Maintenance Planning under Imperfect Monitoring: an Efficient POMDP Model Using Interpolated Value Function

Matthieu Roux, Yiping Fang, A Barros

Maintenance Planning under Imperfect Monitoring: an Efficient POMDP Model Using Interpolated Value Function ⋆

M. Roux, Y.-P. Fang, A. Barros

Univ. Paris-Saclay, CentraleSuplec, LGI EA 2606, France
Chair on Risk and Resilience of Complex Systems
e-mail: {matthieu.roux, yiping.fang, anne.barros }@centralesupelec.fr

Abstract: We develop in this paper a *partially observable Markov decision process* (POMDP) model for a maintenance planning problem and solve it with an efficient point-based value iteration (PBVI) algorithm. We consider a single-unit system, subjected to random degradation and failures, and for which the current degradation state can be partially observed via an imperfect monitoring system. The system state space is finite, and we model the following maintenance operations: i) perfect inspection, ii) preventive maintenance and iii) corrective maintenance. The goal is to optimize the maintenance policy by taking into account the imperfect monitoring data in order to minimize the expected discounted maintenance cost over an infinite time horizon. We formulate the problem as a POMDP where, at each time step, it should be decided whether or not to conduct a maintenance operation, and if so, which one. To keep the model general and flexible, we suppose that monitoring data are collected every K time steps (i.e. one observation epoch). The model is completed by a constraint imposing that only one maintenance operation can be conducted per observation epoch. Eventually, we solve it using a PBVI algorithm. The value function is approximated by interpolation of grid data points, and new relevant points are dynamically added into the grid where they most improve the value function. This approach is compared to a POMDP modeling based on approximate *sample paths* (ASP); when evaluated in different cost scenarios, the proposed approach systematically finds better maintenance policies for a comparable computation time. The computation of a lower bound finally proves that we are able to get the optimal value of the problem with satisfying precision.

Keywords: optimal maintenance planning, condition-based maintenance, imperfect monitoring, partially observable Markov decision process, point-based value iteration.

1. INTRODUCTION

Industrial assets are complex physical systems, made up of many components in interaction with each other. The components are not perfect and they degrade with time, accumulating fatigue and wear. Eventually, they are all subjected to failure, which can occur at random times. For various reasons (cost, reliability, availability of repair crew, etc...), it is often preferable to replace an item preventively, before its failure. Engineers and researchers initially tackled the problem of maintenance planning by optimizing the period at which a system should be preventively replaced, leading to the development of periodic maintenance policies (examples can be found in Barlow (1960) or Bajestani (2016)).

Yet, periodic maintenance is a rigid policy, incapable of adapting the replacement age of the component to the actual degradation. For example, if a maintenance operator observes that the system is in good condition when it should theoretically be replaced, delaying a little bit the preventive replacement would probably be more cost effective. For that reason, *condition-based maintenance* (CBM) policies, which adapt to the actual condition of the system, have received a lot of attention in the past years (Alaswad (2017)). CBM relies on continuous system condition monitoring by means of physical measurements such as temperature, pressure, vibration, noise, etc.

However, very few studies have investigated the potential effects of imperfect condition monitoring on CBM policies (Alaswad (2017)). de Jonge (2020) also suggested that more research could be done on imperfect monitoring, and especially on how to find the optimal dynamic CBM policies for multi-item systems by taking into account the imperfectly monitored condition of each item in the system. We are convinced of the importance of exploring such research direction. When perfect information is not possible or too expensive, an efficient maintenance policy should still be able to capture the value provided by the imperfect condition monitoring. Yet, to the best of our knowledge, efficient tools are still missing for tackling the problem of maintenance planning under imperfect monitoring information. Even on single-unit systems, as a start.

⋆ This work is funded by the Chair on Risk and Resilience of Complex Systems (CentraleSupelec, EDF, Orange, SNCF).

