{b} = 1\}$ is the belief domain (and $\mathbf{1}$ is a column vector of ones). Component s of this vector is $b_k(s) = \mathbb{P}[s_k = s \mid y_1, \ldots, y_k, a_0, \ldots, a_{k-1}]$. Hence, one option is to identify the inner state (even the initial one) with the complete belief (i.e., $\forall k \geq 0 : h_k = \mathbf{b}_k$): the policy is, thus, a belief-to-action function, $a_k = \pi(\mathbf{b}_k)$, and component s' of the updating function $\boldsymbol{\eta}$, following Bayes' formula, is:

$$\begin{aligned} \eta_{s'}(\mathbf{b}, y) &= \mathbb{P}[s_{k+1} = s' | \mathbf{b}_k = \mathbf{b}, a_k = \pi(\mathbf{b}), y_{k+1} = y] \\ &= \frac{P(s,' y \mid \mathbf{b})}{\sum_{s'} P(s,' y \mid \mathbf{b})} \end{aligned} \tag{2}$$

where $P(s,' y \mid \mathbf{b}) = \sum_s b(s) T(s, \pi(\mathbf{b}), s') E(s,' \pi(\mathbf{b}), y)$, and the denominator corresponds to probability function $P(y \mid \mathbf{b})$.

Keeping the complete belief as an inner state presents two advantages. First, as noted previously, optimal policies for POMDPs are based on such belief. Second, the belief is a sufficient statistic, which follows a Markov process as, according to the updating rule of (2), next belief $\mathbf{b}_{k+1}$ is independent of past belief $\mathbf{b}_{k-1}$ conditional to current belief $\mathbf{b}_k$. The corresponding transition function, from current belief $\mathbf{b}$ to next belief $\mathbf{b}'$, is

$$T(\mathbf{b}, \mathbf{b}') = \sum_{s,' y} p(s,' y \mid \mathbf{b}) \mathbb{I}[\mathbf{b}' = \eta(\mathbf{b}, y)]. \tag{3}$$

However, a key challenge for this formulation is that, given that the cardinality of belief domain Ω_B is that of the continuum, function T cannot be exactly represented by any (finite or infinite) transition matrix. To obtain a FSC, it is necessary to limit domain H to a finite set, by assigning a rule for identifying the initial inner state, for updating it, and for assigning actions. These rules can be related to function $m(\mathbf{b}) : \Omega_B \rightarrow H$, mapping the belief into an inner state, so that the initial belief $\mathbf{b}_0$ is converted in initial inner state $h_0 = m(\mathbf{b}_0)$. Two strategies can be followed for defining such a map.

One strategy, that we call "grid method" (GM), is to discretize domain Ω_B in a grid of $|H|$ representative belief points: $\overline{\Omega}_B = \{\overline{\mathbf{b}}_1, \overline{\mathbf{b}}_2, \ldots, \overline{\mathbf{b}}_{|H|}\}$. Now, map $m(\mathbf{b}) = \text{argmin}_i d(\mathbf{b}, i)$ can be based on metric $d(\mathbf{b}, i) = \| \mathbf{b} - \overline{\mathbf{b}}_i \|$, measuring the distance between beliefs, for some appropriate norm $\| \cdot \|$. So, in this strategy, the inner state represents the closest representative belief in the grid. The policy depending on the inner state follows from the belief-to-action function: $\pi(h) = \pi(\overline{\mathbf{b}}_h)$. The updating function can be related to an approximated belief updating, using the nearest neighbor function: $\eta(h, y) = m(\boldsymbol{\eta}(\overline{\mathbf{b}}_h, y))$. These functions define the controller, which can be integrated with the evolution of the external state and of observations, to define the evolution of the joint state according to (1). However, given that the evolution of the complete belief is Markovian in itself, following this approach it is also possible to define an approximate Markovian evolution of the inner state, without any integration of the external one. The corresponding approximated transition probability is $T(i, j) = \sum_y e(i, y) \mathbb{I}[j = \eta(i, y)]$, where emission values $e(i, y) = p(y \mid \overline{\mathbf{b}}_i)$ are computed as in (2). These transition probabilities can be described by a $|H| \times |H|$ stochastic matrix. A generalization of this approach is reported in Appendix A-H.

An alternative strategy for discretizing the belief domain, that we call "optimal regions method" (ORM), is related to the special structure of optimal policies in POMDPs (see also Appendix A-B). Optimality is related to the minimization of the "value," i.e., the long-term expected discounted cost-to-go (actually, the word "value" has been originally defined as the long-term cumulative revenues to be maximized, but we relate it to cost minimization, following a recent work [22]). The optimal value function $V^*(\mathbf{b}) : \Omega_B \rightarrow \mathbb{R}$, is a concave piecewise linear function of the current belief $\mathbf{b}$, and it is the lower envelope of a set of hyperplanes, each defined by a $|S|$-dimensional (column) vector, called an "α-vector." The finite set $\Gamma = \{\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_{|H|}\}$ of $|H|$ α-vectors, where vector h is $\boldsymbol{\alpha}_h = [\alpha_h(1), \alpha_h(2), \ldots, \alpha_h(|S|)]^\top$, is the outcome of point-base value-iteration algorithms for approximately solving POMDPs [10], together with an action associated with each vector. We can relate each α-vector to an inner state, so that the inner state is h if vector $\boldsymbol{\alpha}_h$ dominates the others at the current belief. The policy function can now be taken as identical to the action associated with the corresponding vector. The formal map m between belief and inner state is now defined by an inner product

$$m(\mathbf{b}) = \underset{h\in H}{\operatorname{argmin}}(\boldsymbol{\alpha}_h^\top \mathbf{b}). \tag{4}$$

Using this map, belief domain Ω_B is partitioned in $|H|$ in convex regions, that we call "optimal regions," $\{D_1, D_2, \ldots, D_{|H|}\}$ (some of them could be empty), so that, for every $h \in H$, region D_h is defined as the part of Ω_B where vector $\boldsymbol{\alpha}_h$ dominates the others: $D_h \subseteq \Omega_B : \mathbf{b} \in D_h \Leftarrow h = m(\mathbf{b})$. Hence, the inner state is related to as a convex subset of the belief domain. The relation between α-vectors, inner state and conditional plans is outlined in Appendix VIII-C. This partitioning of the belief domain allows for an exact modeling of the stochastic process if the following condition is satisfied:

$$\forall\{\mathbf{b}, y\} \in (D_h \times Y) : \boldsymbol{\eta}(\mathbf{b}, y) \in D_{\eta(h,y)} \tag{5}$$

that is, if any two beliefs related to the same inner state h are updated into beliefs related to the same inner state η, under any specific observation y. However, the previous condition is not generally satisfied, as the available set of α-vectors Γ is usually incomplete, as it is the outcome of an approximate solver. The exact solution of an infinite horizon is usually related to an infinite number of α-vectors, that cannot be completely identified. When that set is approximated with a finite set Γ, condition in (5) is only approximately satisfied. Appendix A-D provides an algorithm to derive updating function η given set Γ.

F. Graphs Describing the State Evolution and the Policy

Graphs can represent the evolution of the external, inner and joint states, and they provide a useful and intuitive visualization tool. The evolution of the external state under full observability can be represented by a directed "transition graph," where nodes represent states, and links represent positive transition probabilities. State set S is also the node set, node s is linked to node s' only if $T(s, s')$ is positive, and this link is associated with that transition probability. So the graph completely describes the FSMC. The process starts at state/node s_0, randomly moves to one of the linked nodes, according to the probabilities encoded in the links, and the process evolves iteratively. We note that, given policy π under full observability, each node s is also associated with a specific action $a = \pi(s)$. Fig. 2(a) shows an example of transition graph, for $|S| = 3$. In that graph, if the process starts at state $s = 1$, the agent takes action $\pi(1) = 1$. The state stays at $s = 1$ with probability $T(1, 1)$ and evolves to $s = 2$ with probability $T(1, 2)$.

A FSC can be described by a "policy graph." In this graph, nodes represent inner states, and links the updating function η. Inner state set H is the node set, and each node h can be assigned with corresponding action $a = \pi(h)$, according to the selected policy. Node h is linked to node h' if exists an observation y

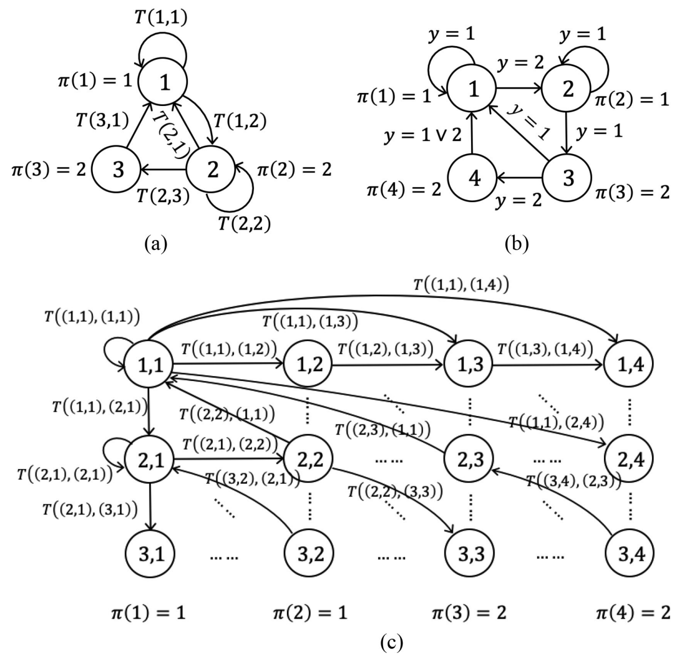

Fig. 2. Examples of transition graph (a). Policy graph (b). Joint graph (c).

is such that $h' = \eta(h, y)$. So up to $|Y|$ links leave each node, one per available observation (but it is possible that more than one observation link the same pair of nodes), and each link is associated with one (or more than one) available observation. The agent starts with an inner state/node h_0, executes the associate action $a_0 = \pi(h_0)$, receives observation y_1 and follows the corresponding link to get next inner state/node $h_1 = \eta(h_0, y_1)$, and the process evolves iteratively. The policy graph does not represent any Markov chain, as there is no probability associated with the links: the probability of receiving observation y cannot be analytically computed from inner state h (and executed action $\pi(h)$). Fig. 2(b) shows an example of policy graph with $|H| = 4$, $|A| = 2$ and $|Y| = 2$. For example, if the process starts at $h = 1$, the agent takes action $\pi(1) = 1$ and, if she receives observation $y = 2$, then the inner state moves to $h = 2$ and action $\pi(2) = 1$ is executed.

In the partially observable setting, the transition and the policy graphs can be merged in an "joint graph," i.e., a transition graph describing the evolution of the joint state as a FSMC. State set W is also the node set for this graph, and each node is related to an external state, an inner one, and an action (according to the adopted policy). Node w is linked to node w' if the transition probability $T(w, w')$ is positive. The process starts at node $w_0 = \{s_0, h_0\}$ and it evolves iteratively, following the transition probabilities on the links, so the graph completely encodes a FSMC. Differently from the fully observable case, the agent cannot identify the current node, as only the inner state is known to her, and the policy depends on that. Nonetheless, the analysis of the Markov graph can provide answers to relevant questions. Fig. 2(c) shows an example of joint transition graph (with the same values of $|S|$, $|H|$, $|A|$, and $|Y|$ of the previous examples): joint states are indicated by the pair (s, h) of external and inner states, respectively; the nodes in each row refer to the same external state, and those in each column to the same inner state (and, hence, to the same action). For example, if the process starts at joint state $\{2,1\}$, the agent takes action $\pi(1)=1$. The process moves to joint state $\{2,2\}$ with probability $T((2,1),(2,2))$ or to other states according to the probabilities of links leaving node $\{2,1\}$.

III. Problem Formulation

We consider FSCs and finite state physical systems, so that the condition evolution can be modeled by a FSMC. Among the state set (the external state, for the fully observable case, or the joint state, for the partially observable case), we identify some target, critical states and some other states potentially identifying the current condition.

In this context, we focus on a set of interconnected questions, related to the transition from some initial state to some critical ones. Basic questions are (i) Is state j reachable from state i? (ii) What are the communication classes of the process, and are those classes closed or open? (iii) What is the asymptotic probability of visiting state j? (iv) What is the expected number of steps for first visiting state j starting from state i? (v) What is the probability of visiting state j, starting from state i, within N steps? (vi) What is the probability of visiting state j before state l, starting from state i?

Questions (iv)–(vi) refer to a starting state i, and they are directly relevant if the current state is known, as in the fully observable case. If the current state is unknown, those questions can be rephrased as referring to a current belief $\mathbf{b}$. All questions can be generalized to a subset of target states J. Also, given that states are related to actions, previous questions can refer to actions; for example, a variation of question (iv) is: what is the number of steps before action a will be taken? We can formulate this question by identifying subset $J \ni j : a = \pi(j)$, where the policy assigns action a.

In the application to infrastructure systems and components, critical states may be those related to failures and malfunctioning, or the execution of important actions. For example, applications of previous questions are: Can a controlled component ever fail (i)? If so, what is the expected failure time (iv)? What is the probability of failure within 10 years (v)? What is the probability that failure will occur before replacement (vi)?

IV. Analysis of Finite State Markov Chains

A. General Notation and Framework

In this section, we present exact methods, based on linear algebra, for addressing the questions introduced previously. The Markov process can be related to the set of external states (in case of fully observability), of inner states (when the belief is modeled as a Markov process), or of the joint states (for the general POMDP case). We present the methods as related to a general FSMC with n states, in set $S=\{1,2,\ldots,n\}$, and transition function $T(i,j): S\times S \to [0,1]$, for every pair of states, described by $n\times n$ stochastic matrix $\mathbf{T}$. As the chain can be represented by a graph, in the following we refer indifferently to states or to nodes.