Partially observable Markov decision processes (POMDPs) (Sondik (1971)) are a natural extension of *Markov decision processes* (MDPs) and provide a fecund framework to deal with this context of partial information. Despite a lot of applications found in the robotics community, POMDPs have not received much attention in the reliability and maintenance community. Ghasemi (2007) used a POMDP to optimize maintenance planning, but without the possibility of field inspection and for which maintenance decision could only be taken at observation time. Papakonstantinou (2014a) and Papakonstantinou (2014b) proposed a thorough study of maintenance planning under imperfect information and examined a detailed POMDP case study with a relatively large state space. However, limitation on the maintenance resource is not explored. Eventually, Byon (2010) proposed a POMDP case study in which they investigated, without really saying it, a model based on approximate *sample paths* (ASP). In their work, the POMDP model is only approximated because future (imperfect) observations are not taken into account when computing the *sample paths* and the dynamic programming value function, which simplifies the computations in the infinite and continuous belief state space.

In this study, we investigate the fundamental question of the extent to which maintenance planning problems under imperfect information can be modeled and efficiently solved by POMDPs. Through a simple case study, we aim at 1) illustrating in practice a POMDP solving technique, the point-based value iteration (PBVI) algorithm combined with interpolated value function; the size of the state space is relatively modest, but the focus here is on having a general model integrating some resource limitations (at most one intervention can be conducted per observation epoch, where imperfect monitoring data is only collected at the beginning of each observation epoch), and 2) comparing this approximate solving technique with the approximate model proposed by Byon (2010). We formulate the maintenance planning problem in section 2. In section 3, the PBVI algorithm is described as well as the lower bound it provides. Eventually, we discuss the numerical results in section 4, where our proposed method is compared to the ASP approximation from Byon (2010).

2. MODEL DESCRIPTION

2.1 Maintenance problem

We study a single-unit system, progressively degrading over time and subject to random failures. We suppose that the state space $\mathcal{S}$ of the system is finite, with $\mathcal{S} = \{S_1, S_2, S_3, S_4, F\}$; S_1 being the *as-good-as-new* state, S_4 the most degraded yet functioning state, and F the failed state (single failure mode). The evolution of the state of the system over time is modeled by a Markov chain. Time is discretized (typically, one time step represents one day) and the degradation process is modeled by random transitions from one state $s \in \mathcal{S}$ to another $s' \in \mathcal{S}$ (potentially identical) between two consecutive time steps. The transition probabilities are given by a transition matrix, noted P. In view of common characteristics of degradation processes in practice, we adopt the following assumptions on the transition matrix P:

i) The state of the system cannot spontaneously improve, meaning that transitions $F \to S_i$ and $S_i \to S_j$ with $j < i$ have a probability of zero.
$$\mathbb{P}(F \to S_i) = P[F, S_i] = 0, \quad \forall\, S_i$$
$$\mathbb{P}(S_i \to S_j) = P[S_i, S_j] = 0, \quad \forall\, i > j$$

ii) The system can transition to failure from any functioning state, but the more degraded the state, the most likely a failure can occur at the next time step.
$$\mathbb{P}(S_i \to F) = P(S_i, F) > 0, \quad \forall\, S_i$$
$$P(S_i, F) < P(S_j, F), \quad \forall\, i < j$$

When the system fails, which corresponds to a transition $S_i \to F$, a corrective maintenance (**CM**) can be conducted to replace it (the system is brought back to the *as-good-as-new* state S_1), with a cost noted $cost_{CM}$. If the system is not replaced immediately after the failure, an opportunity cost $cost_{OP}$ will be incurred for each time step when the system is left in failed state F.

As **CM** is typically expensive, it is possible to conduct preventive maintenance (**PM**), which consists in preventively replacing the system with a new one before a failure. Such intervention also brings the system back to state S_1, at a cost $cost_{PM} < cost_{CM}$. Based on the knowledge of the current state of the system, a CBM policy should then find a tradeoff between too frequent **PM**s leading to high maintenance cost and too frequent unexpected failures leading to too many expensive **CM**s.

Nevertheless, considering that the decision-maker has a perfect knowledge of the exact degradation state is a quite strong assumption and many applications must take into account the imperfection of a *monitoring system*. By *monitoring system*, we refer to the set of all sensors collecting physical measurements and subsequent communicative and analytic technologies implemented to infer and track the health condition of an industrial asset. In practice, system health state information provided by a monitoring system is inevitably subject to noise and modeling inaccuracy and, therefore, is flawed. In the following work, we suppose that the monitoring is imperfect, and we develop a CBM strategy well adapted for this context of imperfect information.