B. Topological Analysis

Some questions about the system dynamic can be related to the graph topology, and not to the specific positive values in the transition matrix; i.e., to the set of links, and not to the probabilities assigned to them. Let $L(i,j)=\mathbb{I}[T(i,j)>0]$ be a binary link function, so that $L(i,j)=1$ if node i is linked to node j, with any positive transition probability. Let $\mathbf{L}$ be the corresponding $n\times n$ matrix. Some long-term properties of the chain process are related to matrix $\mathbf{L}$, regardless of the specific value in $\mathbf{T}$.

The binary reachability function R is related to the link function L. The corresponding $n\times n$ matrix is $\mathbf{R}=\mathbb{I}[(\mathbf{I}+\mathbf{L})^{n-1}>0]$, where $\mathbf{I}$ is the identity matrix. If node j is reachable (in an arbitrary number of steps) from node i, then $R(i,j)$ is one, and it is zero otherwise. The binary communication function $Q(i,j)=R(i,j)R(j,i)$ is one only if nodes i and j "communicates," i.e., if they are reachable from one another. By analyzing the corresponding $n\times n$ symmetric matrix $\mathbf{Q}$, we can partition the node set in a set C of "communication classes," where two nodes belong to the same class only if they communicate with each other. If there is at least one node belonging to class $\mathscr{C}$, say node $i\in\mathscr{C}$, that can reach a node $j\notin\mathscr{C}$, outside that class, i.e., if $R(i,j)=1$ and $Q(i,j)=0$, then class $\mathscr{C}$ is open (toward the class j belongs to). Otherwise, class $\mathscr{C}$ is closed. For sure, at least one class in set C is closed. The classes can be arranged into a directed acyclic graph (DAG), where a node represents a class, and class node $\mathscr{A}$ is linked to node $\mathscr{B}$ if class $\mathscr{A}$ is open toward class $\mathscr{B}$. Closed classes are leaves of the DAG, as the corresponding nodes are not linked to any node. The class set and its DAG can be derived by analyzing matrices $\mathbf{R}$ and $\mathbf{Q}$, e.g., adopting classical algorithms for identifying strong connectivity (or strongly connected components) [30], [31].

Class analysis is relevant for predicting the long-term behavior of a Markov chain. Interestingly, even if the chain is fundamentally described by events that can happen (i.e., the transitions between nodes), one can identify events that must certainly happen in the long term (with probability one). For example, all nodes in closed classes are "recurrent": if the process starts in a node belonging to a closed class, it will visit all and only the states in that class, sooner, or later, and keep returning to them indefinitely (the set of nodes of every closed class forms a transition (sub)graph, i.e., the corresponding matrix is a stochastic one). All states belonging to open classes are "transient," and sooner or later they will stop recurring in the process. "Absorbing" states are a special case of recurrent ones: once the process hits an absorbing state, it will stay there forever. Every absorbing state i forms a closed class by itself, and $T(i,i)=1$. For example, a failure condition can be represented by an absorbing state if no remediation action is executed when a component fails.

If there is just one class, then that is necessarily closed, and the chain is "irreducible." If that chain is also a-periodic, then it is "ergodic." In an irreducible and a-aperiodic chain, we can identify a number of steps k, so that all nodes are reachable from all nodes in any number of steps abovementioned k. To check that the irreducible chain is a-periodic, one can compute the reachability matrix for k steps $\mathbf{R}(k) = \mathbb{I}[\mathbf{L}^k > 0]$, with $k = (n-1)^2 + 1$ [32]: if all its entries are one, then the chain is ergodic.

Computationally, topological analysis scales well with system size n, and many efficient algorithms for network analysis can be implemented [33]. Other traditional topological questions are: what is the shortest path between two nodes? Such question is related to the shortest time to visit one state from another. For applications to the control of systems and components, the topological analysis can answer questions as: is a condition ever reachable? Will an action ever be taken? It may turn out, for example, that a conservative controller prevents one condition from ever being visited, or an action from ever being taken. It is to be noted that outcomes of this analysis strongly depend on the details of the assumed chains. For example, link function $L(i,j)$ is one is $T(i,j)$ is positive, regardless of how small this value is. Hence, any approximation of (small) transition probabilities to zero has high potential impact on the analysis.

C. Long-Term and Asymptotic Behavior

While function T describes the transition probability in one step, the transition in k steps, T_k, can be derived by the Chapman–Kolmogorov rule: the corresponding $n \times n$ matrix $\mathbf{T}_k$ is equal to $\mathbf{T}^k$.

Given an ergodic FSMC, there exists a stochastic vector $\boldsymbol{\beta} = [\beta(1), \beta(2), \ldots, \beta(n)]$, defining the asymptotic distribution

$$\forall i, \quad \beta(j) = \lim_{k \to \infty} (\mathbf{T}^k)_{ij} \tag{6}$$

where $\beta(j)$ represents the long-term marginal probability of visiting state j, independently of the initial condition. Because of the ergodic theorem [34], the expected number of steps between consecutive visits of node i is $1/\beta(i)$. Vector $\boldsymbol{\beta}$ is a left eigenvector of the transition matrix: $\boldsymbol{\beta}\mathbf{T} = \boldsymbol{\beta}$, associated with a unitary eigenvalue, and it can be identified using algorithms for eigenvalue analysis. If the chain is not irreducible, each close class can be related to its own asymptotic distribution, while the asymptotic value assigned to all (transient) nodes in open classes is zero.

D. Transient and Equilibrium Analysis

Another important class of analyses is related to transient and equilibrium values along the chain. To assess these features, we can transform the FSMC into a Markov reward process [35]. To every node i in the graph, let us assign (virtual) cost $c(i)$, so that such cost will be incurred every time the process visits that state, and (virtual) terminal cost $v_0(i)$. Costs are discounted with factor γ at every step. Let us assume that the Markov process follows transition function $\tilde{T}$ (which can be seen as a variation of original function T, where all nodes in a selected subset are transformed into absorbing nodes), described by $n \times n$ stochastic matrix $\tilde{\mathbf{T}}$. For any state trajectory $\mathbf{s} = \{s_0, \ldots, s_k\}$ of $(k+1)$ steps, the cumulative discounted (virtual) cost is

$$G_k(\mathbf{s}) = \sum_{j=0}^{k-1} \gamma^j c(s_j) + \gamma^k v_0(s_k). \tag{7}$$

The corresponding (virtual) value for initial state $s_0 = i$, i.e., the expected discounted cumulated costs, is defined by taking the expectation on trajectories starting from that state

$$\forall i, \quad v_k(i) = \mathbb{E}_{\mathbf{s}|s_0=i} G_k(\mathbf{s}). \tag{8}$$

Node costs and values for each $k \geq 0$ are listed in n-dimensional vectors $\mathbf{c} = [c(1), c(2), \ldots, c(n)]^\top$ and $\mathbf{v}_k = [v_k(1), v_k(2), \ldots, v_k(n)]^\top$. The Bellman equation allows us to express the value for $(k+1)$ steps as a function of that for k steps. Node and system equations read

$$\begin{aligned} \forall i, k, \quad v_{k+1}(i) &= c(i) + \gamma \sum_j \tilde{T}(i,j)\, v_k(j) \\ &\to \quad \mathbf{v}_{k+1} = \mathbf{c} + \gamma\, \tilde{\mathbf{T}}\, \mathbf{v}_k. \end{aligned} \tag{9}$$

For every k, value vector $\mathbf{v}_k$ can be computed iteratively from the terminal values vector $\mathbf{v}_0$, by applying (9) for k times. The value vector can converge to asymptotic equilibrium $\mathbf{v} = [v(1), v(2), \ldots, v(n)]^\top$, for k going to infinity. If so, equilibrium values are defined via the Bellman recursive equation

$$\forall i, \quad v(i) = c(i) + \gamma \sum_j \tilde{T}(i,j)\, v(j) \quad \to \quad \mathbf{v} = \mathbf{c} + \gamma\, \tilde{\mathbf{T}}\, \mathbf{v} \tag{10}$$

and $\mathbf{v}$ is the solution of linear system $\mathbf{A}\mathbf{v} = \mathbf{c}$, with $\mathbf{A} = \mathbf{I} - \gamma\tilde{\mathbf{T}}$. Equation (10) is traditionally related to policy evaluation in dynamic programming [9]. When γ is one, matrix $\mathbf{A}$ is rank deficient, and the uniform vector becomes part of its null-space (generally, the cumulative cost can diverge, when the discount factor is unitary). In that case, the solution of (10) can be found imposing specific values to a subset of nodes. A way of solving this linear problem is reported in Section VIII-E.

While (9) and (10) allow for identifying the first moment of the cumulative cost G_k, it is also possible to compute higher moments of this variable, to better characterize its distribution. In Appendix A-F, we show how to compute higher moments, via similar methods of linear algebra.

This general framework has several applications. One is to assess the actual economic cumulated discounted costs, when actual costs are associated with the nodes. In the MDP and POMDP cases, costs are associated with each pair of external state and action (as detailed in Appendices A-A and A-B). When a policy is assigned (both for the fully and for the partially observable cases), these costs are also associated with nodes in the graph. Hence, the analysis allows us to compute the corresponding economic value for each node, each time horizon (reconstructing the set of α-vectors) and, assessing higher movements, also the variance and other features of the distribution.

Another application of the equilibrium analysis of (10) is related to the expected first passage time. For a close communication class modeled by function T, the expected number of steps needed to visit any target node from any starting node is finite. To compute the expected number of steps to visit any node in subset

J, we transform every node in J into an absorbing one, obtaining adjusted transition function $\tilde{T}$. We set γ equal to one, and get a corresponding matrix $\mathbf{A}$ with rank $n - |J|$. Clearly, the cost assigned to every node in J must be zero, otherwise the process will cumulate an infinite (i.e., diverging) value at those nodes. So we set $c(j) = 0$ for every $j \in J$, to get a feasible solution. To every other node $i \notin J$, we set a unitary step $c(i) = 1$, so the value is counting the steps along the process. Among the feasible solutions, the correct one is that with zero value $v(j) = 0$ at every node $j \in J$, given the interpretation we are giving to the value (i.e., for a process currently at node $j \in J$, the expected number of steps to reach set J is zero). An integrated method for assessing moments of first passage time proposed by Hunter [8] is reported in the Appendix A-G.

Yet another application is to assess the probability of visiting any state in subset J within K steps. To evaluate this, again, we turn every node $j \in J$ of the target subset into an absorbing node, obtaining function $\tilde{T}$. We initialize unitary value $v_0(j) = 1$ for every node $j \in J$ in the target subset, and zero value $v_0(i) = 0$ for every other node $i \notin J$. With zero immediate cost for all nodes, $\mathbf{c} = \mathbf{0}$, and unitary γ, we apply (9) iteratively for k times, to get vector $\mathbf{v}_k$, which represents the queried probabilities. Also, by computing the difference between $\mathbf{v}_k$ and $\mathbf{v}_{k-1}$, we get the probability of visiting subset J for the first time in exactly in k steps, from all nodes. For k going to infinity, $v_k(i)$ converges to the probability of ever visiting node subset J from node i. This probability is one if the original chain is ergodic. If it is not ergodic, applying (10), the asymptotic vector $\mathbf{v}$ belongs to the null space of $\mathbf{A}$ and it can be identified by imposing $v(j) = 1$ for each node $j \in J$ in the target subset.

The last application that we mention is the assessment of the probability of reaching some nodes before others. Given two disjoints sets of nodes X and Z, what is the probability that the process will visit any node in X before any node in Z? To assess this, we turn all nodes in X or in Z into absorbing nodes, getting function $\tilde{T}$. Again, costs are set to zero, $\mathbf{c} = \mathbf{0}$, and γ to one. Again, the solution $\mathbf{v}$ belongs to the null space of $\mathbf{A}$, and it can be identified by imposing $v(j) = 1$ for every node $j \in X$, and $v(j) = 0$ for every node $j \in Z$.