To complement these partial data, we add the possibility to conduct a perfect inspection (**I**), with a cost $cost_I < cost_{PM}$. When performing intervention **I**, the decision-maker observes the true state of the system and, based on the result of this inspection, may decide to conduct a **PM** in the same time step (if necessary). In our maintenance problem, perfect inspection is therefore the only way to have access to the true degradation state of the system with certainty.

2.2 Imperfect monitoring

For numerous reasons, it is sometimes impossible to directly access to the degradation state of a system. It can be, for example, because the system is remote or too difficult to inspect without shutting it down. In those cases, the implementation of sensors may be valuable since it provides real-time operational data at affordable costs while the system keeps operating. It is important to note that this remains true even if this information is only par-

tial: observing an unusual data point may not necessarily correspond to an abnormal (e.g., degraded) state, but the repetition of several similar measures at different times is *probably* the indication of ongoing degradation.

We suppose the monitoring system has a finite set $\mathcal{O}$ of possible observations, or outputs. An imperfect monitoring system means that we do not have a deterministic relation between the observation $o \in \mathcal{O}$ and the underlying degradation state $s \in \mathcal{S}$; the best we can do is exploit the stochastic dependence between o and s. We assume that, in order to optimize our maintenance strategy, we know the average performance of the monitoring system, which consists in knowing $\mathbb{P}(o|s) = Q(s,\, o)$ the conditional probability of receiving the observation $o \in \mathcal{O}$ given that the system state is $s \in \mathcal{S}$. The matrix Q will be called the *monitoring matrix*.

In this work, we assume that failures are self-announcing, meaning that the decision-maker has perfect knowledge about the failed state as soon as a failure occurs (no need for an inspection to notice a failure). This also means that observations are not needed when the system is failed, since this information is already known with certainty. Consequently, the matrix Q only needs to be defined on functioning states S_i (i.e. for $Q(S_i,\, o)$).

2.3 A POMDP model

Our problem typically falls in the category of POMDPs. POMDPs are a natural extension of MDPs, where one decision should be taken at each time step. The goal here is to optimize the maintenance policy in order to minimize the total discounted maintenance cost over an infinite time horizon. In our case, the set of possible decisions (or *actions*) is the following: $\mathcal{A} = \{\mathbf{NA},\, \mathbf{I},\, \mathbf{PM},\, \mathbf{CM}\}$, where **NA** (*no action*) is simply the situation when no intervention is conducted on the system.

Because we optimize over an infinite time horizon, we search for a stationary policy. However, in this context of imperfect information, a policy cannot just map a state s to an action $a \in \mathcal{A}$, for the reason that we do not have access to the true degradation state s when taking maintenance decisions. Consequently, a policy in this context should map our *best knowledge* about the degradation state to an action a. But how do we define our *best knowledge* ? As our knowledge about the system is uncertain, it can thus be described by a *belief* $b \in \mathcal{B} = \{b \in [0,\, 1]^{|\mathcal{S}|} : \sum_{s\in\mathcal{S}} b[s] = 1\}$, which is nothing more than just a probability distribution over $\mathcal{S}$ describing our uncertainty about s. We note $b[s]$ the probability that, given the best of our knowledge, the system is in state s. Eventually, here the best of our knowledge is all the information the decision-maker has had access to before taking the decision. It is basically the history of past observations and actions since the last **PM**, **CM** or **I** (moment when the true state of the system was known with certainty). In fact, as stated in Papakonstantinou (2014a), the belief b constitutes a sufficient statistic summarizing all the information of past observations and actions.

Each time a new observation o is acquired, the current belief b should be updated to b' using Bayes' formula:

$$b'[s] = \frac{Q(s,\, o)b[s]}{\sum_{s_k\in\mathcal{S}\setminus\{F\}} Q(s_k,\, o)b[s_k]} \tag{1}$$

Additionally, the belief b should also take into account the effect of random transitions between two consecutive time steps. Let $p(b) = \sum_{s\in\mathcal{S}} P(s,\, F)b[s]$ be the total probability of failure at time $t+1$, conditionally to the fact that at time t, the state of the system is distributed as b. Then, if the unit does not fail at time $t+1$, the next belief b' is defined as

$$b'[s] = \frac{\sum_{s_k\in\mathcal{S}\setminus\{F\}} P(s_k,\, s)b[s_k]}{1 - p(b)} \tag{2}$$

Finally, we do not detail it here as it is quite straightforward but actions should also lead to updating the current belief:

- **NA** does not change anything
- **PM** and **CM** update b to the perfect knowledge of as-good-as-new state S_1 (we allow ourselves to abuse the notation S_1 to also refer to the extreme belief where state S_1 is the true state with probability 1)
- **I** (not followed by a **PM**) should update b with the perfect knowledge resulting from the perfect inspection

Then, such class of POMDP problem for identifying the optimal maintenance policies can be solved by dynamic programming, e.g., using *value iteration* to search for the optimal value function. We define the value function $V_\pi(b)$ as being the expected total discounted maintenance cost over an infinite time horizon when applying the policy π to a system initially in a state described by the belief b. Our objective is to find a policy π^* such that:

$$V_{\pi^*}(b) := \min_{\pi\in\Pi} V_\pi(b) = \mathbb{E}\Big[\sum_{t=0}^{+\infty} \gamma^t c(s_t,\, a_t)\Big] \tag{3}$$

with $a_t := \pi(b_t)$ action taken at time t
b_t : belief at time t ($b_0 = b$)
$0 < \gamma < 1$: discount factor
s_t : degradation state at time t
s_0 : initial state of the system, distributed as b
$c(s,\, a)$: cost resulting from being in state s and taking action a

2.4 Specific constraints

To model our problem, we introduce a couple of specific constraints for maintenance planning, which usually do not appear in classical POMDP models:

- **Observation epoch.** To remain generic and flexible, we assume that data is collected from the monitoring system every $K \in \mathbb{N}^*$ time steps, at the beginning of each observation epoch. In other words, observation and decision are not necessarily synchronous.
- **Limitation on the maintenance resources.** In order to pave the way for future more realistic use cases, we explicitly limit the use of maintenance resources. This problem might not be prominent for

single-unit systems, but we plan soon to extend this work to multi-units systems where the maintenance resources are limited and shared over the whole system. Thus, here we assume that at most one maintenance intervention could be conducted during each observation epoch; a maintenance intervention being either **I**, **PM** or **CM**.

2.5 Details on the value function

With the two additional constraints, the value function does not uniquely depend on the current belief b, but also on:

i) $k \in \{0, 1, 2, ..., K-1\}$ the current position within the ongoing observation epoch ($k = 0$ being the first time step of the observation epoch), and

ii) $\xi \in \{0, 1\}$ indicating the number of interventions already conducted within the ongoing observation epoch.

For any $b \in \mathcal{B}$, $k \in \{0, 1, ..., K-1\}$ and $\xi \in \{0, 1\}$, we define $V_\pi(b, k, \xi)$ as the expected discounted cost that results from implementing the policy π on a system that is initially in state s_0 distributed as b, for which $t = 0$ corresponds to a position k within the initial observation epoch and a situation where ξ interventions have already been conducted. We propose the following formalism to extend (3) and model the additional constraints previously mentioned:

$$V_{\pi^*}(b,k,\xi) := \min_{\pi\in\Pi} V_\pi(b,k,\xi) = \mathbb{E}\Big[\sum_{t=0}^{+\infty} \gamma^t c(s_t, a_t)\Big] \quad (4)$$

$$\text{with } a_t := \pi(b_t,\, k_t,\, \xi_t)$$

$$b_0 = b$$

$$b_{t+1} = \begin{cases} b'(b_t) & \text{if } k_{t+1} > 0 \\ b'(b_t,\, o_{t+1}) & \text{if } k_{t+1} = 0 \end{cases}$$

$$k_0 = k$$

$$k_{t+1} = \begin{cases} k_t + 1 & \text{if } k_t + 1 < K \\ 0 & \text{otherwise.} \end{cases}$$

$$\xi_0 = \xi$$

$$\xi_{t+1} = \begin{cases} \xi_t & \text{if } a_{t+1} = \mathbf{NA} \text{ and } k_{t+1} > 0 \\ \xi_t + 1 & \text{if } a_{t+1} \neq \mathbf{NA} \text{ and } k_{t+1} > 0 \\ 0 & \text{if } a_{t+1} = \mathbf{NA} \text{ and } k_{t+1} = 0 \\ 1 & \text{if } a_{t+1} \neq \mathbf{NA} \text{ and } k_{t+1} = 0 \end{cases}$$

$$\xi_t \leq 1 \quad \text{constraint on the maintenance resource}$$

Below are given the Bellman equations from which the value function V is defined. Here, we use v^a to denote the optimal value of taking action a at the current time step and then acting optimally (i.e. following the optimal policy).