E. Structure of the Solution, as a Function of the Belief

Most topological, iterative, and equilibrium analyses are directly applicable to processes, where the current state is known to the agent. In the partially observable case, when the evolution is captured by the joint state, the agent knows only the current inner state, which represents a subset of the nodes in the joint graph. The iterative and the equilibrium analyses identify a $|W|$-dimensional vector $\mathbf{v}$ (where we have dropped subscript k, if present), which assigns a value to every joint state. As the inner state h is known, we can extract the $|S|$-dimensional subvector $\mathbf{v}_{(h)} = [v_{(h)}(1), v_{(h)}(2), \ldots, v_{(h)}(|S|)]^\top$, with $v_{(h)}(s) = v(w)$ and $w = \{s, h\}$. The expected (virtual) value associated with belief $\mathbf{b}$ is

$$V(\mathbf{b}) = \sum_s v_{(m(\mathbf{b}))}(s) b(s) = \mathbf{v}_{(m(\mathbf{b}))}^\top \mathbf{b}. \qquad (11)$$

TABLE I
LINK MATRIX UNDER THE OPTIMAL POLICY FOR THE MDP EXAMPLE 1

$$\mathbf{L} = \begin{bmatrix} 1&1&1&0&0&0&0&0\\ 1&1&1&1&0&0&0&0\\ 1&1&1&1&1&0&0&0\\ 1&1&1&1&1&0&0&0\\ 1&1&1&1&1&0&0&0\\ 1&1&1&1&1&0&0&0\\ 1&1&1&1&1&1&0&0\\ 1&1&1&1&1&1&1&0 \end{bmatrix}$$

In the ORM, every belief $\mathbf{b} \in D_h$ is mapped into the same inner state $h = m(\mathbf{b})$ and, hence, to the same value vector $\mathbf{v}_{(h)}$. Hence, (11) shows that the value is a linear function of the belief, for all belief in the region dominated by the same α-vector. However, for belief $\mathbf{b}$ at the border between regions, the map $m(\mathbf{b})$ is discontinuous, so is vector $\mathbf{v}_{(m(\mathbf{b}))}$ and so the expected value. For those beliefs at a border, a small perturbation of the belief changes the conditional plan followed by the controller (e.g., sometimes even the current action selected). Overall, the value function resulting from the analyses is nor concave nor continuous. As noted previously, the optimal value function for the POMDPs is continuous, concave and piecewise linear. The difference between the properties of these two value functions is due to the following reason. The optimal controller of a POMDP aims at minimizing the value itself, hence, at the border between two regions the value related to the corresponding two conditional plans must be the same, and the corresponding value function is continuous. Instead, the controller does not aim at minimizing any (virtual) values analyzed in previous section (e.g., the expected number of steps to reach a subset of nodes). However, the two value functions share a common property: they are both linear inside each optimal belief region (i.e., in each region dominated by an alpha vector).

V. EXAMPLES OF ANALYSES

A. MDP Example 1: Topological Analysis

We provide some examples of analysis. We start by analyzing an MDP example, taken from the context of pavement management [36]. A pavement segment condition is discretized into $n = 8$ states: state 1 indicates an *intact*, smooth pavement, a higher state indicates a higher damage level up to state 8, which indicates a failure condition. Time is discretized in years, and more details on the example are reported in the Appendix X-A. The optimal policy (which minimizes the discounted expected long-term cost) prescribes to repair the segment, with different levels of overlay, in states from 1 to 5 and to replace the pavement in states from 6 to 8, but it is not directly informative on the evolution of the controlled process. We can investigate, for example, if the failure is ever possible, when the management process starts with an intact segment. By applying the topological analysis of Section IV-B, we derive the link matrix shown in Table I. The corresponding communication classes are arranged as in the DAG reported in Fig. 3: states from 1 to 5 form a closed class, while each state from 6 to 8 forms an individual open class. Hence, states from 6 to 8 are transient, and they cannot be

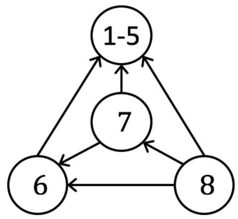

Fig. 3. DAG of the communication classes in MDP example 1.

TABLE II
TRANSITION MATRIX FOR THE MDP EXAMPLE 2

$$\mathbf{T} = \begin{bmatrix} 0.90 & 0.05 & 0.02 & 0.02 & 0.010 & 0 \\ 0.40 & 0.32 & 0.10 & 0.10 & 0.078 & 0.002 \\ 0.40 & 0.32 & 0.10 & 0.10 & 0.078 & 0.002 \\ 0.40 & 0.32 & 0.10 & 0.10 & 0.078 & 0.002 \\ 0.90 & 0.05 & 0.02 & 0.02 & 0.010 & 0 \\ 0.90 & 0.05 & 0.02 & 0.02 & 0.010 & 0 \end{bmatrix}$$

TABLE III
MEAN AND STANDARD DEVIATION OF FIRST PASSAGE TIME IN YEARS FOR THE MDP EXAMPLE 2

$$\mathbf{M} = \begin{bmatrix} 1.22 & 15.9 & 34.9 & 34.9 & 48.2 & 3177 \\ 2.29 & 10.8 & 30.6 & 30.6 & 42.5 & 3166 \\ 2.29 & 10.8 & 30.6 & 30.6 & 42.5 & 3166 \\ 2.29 & 10.8 & 30.6 & 30.6 & 42.5 & 3166 \\ 1.22 & 15.9 & 34.9 & 34.9 & 48.2 & 3177 \\ 1.22 & 15.9 & 34.9 & 34.9 & 48.2 & 3177 \end{bmatrix}$$

$$\mathbf{S} = \begin{bmatrix} 0.822 & 15.1 & 34.0 & 34.0 & 46.8 & 3166 \\ 1.600 & 14.1 & 33.6 & 33.6 & 46.4 & 3166 \\ 1.600 & 14.1 & 33.6 & 33.6 & 46.4 & 3166 \\ 1.600 & 14.1 & 33.6 & 33.6 & 46.4 & 3166 \\ 0.822 & 15.1 & 34.0 & 34.0 & 46.8 & 3166 \\ 0.822 & 15.1 & 34.0 & 34.0 & 46.8 & 3166 \end{bmatrix}$$

reached from any of the first 5 states: if the process starts from any of the first 5 states, no state above 5 will ever be visited, and hence, no replacement action will be taken, following that conservative policy.

B. MDP Example 2: Equilibrium Analysis of a Single Component

As a second example, also inspired by the same deterioration model [36], let us consider a component with $n = 6$ states: states $1-5$ refer to five levels of deterioration, from *intact* to *severely damaged*, and state 6 to a critical *failure* condition. The three available actions are doing-nothing (action 1), performing minor repair (action 2), and replacing the component (action 3). The details of the example are reported in Appendix B-B.

By solving that MDP, we obtain vector $[1,\ 2,\ 2,\ 2,\ 3,\ 3]$, representing the optimal policy π for all states from 1 to 6. This policy leads to the minimum discounted expected long-term cost, but it is not directly informative on the evolution of the controlled process. The corresponding transition matrix under this policy is shown in Table II. The FSMC is ergodic, and Table III shows the matrix of mean visiting time $\mathbf{M}$, as defined in Appendix A-G, in years, so that entry $\mu(i,j)$ is the expected

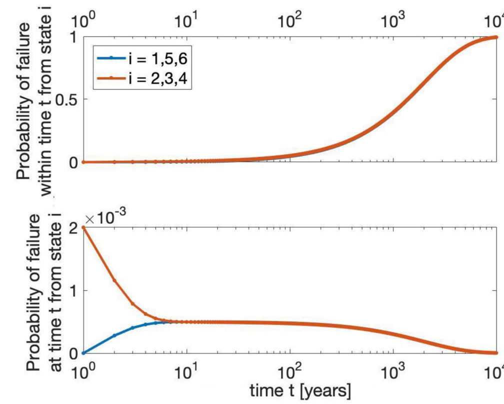

Fig. 4. Probability of transition time to failure for the MDP example 2.

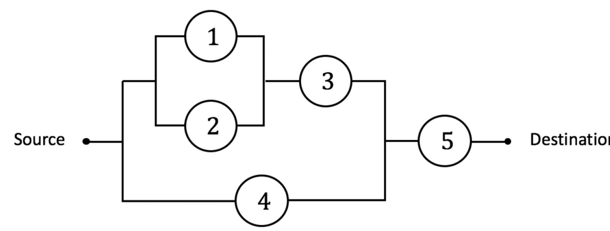

Fig. 5. Block diagram of the system for the MDP example 3.

number of years for visiting state j from state i (or returning to the same state, if $i = j$). We note that the expected time to first reach the failure state is much higher than that to first reach any other state, from any initial state. Matrix $\mathbf{S}$ is also reported in the table, whose entries are the standard deviation, in years, of the visiting or the returning times, and we note that most corresponding coefficients of variation are around 1 (as it is for exponentially distributed visiting times).

Fig. 4 illustrates the probability of visiting the failure state from any state, as a cumulative distribution function (a) and as a probability mass function (b). The probability distributions starting from states 1,5,6 are identical, because the policy assigns the *replacement* action to states 5 and 6, and the evolution after such an action is identical of that of an *intact* component. Similarly, from states 2,3,4, the same action *minor repair* is taken, and the evolution follows the same law. For example, the probability of failure within 20 years starting from the intact state is 5.4×10^{-3}. Another relevant question is about the fate of the component: what is the probability of a failure occurring before the replacement occurs (i.e., of visiting state 6 before state 5)? Applying the method presented in Section IV-D, this probability is 1.5% from the intact state, and 1.67% from states 2, 3, or 4.

C. MDP Example 3: A 5 Component System

The third example refers to a parallel-series system shown, as a block diagram, in Fig. 5, inspired by a recent work [37]. The system consists of 5 identical and independent components, each with 4 possible condition states: state 1, 2, 3, and 4 refers to the

TABLE IV
TRANSITION MATRICES FOR THE MDP EXAMPLE 3

$$\mathbf{T}_1 = \begin{bmatrix} 0.92 & 0.04 & 0.03 & 0.01 \\ 0 & 0.92 & 0.05 & 0.03 \\ 0 & 0 & 0.70 & 0.30 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{T}_2 = \begin{bmatrix} 0.92 & 0.04 & 0.03 & 0.01 \\ 0.92 & 0.04 & 0.03 & 0.01 \\ 0.92 & 0.04 & 0.03 & 0.01 \\ 0.92 & 0.04 & 0.03 & 0.01 \end{bmatrix}$$

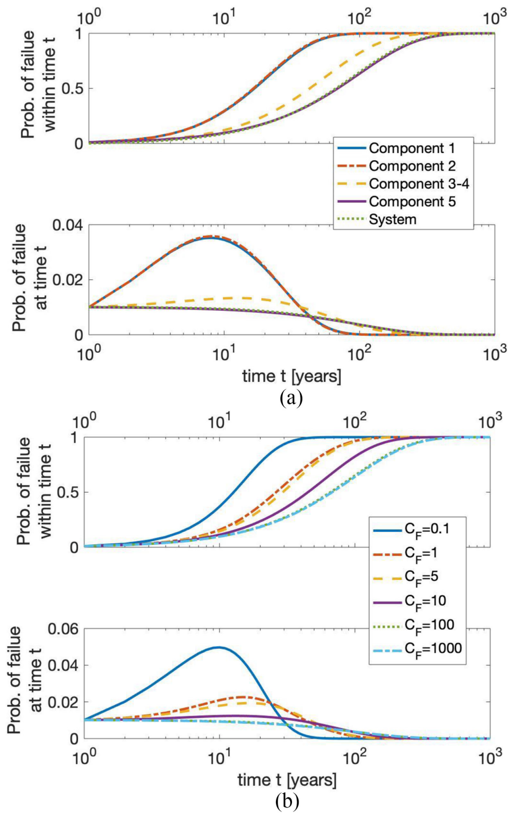

Fig. 6. Probability of transition time to failure for MDP example 3.

intact, *minor damaged*, *major damaged,* and *failure* condition of the component, respectively. For each component, the agent selects annually one among two possible actions: action 1 and 2 indicate to *do-nothing* and to *repair* the component, respectively. The system fails when, because of the failure of a subset of components, the source is not linked to the destination. The evolution of each component follows the transition matrixes defined in Table IV (matrix $\mathbf{T}_i$ refers to action i).

The agent adopts the optimal policy assuming that the cost for system failure is $C_F = 100$ times that for repairing a single component, and that the one-step discount factor γ is 0.95. Fig. 6(a) shows the probability of transition time from the configuration where all components are intact to the failure of individual components and of the system. Components 1 and 2 have almost identical distributions, because their topological role in the system is the same. Components 3 and 4 also have identical distributions, despite their different topological role. Component 5 is the least likely to fail, because it is often repaired given its critical role in the system. In Table V, we list the asymptotic probability of each component being in each

TABLE V
ASYMPTOTIC DISTRIBUTIONS

Component	State	Probability
1	1	0.3499
	2	0.1902
	3	0.0676
	4	0.3922
2	1	0.2615
	2	0.1421
	3	0.0521
	4	0.5443
3	1	0.6133
	2	0.3333
	3	0.0367
	4	0.0167
4	1	0.6133
	2	0.3333
	3	0.0367
	4	0.0167
5	1	0.9200
	2	0.0400
	3	0.0300
	4	0.0100

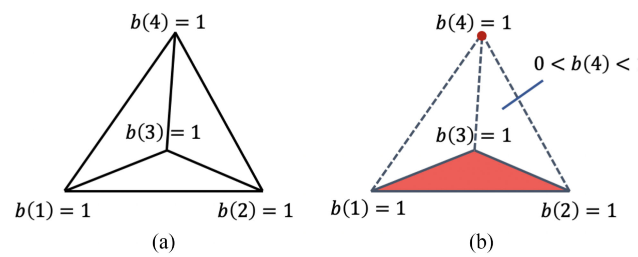

Fig. 7. Belief domain for the POMDP example (a) and reduced subset with perfect knowledge of the failure (b).

condition state. Components 1 and 2 are often in the failure condition, while components 3 and 4 are rarely in that condition. The policy prescribes to repair component 5 when it is damaged (or failed), and so its asymptotic distribution is identical to the first row of the transition matrices (of both actions).

Fig. 6(b) reports the outcomes of a parametric analysis, when the system failure cost C_F varies from 0.1 to 1000 (of the cost for repairing a single component), in terms of probability of time to system failure from all intact components. As C_F increases, the policy becomes more conservative, and the probability of failure decreases. However, when the cost of failure is larger than 100, the optimal policy is almost insensitive to C_F, and so is the probability of time to failure.