Performing a **PM** or a **CM** brings the system back to state S_1:

$$v^{PM}(k) = cost_{PM} + V(S_1,\, k,\, 1), \quad \forall\, k \quad (5)$$

$$v^{CM}(k) = cost_{CM} + V(S_1,\, k,\, 1), \quad \forall\, k \quad (6)$$

When the system is failed and **CM** cannot be performed, i.e. $\xi = 1$, the system remains failed and an opportunity cost is incurred; a **CM** will be possible at the next time step only if it is the start of a new observation epoch. $V_F(k)$ will refer to the value of being in state F at the beginning of period k within the initial observation epoch and with no possibility of doing a **CM** immediately:

$$V_F(k) = cost_{OP} + \gamma V_F(k+1), \quad \forall\, k < K-1 \quad (7)$$

$$V_F(K-1) = cost_{OP} + \gamma v^{CM}(0) \quad (8)$$

Choosing **NA** when $k < K-1$ means that no observation will be acquired at the next time step; the belief at the next time step is then noted b':

$$v^{NA}(b,\, k,\, \xi = 0) = 0 + p(b)\gamma v^{CM}(k+1) + (1-p(b))\gamma V(b',\, k+1,\, 0), \quad \forall\, k < K-1 \quad (9)$$

$$v^{NA}(b,\, k,\, \xi = 1) = 0 + p(b)\gamma V_F(k+1) + (1-p(b))\gamma V(b',\, k+1,\, 1), \quad \forall\, k < K-1 \quad (10)$$

When $k = K-1$, choosing **NA** means that, if the unit does not fail, an observation $o \in \mathcal{O}$ will be acquired at the next time step; the belief at this next time step, which should depend on the observation, is noted $b'(o)$; moreover, as the next time step corresponds to the start of a new observation epoch, a **CM** will be possible in case of sudden failure whatever the current value of ξ. Therefore, we have:

$$v^{NA}(b,\, K-1,\, \xi) = 0 + p(b)\gamma v^{CM}(0) + (1-p(b))\gamma \sum_{o\in\mathcal{O}} \mathbb{P}(o|b')V(b'(o),\, 0,\, 0), \quad \forall\, \xi \quad (11)$$

Performing a perfect inspection **I** allows to choose the best option between **NA** and **PM** while having access to the true degradation state; the parameter ξ is not explicitly required in v^I since such inspection is only possible when $\xi = 0$:

$$v^I(b,\, k) = cost_I + \sum_{s\in\mathcal{S}} b[s] \min\{v^{NA}(s,\, k,\, 1);\, v^{PM}(k)\}, \quad \forall\, k \quad (12)$$

Eventually, the value function V is computed as the min of all feasible alternatives. When $\xi = 1$, only **NA** is feasible. When $\xi = 0$, all actions are possible and we should choose the best one as follows:

$$V(b,\, k,\, \xi = 1) = v^{NA}(b,\, k,\, 1), \quad \forall\, k \quad (13)$$

$$V(b,\, k,\, \xi = 0) = \min\{v^{NA}(b,\, k,\, 0);\, v^I(b,\, k);\, v^{PM}(k)\}, \quad \forall\, k \quad (14)$$

3. SOLVING TECHNIQUE BY POINT-BASED VALUE ITERATION AND INTERPOLATED VALUE FUNCTION

3.1 Approximation of the value function by interpolation

The problem with POMDPs is that the associated value function is defined on an infinite and continuous space (the *belief state space* $\mathcal{B}$). It is a real computational challenge, and it may explain why researchers in reliability seem to have been reluctant to use such framework (until recently), or use modeling approximations that bypass this issue. Our goal with this work is also to prove that POMDP can be a relevant framework to model CBM problems

with imperfect monitoring, and that the solving step, even though requiring some approximations, can lead to very good results.