D. POMDP Example

We now move to a partially observable context, where the inner state differs from the external one. We consider an infrastructure component with four states: *intact* ($s = 1$), *minor damaged* ($s = 2$), *major damaged* ($s = 3$), and *failure* ($s = 4$). The belief domain Ω_B is represented by a tetrahedron, as shown in Fig. 7(a). Each vertex indicates perfect knowledge of the external state. Four maintenance actions are available: to *do nothing* (action 1), *inspect* (action 2), *repair* (action 3), and *replace* (action 4) the component. The cost of repairing, of

TABLE VI
TRANSITION AND EMISSION MATRICES FOR THE POMDP EXAMPLE

$$\mathbf{T}_{1,2} = \begin{bmatrix} 0.983 & 0.015 & 0.002 & 0 \\ 0 & 0.719 & 0.280 & 0.001 \\ 0 & 0 & 0.650 & 0.350 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{T}_3 = \begin{bmatrix} 0.9830 & 0.0150 & 0.0020 & 0 \\ 0.3200 & 0.5595 & 0.1200 & 0.0005 \\ 0.1600 & 0.2000 & 0.5200 & 0.1200 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{T}_4 = \begin{bmatrix} 0.983 & 0.015 & 0.002 & 0 \\ 0.983 & 0.015 & 0.002 & 0 \\ 0.983 & 0.015 & 0.002 & 0 \\ 0.983 & 0.015 & 0.002 & 0 \end{bmatrix}$$

$$\mathbf{E}_{1,3,4} = \begin{bmatrix} 1/3 & 1/3 & 1/3 & 0 \\ 1/3 & 1/3 & 1/3 & 0 \\ 1/3 & 1/3 & 1/3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{E}_2 = \begin{bmatrix} 1-\varepsilon & 2\varepsilon/3 & \varepsilon/3 & 0 \\ \varepsilon/2 & 1-\varepsilon & \varepsilon/2 & 0 \\ \varepsilon/3 & 2\varepsilon/3 & 1-\varepsilon & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

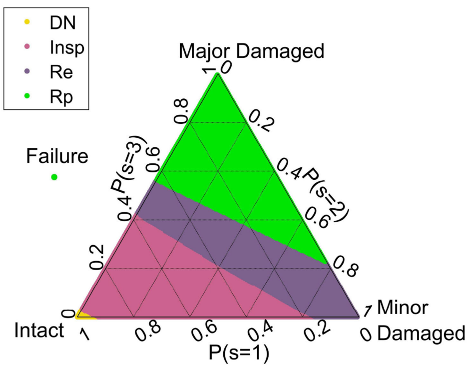

Fig. 8. Optimal policy for the POMDP example.

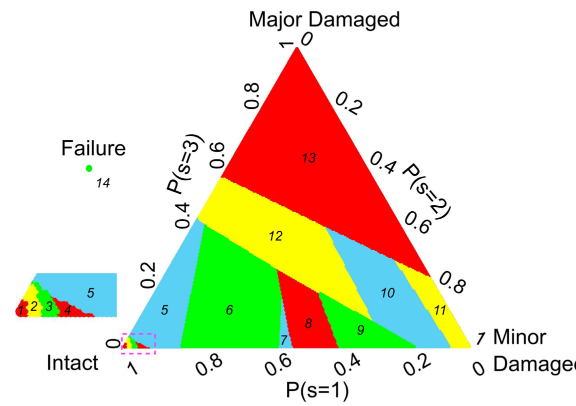

Fig. 9. Belief optimal regions partitioning domain Ω_B.

TABLE VII
ASYMPTOTIC DISTRIBUTIONS FOR BELIEF REGIONS

Inner state h	1	2	3	4	5
Probability	0.2365	0.1658	0.2358	0.1228	0.1811
Inner state h	6	7	8	9	10
Probability	0.0036	0	0.0023	0.0192	0.0014
Inner state h	11	12	13	14	
Probability	0.0053	0.0078	0.0145	0.0039	

replacing and of failure is 6, 35, 90 times that of inspecting, respectively, and the annual discounted factor γ is 0.95. The annual transition and emission matrices are shown in Table VI. In the emission matrix, parameter ε models the inaccuracy of the inspections, and it is set to 0.05 (matrix $\mathbf{E}_i$ refers to action i). As the failure state is perfectly observable, the fourth entry of the belief vector is either zero or one: in the former case, the agent knows that the component is not failed, in the latter case, she knows it is for sure. The part of the belief domain with nil probability of failure can be represented by a triangle, where vertexes indicate perfect knowledge on the external state (while the perfect knowledge of failure is represented by a single point), as shown in Fig. 7(b).

The optimal policy is identified by the successive approximations of the reachable space under optimal policies [38] method, as implemented in the R library for solving POMDPs [39]. The corresponding value function is made up by 17 α-vectors, but only 13 of them are not dominated in the subdomain with perfect knowledge of failure. Following the ORM, we derive a policy graph modeling the behavior of the agent. We isolate the (perfect knowledge of the) failure as a separate node in the policy graph (i.e., an additional inner state), so that the policy graph is made up by $13 + 1 = 14$ nodes. Fig. 8 shows the optimal policy in the subdomain with perfect knowledge of no failure, and Fig. 9 shows the optimal regions $\{D_1, \ldots, D_{13}\}$, each dominated by an α-vector, while the failure is indicated by inner state 14, as a separate point. All the following analyses are based on the ORM.

Fig. 10(a)–(c) illustrates the asymptotic distributions of the joint state, when the external state is $s = 1$, $s = 2$, and $s = 3$, respectively. The failure state, $s = 4$, is visited with probability 3.30×10^{-3}. If the agent had perfect knowledge of the external state, her belief would be concentrated in a vertex of the triangle (or in the point indicating the failure). The graphs in Fig. 10 show the impact of partial and imperfect observations: the belief tends to be close to the corresponding vertex, but it is likely that a *minor damage* condition can be related to a strong belief that the external state is *intact*. Graph (d) illustrates the marginal probability of the inner state. The same probability distribution is also reported in Table VII. The first five inner states have a cumulative annual probability of about 82%. The joint transition graph has one close communication class, and three open classes related to inner state 7, that receive zero asymptotic probability. As reported in the table, the asymptotic annual probability of failure is 0.39%.

The probabilities of failure within 20 years and within 50 years are represented in Fig. 11(a) and (b), respectively. The probability ranges from 6.03% to 22.35% within 20 years, and from 15.88% to 30.5% within 50 years. Fig. 12(a) shows the expected time to failure in years, ranging from 227 to 274 years. Graph (b) focuses on the segment in the subdomain where the probabilities of states 2 and 3 are equal, so that the belief vector has form $\mathbf{b} = [1 - d,\ d/2,\ d/2,\ 0]^\top$, where $d \in [0, 1]$ is the probability of damage (minor or major). That subdomain is indicated with a red segment in graph (a), and it is also used in the following two figures. When this probability is high (above 0.78) the component is replaced, so the time to failure is higher. As we have noted in Section IV-E, the outcome is a discontinuous function.

Fig. 13 reports the corresponding expected time to replace the component. In the subdomain where the probabilities of external states 2 and 3 are equal, this probability ranges from 37 to 62 years until the component is immediately replaced.

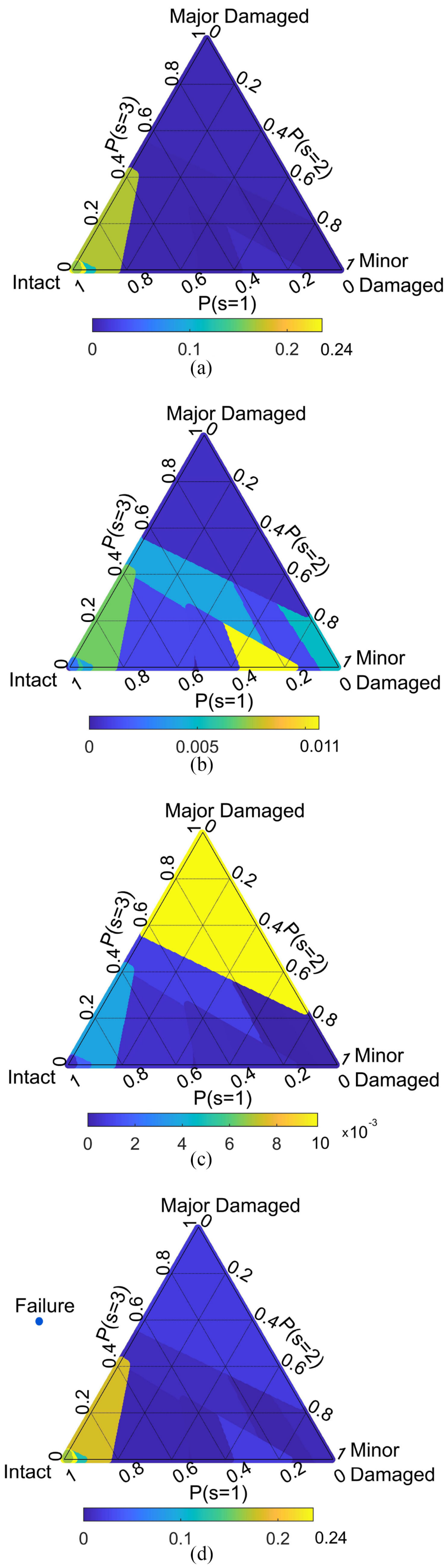

Fig. 10. Asymptotic distribution β for the joint state, when the external state is (a) intact, (b) minor damaged, (c) major damaged, and marginal distribution of the inner state (d).

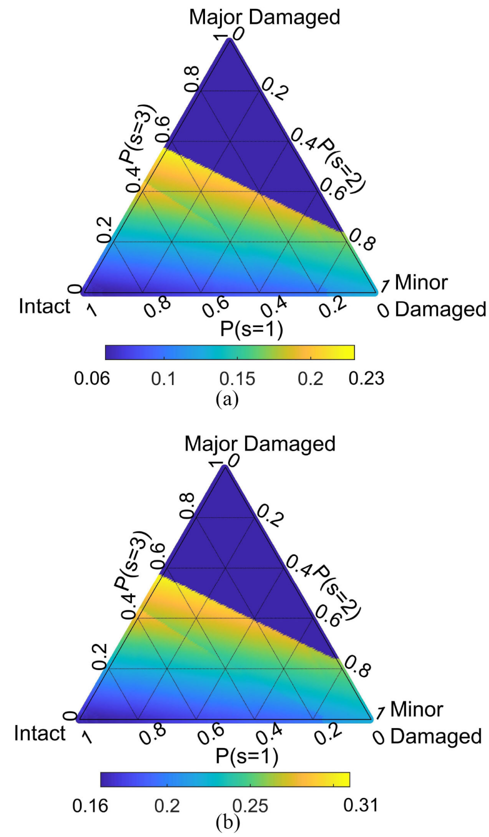

Fig. 11. Probability of failure within 20 (a) and 50 (b) years for POMDP example.

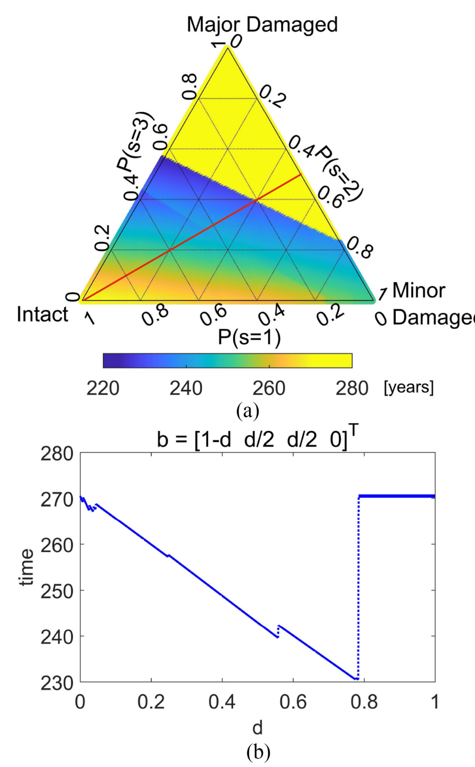

Fig. 12. Expected time to failure in the belief domain (a) and in a linear subset in Ω_B (b).

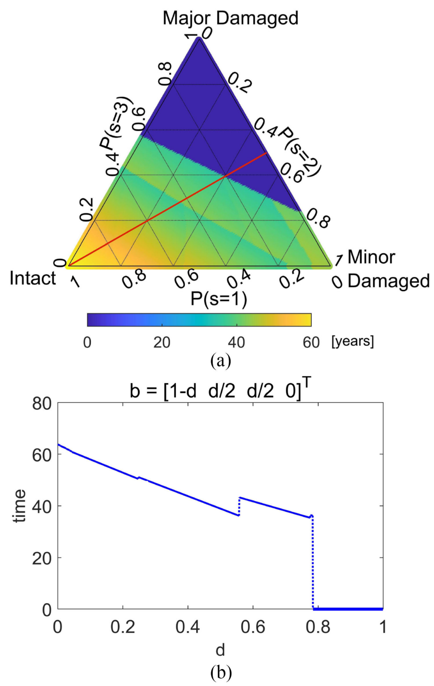

Fig. 13. Expected time to replace in the belief domain (a) and in a linear subset in Ω_B (b).

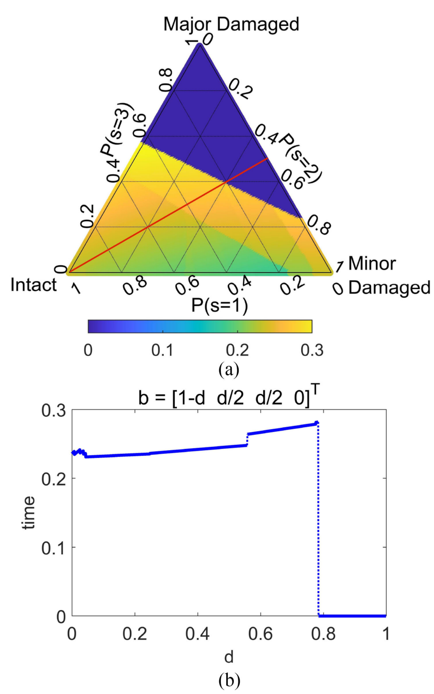

Fig. 14. Probability of failure before replace in the belief domain (a) and in a linear subset in Ω_B (b).

Fig. 14 shows the probability of failure before replacing the component, that is around 25% in the same subdomain, before the component is replaced.

1) Comparison Among Numerical Approaches: The outcomes of the previous analyses, obtained by ORM, can be

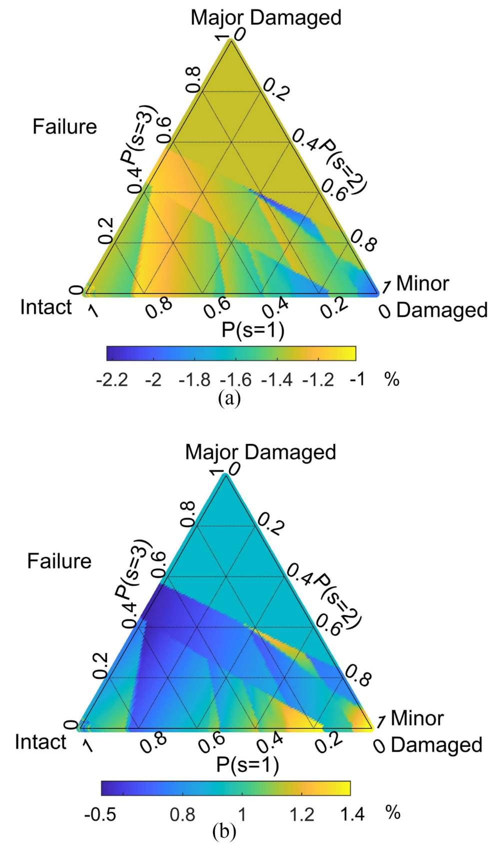

Fig. 15. Relative error comparing predictions of ORM and GM for the POMDP example: Expected time to failure (a), probability of failure within 20 years (b).

compered with those of other methods. Fig. 15 compares them with those obtained by the GM introduced in Section II-E, discretizing the belief domain into 20,100 points. The relative error, defined as $(g_a - g_b)/g_a$, where g_a and g_b represent the values obtained by ORM and GM, respectively, is shown in graphs (a) for the expected time to failure, in graph (b) for the probability of failure in 20 years this relative error is mostly in the order of $1 - 2\%$.

We also compare the predictions with MCSs, starting from perfect knowledge of the intact state. Two approaches can be adopted for performing MCSs. If we assume that agent always follows the policy graph, then ORM is exact. So the estimator based on MCSs following the policy graph converges to the outcome of ORM. However, we can also generate MCSs by assuming that the agent follows the optimal policy $\pi(\mathbf{b})$ defined by the set Γ of α-vectors, and the updating function $\boldsymbol{\eta}(\mathbf{b})$ defined in (2). In this approach, basing on 10 000 000 simulations, we obtain a 95%-confidence interval of [294, 300] years for the expected time to failure. The corresponding result is around 279 years for ORM and it is 288 years for GM. These discrepancies show that neither method can generally claim to exactly represent functions and $\pi(\mathbf{b})$ and $\boldsymbol{\eta}(\mathbf{b})$, in the continuous belief domain. While ORM should represent exactly these functions if set Γ was complete, it only approximately represents them when the set is incomplete, as in the presented analysis, and

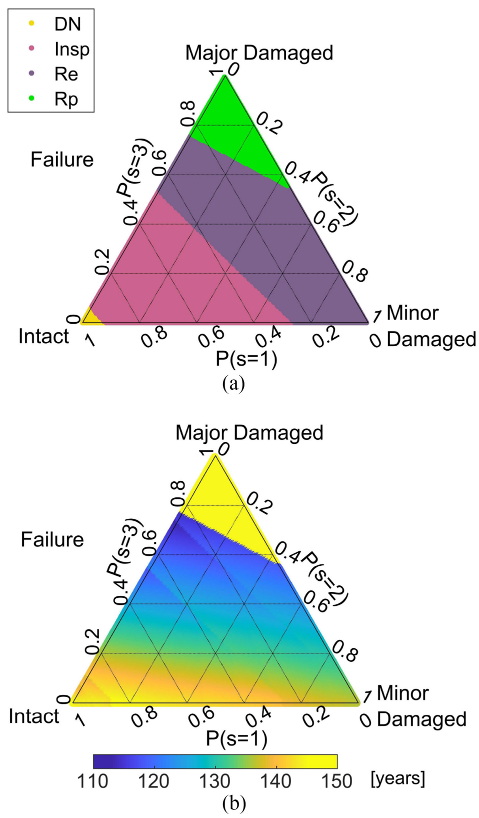

Fig. 16. Optimal policy (a) and expected time to failure (b) with low failure cost (i.e., when $C_F = 45$).

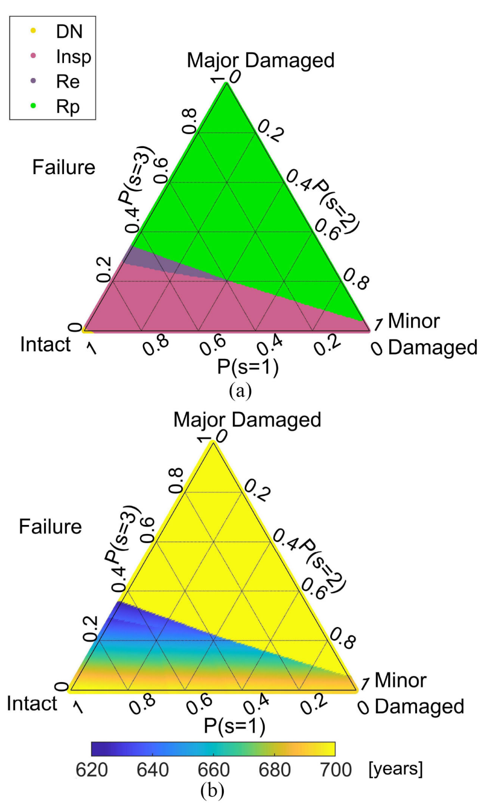

Fig. 17. Optimal policy (a) and expected time to failure (b) with high failure costs (i.e., when $C_F = 200$).

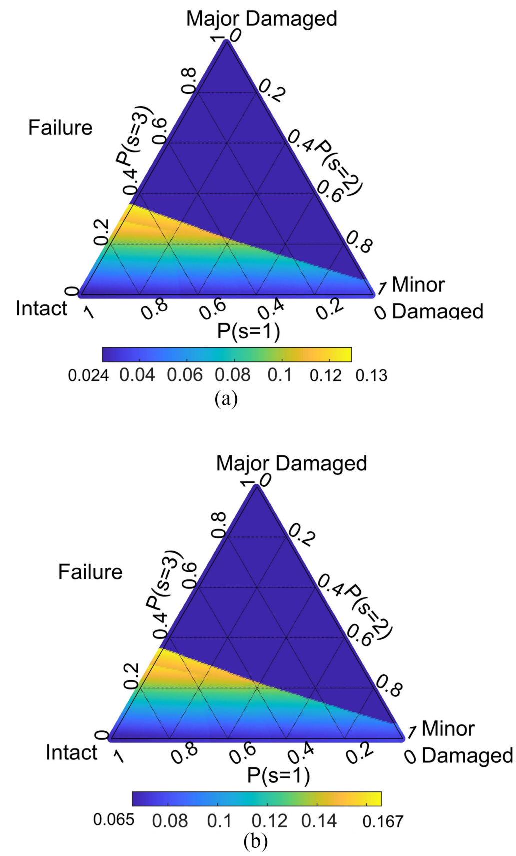

Fig. 18. Probability of failure within 20 (a) and 50 (b) years for POMDP example when $C_F = 200$.

this explains the bias. How to predict this discrepancy (without running MCSs) remains an open question. However, we note the following. The ORM assumes that the agent follows the policy graph identified, as outlined in Section II-E. That policy graph models a complete behavior, and the outcomes of ORM are exact, if the agent follows that behavior, as stated previously. Such control policy can be suboptimal, compared to policy $\pi(\mathbf{b})$. In Appendix B-C, we compare the value function obtained by the two approaches, to show that the policy modeled by the policy graph is almost optimal. So, in summary, the results based on ORM predict exactly the system evolution when the controller follows the policy represented by the graph which is, in this analysis, almost optimal.

2) Variations of the Original Setting: The optimal policy is highly sensitive to the failure cost. If the failure cost is reduced to one half (i.e., to 45 of the inspection cost), the policy is less conservative, as shown in Fig. 16(a). Graph (b) illustrates the corresponding expected time to failure, which ranges from 110 to 150 years. If, instead, the cost increases to 200 times the inspection cost, the policy becomes more conservative, as illustrated in Fig. 17(a) and the expected time to failure, ranging from 640 to 740 years, is shown in Fig. 17(b). The corresponding probability of failure, within 20 and 50 years, is shown in Fig. 18.

We explicitly investigate the impact of the failure cost C_F and of the inaccuracy of inspection ε in the prediction. We vary

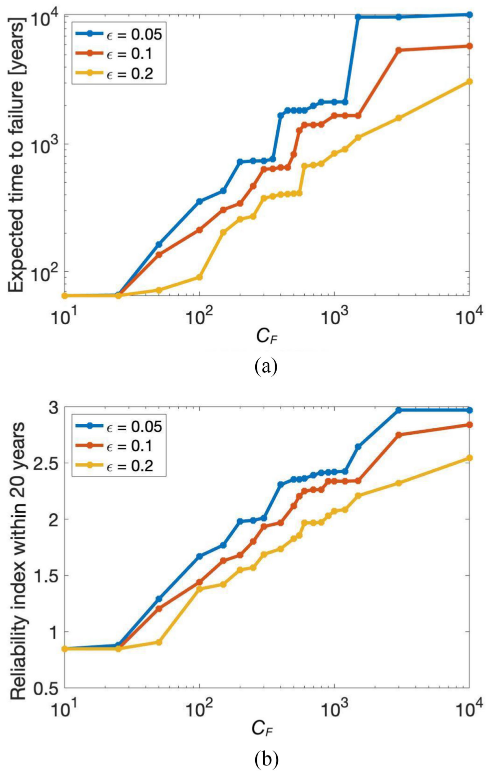

Fig. 19. Parametric analysis. (a) Expected time to failure. (b) Reliability index within 20 years.

C_F from 0 to 3,000 (intended, as abovementioned, as a multiplicative factor of the cost of inspection), and we assume ε to be 5%, 10%, or 20%. These parameters affect the optimal policy and, in turn, the predictions. We plot the resulting expected time to failure and the reliability index (which describes the probability of failure) within 20 years in Fig. 19. The expected time to failure increases and the reliability index decreases with the cost of failure, as the optimal policy becomes more and more conservative. Similarly, more precise observations allow for higher expected time to failure and reliability.

The number of α-vectors in set Γ (which is the cardinality of the set of inner states) depends on the parameter settings. For example, when C_F is 400 and the inaccuracy ε is 20%, the optimal policy is described by 3429 α-vectors, so that the cardinality of the joint states is 13 716. For this setting, the equilibrium analysis is solved in about 12 min on a server using 2 Intel Xeon E5-2690 v4 CPUs at 2.60 GHz, on MathWorks MATLAB.

VI. Conclusion

We illustrated how methods based on linear algebra for analyzing FSMCs can be adopted for predicting the future condition of systems, if their control is modeled by a MDPs or a POMDP. These predictions allow the agent to better understand the consequences of the adopted policy and, possibly, they may also suggest revising this policy.

These methods are based on the recursive properties of FSMCs, and are exact under the assumption that the agent adopts a FSC. This assumption is not restrictive for fully observable settings (if the domain of external states is finite), but it is for partially observable ones. The policy represented by a set Γ of α-vectors can only by approximately represented by a FSC, for example via the ORM we presented. However, the policy related to the FSC can be evaluated and compared with the representation based on Γ. If the two value function are approximately identical, we posit that the predictions based on the FSC are relevant and useful, as we can assume that the agent indeed follows the FSC (so that the predicting methods are exact). The general question of estimating the bias induced by the FSC assumption, for predicting the evolution of a system controlled by the policy represented by Γ, is still open. Similarly, the problem of assessing the accuracy of the policy represented by Γ in representing the actual optimal policy, depending on the number of $\alpha-$vectors, is still open.

Compared with methods based on MCSs, the proposed ones have the advantage of being exact and global, and they are particularly effective for estimating higher order moments or rare events.

The accuracy of the GM suffers from the "curse of dimensionality," as the number of representative points in the grid should grow exponentially with the cardinality of the set of external states, to keep the accuracy constant. Instead, the accuracy of ORM is not direct related to that cardinality, as it is more related to that of Γ and, thus, seems less subject to that curse.

We conclude this summary of ORM noting that we have derived the method starting from a set Γ, hence, from a description related to the optimal control with respect to some cost function (i.e., from the optimal regions). Also, we considered deterministic policies, function of the inner state. The generalization to some classes or random policies, where each inner state is associated with a distribution of actions, would be straightforward. Instead, the generalization to the method for arbitrary control policies was not examined in this article.

While our approach assumes an exhaustive enumeration of all inner and external states, future studies can be conducted to investigate the computational benefit of limiting the analysis to a smaller subset of "reachable" states, following the principles of many point-based value iteration solvers. This path, being less prone to the curse of dimensionality, could approximate predictions for larger systems.

Appendix A

Optimal Sequential Decision Making

A. Markov Decision Processes

In sequential decision making, an agent selects a sequence of actions, paying periodic costs and receiving observations from the system she is interacting with, with the aim of minimizing the long-term expected maintenance costs, i.e., the "value."

A MDP is a discrete time stochastic control process, where the state is fully observable. Formally, a MDP is defined by a 5-tuple: the finite set of states $S = \{1, 2, \ldots, |S|\}$, the finite set of actions $A = \{1, 2, \ldots, |A|\}$, the one-step transition function $T(s, a, s') : S \times A \times S \to [0, 1]$, the cost function $C(s, a) : S \times A \to \mathbb{R}$, and the discount factor $\gamma \in [0, 1]$. At every time step k, the system is in some state $s_k \in S$, the agent chooses action $a_k \in A$ and pays cost $C_k = C(s_k, a_k)$. The state evolves to the next state s_{k+1} with probability $T(s_k, a_k, s_{k+1})$. Let us assume that the agent adopts a stationary policy $\pi(s) : S \to A$. In an infinite horizon problem, the value function under this policy is

$$V_\pi(s) = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k C_k \Big| s_0 = s \right] \quad : S \to \mathbb{R} \tag{12}$$

where the expectation is on the state trajectories following policy π, so that $a_k = \pi(s_k)$, starting from state s. The optimal policy π^* is that minimizing the value function, for all states, and it can be identified by dynamic programming (e.g., using value iteration or policy iteration approaches) [9], [35].

B. Partially Observable Markov Decision Processes

A POMDP extends the MDP assuming that the current condition state is incompletely observable, and decisions cannot be taken as a function of this state. A POMDP is formally defined as an 8-tuple $(S, A, Y, C, T, E, \mathbf{b}_0, \gamma)$. At every step k, the agent receives observation y_k in domain set $Y = \{1, 2, \ldots, |Y|\}$. Emission function $E(s, a, y) : S \times A \times Y \to [0, 1]$ defines the probability of receiving observations: observation y_k is emitted at step k with probability $E(s_k, a_{k-1}, y_k)$. The belief vector, as defined in Section II-E, is updated at every step following (2), as $\mathbf{b}_{k+1} = \boldsymbol{\eta}(\mathbf{b}_k, y_{k+1})$, and $\mathbf{b}_0$ is the belief on to the initial state s_0.

The agent's policy, $\pi(\mathbf{b}) : \Omega_B \to A$, now is a mapping from the domain of beliefs to actions. The corresponding value function is

$$V_\pi(\mathbf{b}) = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k C_k \Big| \mathbf{b}_0 = \mathbf{b} \right] \quad : \Omega_B \to \mathbb{R} \tag{13}$$

where, again, the expectation is on the state trajectories following policy π, starting from belief $\mathbf{b}$. The optimal policy π^* is that minimizing the value function, for all beliefs. The corresponding optimal value function V^* can be approximated by the set Γ of $|H|$ α-vectors (defined also in Section II-E), as

$$V^*(\mathbf{b}) = \min_{h \in H} \left(\boldsymbol{\alpha}_h^\top \mathbf{b} \right) \tag{14}$$

and so it is a piecewise linear, concave function.

C. Relations Between Conditional Plans, Inner States, α-Vectors

Each α-vector is also related to a specific conditional plan [40], so that linear function $(\boldsymbol{\alpha}_h^\top \mathbf{b})$ is the value obtained executing plan h, as a function of the initial belief $\mathbf{b}$. When the initial belief belongs to region D_h, the plan related to vector $\boldsymbol{\alpha}_h$ is the optimal one. The plan functions defined in Section II-E can be derived by combining the policy function with the updating function in some equations: function f_0 is $\pi(m(\mathbf{b}_0))$, function $f_1(y_1)$ is $\pi(m(\boldsymbol{\eta}(\mathbf{b}_0, y_1)))$, function $f_2(y_1, y_2)$ is $\pi(m(\boldsymbol{\eta}(\boldsymbol{\eta}(\mathbf{b}_0, y_1), y_2)))$, and the following functions can be derived iteratively from the initial belief $\mathbf{b}_0$, so that function $f_k(y_1, y_2, \ldots, y_k)$ is $\pi(m(\eta(\ldots\ \eta(\eta(\mathbf{b}_0, y_1), y_2) \ldots, y_k))$.

As the cardinality $|Y|^k$ of the domain of function f_k grows exponentially with time step k, the complexity of a conditional plan becomes quickly unbearable. However, it can be argued that this growing complexity is unnecessary, as the history of past and current observations can be summarized by the "inner state." We can start defining this state h_k, at time t_k, as history $\{y_1, y_2, \ldots, y_k\}$. To prevent the exponential growth, an FSC assumes that different histories, also of different length, can lead to the same inner state. Formally, this is equivalent to a recursive conditional plan. Note that the initial conditional plan can be defined as an initial action and a set of $|Y|$ conditional plans (one per each observation value), starting from next step. In turn, each of those plans can be defined as an action followed by a set of $|Y|$ conditional plans, and so on. All these plans are infinite, when the horizon N is infinity. In a FCS, the plans are recursively defined by using $|H|$ basic recursive plans: the initial plan is one of these $|H|$, and so are, recursively, each plan at any consecutive step.

To better explain this, let us summarize a conditional plan with a planning function F, acting on a sequence of observations $\mathbf{y}$ of arbitrary length and mapping to the action set A so that, if the length of sequence $\mathbf{y}$ is k, then $F(\mathbf{y}) = f_k(\mathbf{y})$. A policy graph is associated with $|H|$ plan functions, $\{F_1, \ldots, F_{|H|}\}$ where function F_h is associated with node h, and it defines the plan starting from that inner state. These planning functions are defined recursively as $F_{\eta(h,y)}(\mathbf{y}) = F_h([\mathbf{y}, y])$. Moreover, when the sequence of observations is empty, the planning function is related to the policy: $F_h(\emptyset) = \pi(h)$. Hence, from the updating function η, the policy π and the initial inner state h, one can reconstruct the full conditional plan.

For example, the policy graph depicted in Fig. 2(b) is related to $|H| = 4$ conditional plans. The planning function for the first node is defined as $F_1([\mathbf{y}, 1]) = F_1(\mathbf{y})$ and $F_1([\mathbf{y}, 2]) = F_2(\mathbf{y})$. Hence, if the starting inner state is 1, then $f_0 = 1$, $f_1(1) = 1$ and $f_1(2) = 1$. Iteratively, the whole conditional plan can be derived.

It is straightforward to notice how each conditional plan is related to an α-vector. Following the approach of the Section IV-D applied to the joint graph, we can assign cost $c(w) = C(s, \pi(h))$ to node $w = \{s, h\}$, use the economic discount factor γ and the unaltered transition function T, to get the corresponding economic value vector $\mathbf{v}$ using (10). The corresponding h-entry $v(h)$ assesses the value of adopting the conditional plan starting from state s and so it coincides with $\alpha_h(s)$, i.e., the s entry of the α-vector h. Hence, the full set of $|H|$ α-vectors can be derived by value vector $\mathbf{v}$. If the agent follows the conditional plan F_h when the external state is s, the value is $\alpha_h(s)$. If the same plan is executed from belief $\mathbf{b}$, following (11), the corresponding value is $(\boldsymbol{\alpha}_h^\top \mathbf{b})$.

D. Deriving the Policy Graph From Set Γ of α-Vectors

The set Γ of α-vectors is generated by a point-based value iteration method for solving POMDPs, and those algorithms that can also provide the corresponding policy graphs [41]–[43]

But, here, we report an algorithm to derive, offline, a policy graph from Γ. While the set of nodes is immediately available, we need to identify the appropriate set of edges. For every node h, we identify the set of edges leaving that node by analyzing the outcome of updating the belief after every available observation $y \in Y$. We start by assigning a representative belief $\mathbf{b}_h$ to each node. As set D_h defines a convex polyhedron, a possible choice is to identify the Chebyshev center of the polyhedron [44], by linear programming. The polyhedron is defined by $(|H|-1)$ linear constraints, and the $j \neq h$ constraint is $(\boldsymbol{\alpha}_j - \boldsymbol{\alpha}_h)^\top \mathbf{b} \geq 0$. Once the representative point $\mathbf{b}_h$ is identified, the policy graph can be obtained by integrating (2)–(4), obtaining $\eta(h,y) = m(\boldsymbol{\eta}(\mathbf{b}_h, y))$. Alternative methods, can be based on estimating some "average point," as the center of gravity of a belief region, starting from examining the dominating points on a regular grid.

E. Undiscounted Policy Evaluation With Some Known Values

We refer to (10), with unitary discount factor γ, so that matrix $\mathbf{A}$ is singular. Let us assume that the set of nodes can be partitioned, possibly after reordering, in subset sets X and Z, or cardinality n_X and n_Z, respectively. The unknown values of nodes in X are listed in n_X-vector $\mathbf{v}_\mathrm{X}$, while those in Z in n_Z-vector $\mathbf{v}_\mathrm{Z}$. Transition matrix $\tilde{\mathbf{T}}$ is partitioned into the $n_\mathrm{X} \times n_\mathrm{X}$ submatrix $\tilde{\mathbf{T}}_\mathrm{XX}$, listing the transition probabilities among nodes in X, and $n_\mathrm{X} \times n_\mathrm{Z}$ submatrix $\tilde{\mathbf{T}}_\mathrm{XZ}$, listing the transition probabilities from nodes in X to nodes in Z. As all nodes in Z are absorbing, the $n_\mathrm{Z} \times n_\mathrm{X}$ submatrix listing probabilities from nodes in Z to nodes in X is made up only by zeros, and the submatrix listing probabilities among nodes in Z is the n_Z-dimensional identity matrix. Costs for nodes is X are listed in n_X-dimensional vector $\mathbf{c}_\mathrm{X}$, while those for nodes in Z are zeros. Hence, the recursive equation for vector $\mathbf{v}_\mathrm{Z}$ is a trivial identity, while that for vector $\mathbf{v}_\mathrm{X}$ is

$$\mathbf{v}_\mathrm{X} = \mathbf{c}_\mathrm{X} + \tilde{\mathbf{T}}_\mathrm{XX}\, \mathbf{v}_\mathrm{X} + \tilde{\mathbf{T}}_\mathrm{XZ}\, \mathbf{v}_\mathrm{Z}. \tag{15}$$

So vector $\mathbf{v}_\mathrm{X}$ can be obtained as solution of linear system $\mathbf{A}_\mathrm{X}\mathbf{v}_\mathrm{X} = \mathbf{c}^+_\mathrm{X}$, with $\mathbf{A}_\mathrm{X} = \mathbf{I} - \tilde{\mathbf{T}}_\mathrm{XX}$ and $\mathbf{c}^+_\mathrm{X} = \mathbf{c}_\mathrm{X} + \tilde{\mathbf{T}}_\mathrm{XZ}\, \mathbf{v}_\mathrm{Z}$.

F. Higher Moments of Cumulative (Virtual) Cost

While in Section IV-D, we have illustrated how to compute v_k, the first moment of the discounted cumulative costs G_k, we now illustrate how to obtain the higher moments. They can be computed interactively, from lower moments. We define the expected squared (virtual) cost as

$$\forall i,k,\;\; v_k^{(2)}(i) = \mathbb{E}_{\mathbf{s}|s_0=i}[(G_k(i))^2]. \tag{16}$$

By writing C_k is recursive form, we can related this value at the next step to quantities at the previous one

$$\forall i,k,\;\; v_{k+1}^{(2)}(i) = c^2(i) + 2\gamma\, c(i) \sum_j \tilde{T}(i,j)\, v_k(j) + \gamma^2 \sum_j \tilde{T}(i,j)\, v_k^{(2)}(j). \tag{17}$$

Listing these values, for every k, in vector $\mathbf{v}_k^{(2)} = [v_k^{(2)}(1), v_k^{(2)}(2), \ldots, v_k^{(2)}(n)]^\top$, previous equation can be written, in matrix form, as

$$\forall k,\;\; \mathbf{v}_{k+1}^{(2)} = \mathbf{c}_k^{(2)} + \gamma^2\, \tilde{\mathbf{T}}\, \mathbf{v}_k^{(2)} \tag{18}$$

where $\mathbf{c}_k^{(2)} = \mathbf{c}^2 + 2\gamma\, \mathbf{C}\, \tilde{\mathbf{T}}\mathbf{v}_k$ is an equivalent cost, $\mathbf{C}$ is a diagonal matrix that reports vector $\mathbf{c}$ on the diagonal, and $\mathbf{c}^2$ is vector whose element i is $c^2(i)$. The cost variance can be then computed as

$$\forall k,\;\; \sigma^2_{k+1}(i) = v_{k+1}^{(2)}(i) - v^2_{k+1}(i) \tag{19}$$

and the standard deviation by taking the square root.

Equations (16)–(18) and (20) can also be written for the infinite horizon case (when $k \to \infty$), dropping the subscript k, obtaining a recursive equation [corresponding to (18)], that can be solved similarly to (10). We also note that matrix $\tilde{\mathbf{T}}$ is the same of (10), and the discount factor is either unitary of less than one in the same condition (as $\gamma^2 = 1$ when $\gamma = 1$). Hence, after the first moment of the cost has been computed, the second moment can be computed with similar computations.

The third moment $v_k^{(3)}(i)$ is defined analogously to (16). Iterating the approach followed previously (with obvious meaning of the notation), the vector of third moments is defined as:

$$\forall k,\;\; \mathbf{v}_{k+1}^{(3)} = \mathbf{c}_k^{(3)} + \gamma^3\, \tilde{\mathbf{T}}\, \mathbf{v}_k^{(3)} \tag{20}$$

where $\mathbf{c}_k^{(3)} = \mathbf{c}^3 + 3\gamma\, \mathbf{C}^2\tilde{\mathbf{T}}\mathbf{v}_k + 3\gamma^2\mathbf{C}\, \tilde{\mathbf{T}}\mathbf{v}_k^{(2)}$ is an equivalent cost. Again, the infinite horizon case is analogously defined.

All higher moments can also be iterative computed, following this path. In the special cases discussed in Section IV-D, when the immediate costs are zero and the γ is one, (18) and (20) are identical to (10), and the computation of higher moments is useless.

G. Moments of First Passage Time

While all moments of the first passage time can be computed following the approach of Section IV-D and Appendix VIII-F, it is worth reporting the effective formulation proposed by Hunter [8]. For an ergodic FSMC, let τ_{ij} denote the number of steps needed to first reach state j from state i (or to return to the same state, when i and j are the same) and $\mu_{ij} = \mathbb{E}[\tau_{ij}]$. Let us arrange all expected times in matrix $\mathbf{M}$, so that $M(i,j) = \mu_{ij}$. When the FSMC is ergodic, $\mu_{ii} = 1/\beta(i)$, where $\beta(i)$ is entry i in the asymptotic distribution vector $\boldsymbol{\beta}$, defined in Section IV-C. So, after having solved (6), the diagonal of matrix $\mathbf{M}$ has been identified. We indicate with $(\cdot)_d$ the operator that acts one a square matrix and get another matrix of the same size with all zero elements, except for the diagonal, which is identical to that of the original matrix. So matrix $\mathbf{M}_d$ has elements $M_d(i,j) = \delta_{ij} m_{ij}$ and it is known. Following the approach of (10), $\mathbf{M}$ can be expressed in matrix equation:

$$\mathbf{M} = \mathbf{U} + \mathbf{T}(\mathbf{M} - \mathbf{M}_d) \tag{21}$$

where $\mathbf{U}$ is a matrix with all elements as one. The equation can be rewritten as

$$(\mathbf{I} - \mathbf{T})\mathbf{M} = \mathbf{U} - \mathbf{T}\mathbf{M}_d. \tag{22}$$

If $\mathbf{G}$ is pseudoinverse of $(\mathbf{I}-\mathbf{T})$, then (22) has solution

$$\mathbf{M} = [\mathbf{GB} - \mathbf{U}(\mathbf{GB})_d + \mathbf{I} - \mathbf{G} + \mathbf{UG}_d]\mathbf{M}_d \tag{23}$$

where $\mathbf{B}$ is a matrix with each row identical to the asymptotic distribution $\boldsymbol{\beta}$. The second moment of first passage time $\mathbf{M}^{(2)}$ is

$$\mathbf{M}^{(2)} = 2[\mathbf{GM} - \mathbf{U}(\mathbf{GM})_d] + [\mathbf{I} - \mathbf{G} + \mathbf{UG}_d][\mathbf{M}_d^{(2)} + \mathbf{M}_d] - \mathbf{M} \tag{24}$$

where $\mathbf{M}_d^{(2)} = 2\mathbf{M}_d(\mathbf{BM})_d - \mathbf{M}_d$. The variance matrix $\mathbf{V}$ can be computed as $\mathbf{V} = \mathbf{M}^{(2)} - \mathbf{M}^2$. The standard deviation matrix $\mathbf{S}$ is obtained by taking the elementwise square root of matrix $\mathbf{V}$. So, its element $S(i,j)$ defines the standard deviation of the first visiting time from state i to state j.

H. Transition Based on GM

Introducing the GM in Section II-E, we have discussed an approximate way to define a transition matrix among the H points in the $\overline{\Omega}_B$ domain, based on the nearest neighbor approach. Generalizations of that approach are based on assuming that, from belief $\overline{\mathbf{b}}_h$ and after observation y, the belief can transit to a set of possible belief, with appropriate probability. To do so, we can define $d(i,j,y) = \parallel \overline{\mathbf{b}}_j - \boldsymbol{\eta}(\overline{\mathbf{b}}_i, y) \parallel$ as the distance between the belief j and the posterior belief after belief i and observation y. We can define a kernel function $q(d)$, for example $q(d) = \exp(-d/\lambda)$, where λ is a parameter to be calibrated. The corresponding transition probability is $T(i,j) = \sum_y e(i,y)\ q(d(i,j,y))/Q(i,y)$, where $Q(i,y) = \sum_j q(d(i,j,y))$ is a normalization factor. For low values of λ, this method converges to the nearest neighbor approach. Other approaches can be based on local linear distributions. Alternative effective approaches are discussed by Zhuo and Hansen [45].

APPENDIX B

PARAMETER SETTINGS OF EXAMPLES

A. MDP Example 1

The parameters of the MDP example 1 are taken from a classical paper [36]. The pavement segment condition is discretized into $n = 8$ states using the pavement condition index rating presented in [46]. State 1 indicates a brand new pavement, and a higher state indicates a higher damage level up to state 8, which indicates a failure condition. Seven maintenance actions are available: do-nothing ($a = 1$), routine maintenance ($a = 2$), 1-in overlay ($a = 3$), 2-in overlay ($a = 4$), 4-in overlay ($a = 5$), 6-in overlay ($a = 6$), and reconstruction ($a = 7$). Time is discretized in years. The transition matrices for each action are shown in Table VIII. We assume the policy is described by vector $[3,\ 4,\ 4,\ 5,\ 6,\ 7,\ 7,\ 7]$, that maps all states to the corresponding action. This is the optimal policy for the infinite time horizon case, derived by [36].

B. MDP Examples 2

The second example is also derived taken from the same paper [36], by updating some parameters. We assume 3 actions: *do-nothing*, performing *minor repair,* and *major repair*. The annual transition probabilities for each action are shown in Table IX. The costs of *minor repair*, *major repair,* and of *failure* are $\$8\ K$, $\$20\ K$, and $\$500\ K$, respectively, time is discretized in years and the annual discount factor γ is 0.95. By solving that MDP, we obtain vectors $[1,\ 2,\ 2,\ 2,\ 3,\ 3]$ and

TABLE VIII
TRANSITION MATRICES FOR THE MDP EXAMPLE 1

$$\mathbf{T}_3 = \begin{bmatrix} 0.858 & 0.1358 & 0.0062 & 0 & 0 & 0 & 0 & 0 \\ 0.3605 & 0.4975 & 0.1358 & 0.0062 & 0 & 0 & 0 & 0 \\ 0.0371 & 0.3234 & 0.4975 & 0.1358 & 0.0062 & 0 & 0 & 0 \\ 0.0007 & 0.0364 & 0.3234 & 0.4975 & 0.1358 & 0.0062 & 0 & 0 \\ 0 & 0.0007 & 0.0364 & 0.3234 & 0.4975 & 0.1358 & 0.0062 & 0 \\ 0 & 0 & 0.0007 & 0.0364 & 0.3234 & 0.4975 & 0.1358 & 0.0062 \\ 0 & 0 & 0 & 0.0007 & 0.0364 & 0.3234 & 0.4975 & 0.142 \\ 0 & 0 & 0 & 0 & 0.0007 & 0.0364 & 0.3234 & 0.6395 \end{bmatrix}$$

$$\mathbf{T}_4 = \begin{bmatrix} 0.9839 & 0.0159 & 0.0002 & 0 & 0 & 0 & 0 & 0 \\ 0.7625 & 0.2215 & 0.0159 & 0.0002 & 0 & 0 & 0 & 0 \\ 0.2375 & 0.5249 & 0.2215 & 0.0159 & 0.0002 & 0 & 0 & 0 \\ 0.0161 & 0.2215 & 0.5249 & 0.2215 & 0.0159 & 0.0002 & 0 & 0 \\ 0.0002 & 0.0159 & 0.2215 & 0.5249 & 0.2215 & 0.0159 & 0.0002 & 0 \\ 0 & 0.0002 & 0.0159 & 0.2215 & 0.5249 & 0.2215 & 0.0159 & 0.0002 \\ 0 & 0 & 0.0002 & 0.0159 & 0.2215 & 0.5249 & 0.2215 & 0.0161 \\ 0 & 0 & 0 & 0.0002 & 0.0159 & 0.2215 & 0.5249 & 0.2375 \end{bmatrix}$$

$$\mathbf{T}_5 = \begin{bmatrix} 0.9993 & 0.0007 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.9629 & 0.0364 & 0.0007 & 0 & 0 & 0 & 0 & 0 \\ 0.6395 & 0.3234 & 0.0364 & 0.0007 & 0 & 0 & 0 & 0 \\ 0.142 & 0.4975 & 0.3234 & 0.0364 & 0.0007 & 0 & 0 & 0 \\ 0.0062 & 0.1358 & 0.4975 & 0.3234 & 0.0364 & 0.0007 & 0 & 0 \\ 0 & 0.0062 & 0.1358 & 0.4975 & 0.3234 & 0.0364 & 0.0007 & 0 \\ 0 & 0 & 0.0062 & 0.1358 & 0.4975 & 0.3234 & 0.0364 & 0.0007 \\ 0 & 0 & 0 & 0.0062 & 0.1358 & 0.4975 & 0.3234 & 0.0371 \end{bmatrix}$$

$$\mathbf{T}_6 = \begin{bmatrix} 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.9979 & 0.0021 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.9235 & 0.0744 & 0.0021 & 0 & 0 & 0 & 0 & 0 \\ 0.5 & 0.4235 & 0.0744 & 0.0021 & 0 & 0 & 0 & 0 \\ 0.0766 & 0.4235 & 0.4234 & 0.0744 & 0.0021 & 0 & 0 & 0 \\ 0.0021 & 0.0745 & 0.4235 & 0.4234 & 0.0744 & 0.0021 & 0 & 0 \\ 0 & 0.0021 & 0.0754 & 0.4235 & 0.4234 & 0.0744 & 0.0021 & 0 \\ 0 & 0 & 0.0021 & 0.0745 & 0.4235 & 0.4234 & 0.0744 & 0.0021 \end{bmatrix}$$

$$\mathbf{T}_7 = \begin{bmatrix} 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.9998 & 0.0002 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.9839 & 0.0159 & 0.0002 & 0 & 0 & 0 & 0 & 0 \\ 0.7625 & 0.2215 & 0.0159 & 0.0002 & 0 & 0 & 0 & 0 \\ 0.2375 & 0.5249 & 0.2215 & 0.0159 & 0.0002 & 0 & 0 & 0 \\ 0.0161 & 0.2215 & 0.5249 & 0.2215 & 0.0159 & 0.0002 & 0 & 0 \\ 0.0002 & 0.0159 & 0.2215 & 0.5249 & 0.2215 & 0.0159 & 0.0002 & 0 \end{bmatrix}$$

$$\mathbf{T} = \begin{bmatrix} 0.858 & 0.1358 & 0.0062 & 0 & 0 & 0 & 0 & 0 \\ 0.7625 & 0.2215 & 0.0159 & 0.0002 & 0 & 0 & 0 & 0 \\ 0.2375 & 0.5249 & 0.2215 & 0.0159 & 0.0002 & 0 & 0 & 0 \\ 0.142 & 0.4975 & 0.3234 & 0.0364 & 0.0007 & 0 & 0 & 0 \\ 0.0766 & 0.4235 & 0.4234 & 0.0744 & 0.0021 & 0 & 0 & 0 \\ 0.2375 & 0.5249 & 0.2215 & 0.0159 & 0.0002 & 0 & 0 & 0 \\ 0.0161 & 0.2215 & 0.5249 & 0.2215 & 0.0159 & 0.0002 & 0 & 0 \\ 0.0002 & 0.0159 & 0.2215 & 0.5249 & 0.2215 & 0.0159 & 0.0002 & 0 \end{bmatrix}$$

TABLE IX
TRANSITION MATRICES FOR THE MDP EXAMPLE 2

$$\mathbf{T}_1 = \begin{bmatrix} 0.9 & 0.05 & 0.02 & 0.02 & 0.01 & 0.00 \\ 0.0 & 0.80 & 0.10 & 0.06 & 0.03 & 0.01 \\ 0.0 & 0.00 & 0.75 & 0.10 & 0.10 & 0.05 \\ 0.0 & 0.00 & 0.00 & 0.50 & 0.30 & 0.20 \\ 0.0 & 0.00 & 0.00 & 0.00 & 0.35 & 0.65 \\ 0.0 & 0.00 & 0.00 & 0.00 & 0.00 & 1.00 \end{bmatrix}$$

$$\mathbf{T}_2 = \begin{bmatrix} 0.4 & 0.32 & 0.1 & 0.1 & 0.078 & 0.002 \\ 0.4 & 0.32 & 0.1 & 0.1 & 0.078 & 0.002 \\ 0.4 & 0.32 & 0.1 & 0.1 & 0.078 & 0.002 \\ 0.4 & 0.32 & 0.1 & 0.1 & 0.078 & 0.002 \\ 0.0 & 0.00 & 0.0 & 0.0 & 0.600 & 0.400 \\ 0.0 & 0.00 & 0.0 & 0.0 & 0.000 & 1.000 \end{bmatrix}$$

$$\mathbf{T}_3 = \begin{bmatrix} 0.9 & 0.05 & 0.02 & 0.02 & 0.008 & 0.002 \\ 0.9 & 0.05 & 0.02 & 0.02 & 0.008 & 0.002 \\ 0.9 & 0.05 & 0.02 & 0.02 & 0.008 & 0.002 \\ 0.9 & 0.05 & 0.02 & 0.02 & 0.008 & 0.002 \\ 0.9 & 0.05 & 0.02 & 0.02 & 0.008 & 0.002 \\ 0.9 & 0.05 & 0.02 & 0.02 & 0.008 & 0.002 \end{bmatrix}$$

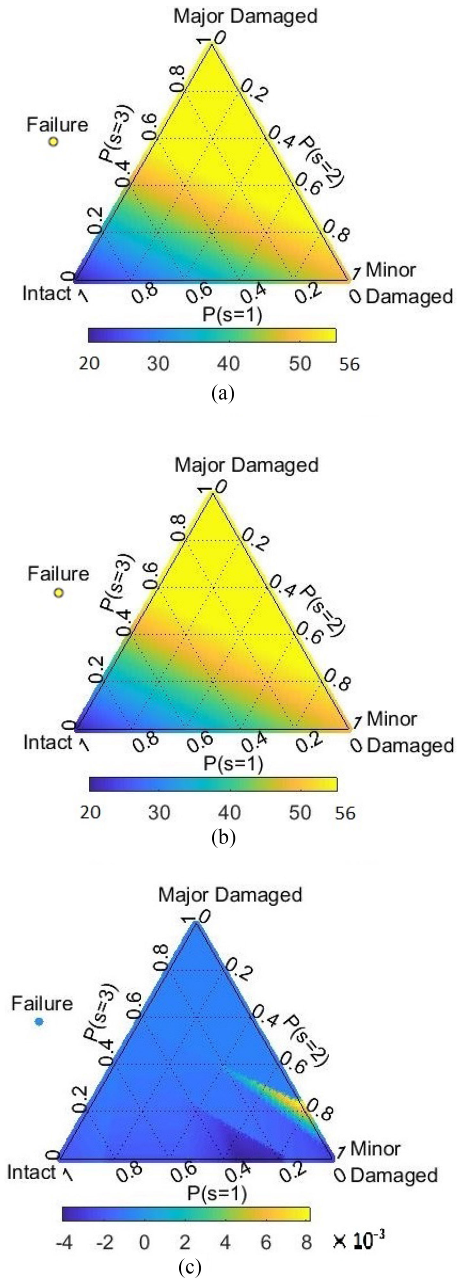

Fig. 20. Value function obtained from the set Γ of α-vectors (a), ORM (b), and their difference (c).

\$$[33.50,\ 50.91,\ 50.91,\ 50.90,\ 53.53,\ 553.53]$K, which represent the optimal policy π and the corresponding optimal value, for all states from 1 to 6, respectively.

C. Reconstructed Value Function

We now illustrate how the policy related to ORM, in the analysis of the example of Section V-D, is only slightly suboptimal, respected to that defined by the set Γ of α-vectors. Once the policy graph has been identified via ORM following the approach of Appendix A-D, we can assess the corresponding value, following the approach of Section IV-D and Appendix VIII-C. Fig. 20 shows the value function (as a multiple of the inspection cost) related to Γ (a), and to ORM (b). The two functions are almost identical, as they difference (c) is less than 0.5% of the value. For the failure state, the values are also almost identical: 150.03 from analysis of Γ, 150.07 from ORM. We conclude that the policy related to ORM is almost optimal, and the small differences between the two value functions can be related to the approximation in the numerical solver that generates set Γ.

References

[1] W. Feller, *An Introduction to Probability Theory and Its Applications*. Hoboken, NJ, USA: Wiley, 1971.

[2] G. Grimmett and D. Stirzaker, *Probability and Random Processes*. Oxford, U.K.: Oxford Univ. Press, 2001.

[3] R. G. Gallager, *Stochastic Processes: Theory for Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2013.

[4] J. R. Benjamin and C. A. Cornell, *Probability, Statistics, and Decision for Civil Engineers*. New York, NY, USA: McGraw-Hill, 1970.

[5] A. Ang and W. Tang, *Probability concepts in engineering planning and design*: *Decision, Risk, and Reliability*, vol. 2. Hoboken, NJ, USA: Wiley, 1984.

[6] T. J. Sheskin, "Computing mean first passage times for a Markov chain," *Int. J. Math. Educ. Sci. Technol.*, vol. 26, no. 5, pp. 729–735, 1995.

[7] J. J. Hunter, "Simple procedures for finding mean first passage times in Markov chains," *Asia-Pacific J. Oper. Res.*, vol. 24, no. 6, pp. 813–829, 2007.

[8] J. J. Hunter, "Variances of first passage times in a Markov chain with applications to mixing times," *Linear Algebra Appl.*, vol. 429, no. 5/6, pp. 1135–1162, 2008.

[9] D. P. Bertsekas, Dynamic programming and optimal control, *Belmont: Athena Sci.*, vol. 1, 2000.

[10] G. Shani, J. Pineau, and R. Kaplow, "A survey of point-based POMDP solvers," *Auton. Agent Multiagent Syst.*, vol. 27, pp. 1–51, 2013.

[11] D. Braziunas, "POMDP solution methods," University of Toronto, Canada, Tech. Rep. 2003.

[12] H. Yu and D. P. Bertsekas, "On near optimality of the set of finite-state controllers for average cost POMDP," *Math. Oper. Res.*, vol. 33, no. 1, pp. 1–11, 2008.

[13] E. A. Hansen, "Solving POMDPs by searching in policy space," in *Proc. UAI*, 1998, *arXiv:1301.7380*.

[14] N. Meuleau, L. Peshkin, K.-E. Kim, and L. P. Kaelbling, "Learning finite-state controllers for partially observable environments," in *Proc. UAI*, 1999, *arXiv:1301.6721*.

[15] W. T. Scherer and D. M. Glagola, "Markovian models for bridge maintenance management," *J. Transp. Eng.*, vol. 120, no. 1, pp. 37–51, 1994.

[16] A. Platis, N. Limnios, and M. Le Du, "Performability of electric-power systems modeled by non-homogeneous Markov chains," *IEEE Trans. Rel.*, vol. 45, no. 4, pp. 605–610, Dec. 1996.

[17] S. V. Dhople, Y. C. Chen, and A. D. Domínguez-García, "A set-theoretic method for parametric uncertainty analysis in Markov reliability and reward models," *IEEE Trans. Rel.*, vol. 62, no. 3, pp. 658–669, Sep. 2013.

[18] K. Smilowitz and S. Madanat, "Optimal inspection and maintenance policies for infrastructure networks," *Comput.-Aided Civil Infrastructure Eng.*, vol. 15, no. 1, pp. 5–13, 2000.

[19] C.-A. Robelin and S. Madanat, "History-dependent bridge deck maintenance and replacement optimization with Markov decision processes," *J. Infrastructure Syst.*, vol. 13, no. 3, pp. 195–201, 2007.

[20] H. Gao and X. Zhang, "A markov-based road maintenance optimization model considering user costs," *Comput.-Aided Civil Infrastructure Eng.*, vol. 28, no. 6, pp. 451–464, 2013.

[21] K. G. Papakonstantinou and M. Shinozuka, "Planning structural inspection and maintenance policies via dynamic programming and markov processes part ii: POMDP implementation," *Rel. Eng. Syst. Saf.*, vol. 130, pp. 214–224, 2014.

[22] M. Pozzi, M. Memarzadeh, and K. Klima, "Hidden-model processes for adaptive management under uncertain climate change," *J. Infrastructure Syst.*, vol. 23, no. 4, 2017, Art. no. 04017022.

[23] M. Memarzadeh, M. Pozzi, and J. Zico Kolter, "Optimal planning and learning in uncertain environments for the management of wind farms," *J. Comput. Civil Eng.*, vol. 29, no. 5, 2014, Art. no. 04014076.

[24] M. Memarzadeh and M. Pozzi, "Value of information in sequential decision making: Component inspection, permanent monitoring and system-level scheduling," *Rel. Eng. Syst. Saf.*, vol. 154, pp. 137–151, 2016.
[25] Y. Pan and M. U. Thomas, "Repair and replacement decisions for warranted products under Markov deterioration," *IEEE Trans. Rel.*, vol. 59, no. 2, pp. 368–373, Jun. 2010.
[26] R. B. Corotis, J. Hugh Ellis, and M. Jiang, "Modeling of risk-based inspection, maintenance and life-cycle cost with partially observable markov decision processes," *Struct. Infrastructure Eng.*, vol. 1, no. 1, pp. 75–84, 2005.
[27] R. Srinivasan and A. K. Parlikad, "Semi-Markov decision process with partial information for maintenance decisions," *IEEE Trans. Rel.*, vol. 63, no. 4, pp. 891–898, Dec. 2014.
[28] A. Arab, E. Tekin, A. Khodaei, S. K. Khator, and Z. Han, "System hardening and condition-based maintenance for electric power infrastructure under hurricane effects," *IEEE Trans. Rel.*, vol. 65, no. 3, pp. 1457–1470, Sep. 2016.
[29] M. Chen, H. Fan, C. Hu, and D. Zhou, "Maintaining partially observed systems with imperfect observation and resource constraint," *IEEE Trans. Rel.*, vol. 63, no. 4, pp. 881–890, Dec. 2014.
[30] L. C. E. R. R. L. Cormen, T. H. and C. Stein, *Introduction to Algorithms*. Cambridge, MA, USA: MIT Press, 2009.
[31] J. Erickson, *Algorithms*, 2019. [Online]. Available: http://algorithms.wtf
[32] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*. Philadelphia, PA, USA: SIAM, vol. 71, 2000.
[33] M. Salter-Townshend, A. White, I. Gollini, and T. B. Murphy, "Review of statistical network analysis: Models, algorithms, and software," *Stat. Anal. Data Mining, ASA Data Sci. J.*, vol. 5, no. 4, pp. 243–264, 2012.
[34] M. Kac, "On the notion of recurrence in discrete stochastic processes," *Bull. Amer. Math. Soc.*, vol. 53, pp. 1002–1010, 1947.
[35] R. A. Howard, *Dynamic Probabilistic Systems*, Vol. II: *Semi-Markov and Decision Processes*. Hoboken, NJ, USA: Wiley, 1971.
[36] P. L. Durango and S. Madanat, "Optimal maintenance and repair policies in infrastructure management under uncertain facility deterioration rates: An adaptive control approach," *Transp. Res. A, Policy Pract.*, vol. 36, no. 9, pp. 763–778, 2002.
[37] C. Andriotis and K. Papakonstantinou, "Managing engineering systems with large state and action spaces through deep reinforcement learning," *Rel. Eng. Syst. Saf.*, vol. 191, 2019, Art. no. 106483.
[38] H. Kurniawati, D. Hsu, and W. S. Lee, "SARSOP: Efficient point-based POMDP planning by approximating optimally reachable belief spaces," in *Robotics: Science and systems*, vol. 2008. Zurich, Switzerland., 2008.
[39] C. Boettiger, "A library for solving POMDPs," Github repository, 2019. [Online]. Available: https://github.com/boettiger-lab/sarsop
[40] S. J. Russell and P. Norvig, *Artificial Intelligence, A Modern Approach. Englewood Cliffs, NJ, USA: Prentice-Hall*, 2010.
[41] H. Kamalzadeh, M. Hahsler, and A. Cassandra, POMDP: *Solver for Partially Observable Markov Decision Processes*, *r package version 0.9.2.*, 2019. [Online]. Available: https://CRAN.R-project.org/package=pomdp
[42] "POMDP-solve." Accessed:Apr. 28, 2015. [Online]. Available: https://pomdp.org/code/index.html
[43] "Approximate POMDP Planning Software." Accessed: Apr. 28, 2014. [Online]. Available: https://bigbird.comp.nus.edu.sg/pmwiki/farm/appl/.
[44] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[45] R. Zhuo and E. A. Hansen, "An improved grid-based approximation algorithm for POMDPs," in *Proc. Int. Joint Conf. Artif. Intell.*, 2001, pp. 707–716.
[46] J. Camahan, W. Davis, M. Shahin, P. Keane, and M. Wu, "Optimal maintenance decisions for pavement management," *J. Transp. Eng.*, vol. 113, no. 5, pp. 554–572, 1987.

Shuo Li received the B.S. degree in theoretical and applied mechanical engineering from Sun Yat-sen University, Guangzhou, China, in 2016, and the M.S. degree in civil engineering in 2018 from Carnegie Mellon University, Pittsburgh, PA, USA, where he is currently working toward the Ph.D. degree with the Department of Civil and Environmental Engineering.

His research interests include value of information and partially observable Markov decision processes, applied to civil infrastructure systems.

Matteo Pozzi received a Laurea degree in civil engineering and the Ph.D. degree in structural engineering from the University of Trento, Trento, Italy, in 2003 and 2007, respectively.

After a Postdoc with the University of California, Berkeley, in 2012, he was with the Department of Civil and Environmental Engineering, Carnegie Mellon University, where is currently an Associate Professor. He is also affiliated with the Wilton E. Scott Institute for Energy Innovation at the same university. He has authored 40 journal papers. His research interests include probabilistic data processing, reliability analysis, and sequential decision optimization for civil infrastructures, with application to maintenance and inspection scheduling, climate change and extreme heat, and interconnected systems.