In order to apply value iteration, and because it is impossible to compute the value function on every point of the continuous *belief state space*, we need to use an approximation of V. One way to proceed, which is suggested by Papakonstantinou (2014a), is to approximate it by interpolation. It may not be the most effective state-of-the-art technique used in modern POMDP solvers (Hauskrecht (2000)), but it has the advantage of being relatively easy to implement from scratch, with good structural properties (cf. lower bound) and providing sufficiently precise results for the purpose of our work. Specifically, the belief state space $\mathcal{B}$ is discretized into a regular grid $\hat{\mathcal{B}}_{regular}$, with parameter $0 < h < 1$.

$$\hat{\mathcal{B}}_{regular} = \{b \in \mathcal{B} : \forall\, s \in \mathcal{S},\, b[s] = n_s h \text{ with } n_s \in \mathbb{N}\} \tag{15}$$

Value iteration is then performed on each point of the grid. When the value of a point outside of the grid is required in the computation, interpolation with the values of close grid points is used as an approximation. For example, if $b \notin \hat{\mathcal{B}}_{regular}$, but we have a set of points $\{b_1, b_2, ..., b_l\} \subset \hat{\mathcal{B}}_{regular}$ such that b can be expressed as a convex combination of them:

$$b = \sum_{i=1}^{l} \lambda_i b_i, \quad \text{with } 0 < \lambda_i \leq 1 \text{ and } \sum_{i=1}^{l} \lambda_i = 1 \tag{16}$$

Then, we can use the following interpolation as approximation for the value function V

$$V(b, k, \xi) \approx \sum_{i=1}^{l} \lambda_i V(b_i, k, \xi) \tag{17}$$

3.2 Dynamic grid

In the proposed approach, the design of the grid is quite crucial. If it contains too few points, the approximation will be of poor quality, but if it contains too many points, the computation time will increase significantly. For constructing a suitable grid, several approaches are possible. The most straightforward idea would be to use a fixed and regular grid, like $\hat{\mathcal{B}}_{regular}$, and to apply the value iteration algorithm on all its points. However, the structure of such grid, defined *a priori*, may not be very relevant because some areas of the belief state space may require more grid points than others in order to approximate V with a certain precision. This is why we choose to adopt a dynamic grid.

The dynamic grid we propose is based on the concept of *reachable belief points*, developed in Kurniawati (2008). By periodically running several simulations with the current policy, we identify belief points $b \in \hat{\mathcal{B}}_{regular}$ that are most likely to be reached by the policy, and dynamically add them into the grid. A selection heuristic must be used to select, among all the computed reachable belief points (that might be numerous), which ones should be included into the grid, but we do not detail it here.

Eventually, and in order to simplify the management of an irregular grid, we use Delaunay triangulation to partition the belief state space and compute interpolated values when needed. Thereby, for a dynamic grid $\hat{\mathcal{B}}_{dyn} \subset \hat{\mathcal{B}}_{regular}$, and a belief point $b \notin \hat{\mathcal{B}}_{dyn}$, the Delaunay triangulation is used to identify the belief points $\{b_1, ..., b_l\} \subset \hat{\mathcal{B}}_{dyn}$ and coefficients $(\lambda_i)_{1 \leq i \leq l}$ such that b can be expressed as a convex combination of the $(b_i)_{1 \leq i \leq l}$, as in (16). Then, the value function can be approximated by interpolation as in (17).

3.3 Bounds

What is interesting with the proposed dynamic grid-based interpolation technique is that it easily provides bounds to estimate the quality of the approached solution.

Lower bound We know from Sondik (1971) that the value function is concave (minimization here). As interpolation is a convex combination of concave functions, our approximation of the value function preserves its concavity. Moreover, provided that during the value iteration algorithm, V is initialized with a lower bound (e.g., initializing the value function at zero for our problem), the concavity of the value function combined with our approximation by convex combination guarantees that each iteration of the value iteration algorithm preserves the lower bound. Therefore, the approximated value function obtained in the end will provide a lower bound of the optimal value function.

Upper bound estimation Once we have computed an approximated value function, we propose to compute an estimation of an upper bound by simulating the obtained policy, since any feasible policy provides an upper bound on the value of the problem. The only limitation is that we cannot analytically compute the value associated with that policy, this is why we resort to Monte-Carlo simulations.

3.4 Solving algorithm by successive value iterations & simulations

In order to identify the *reachable* belief points that would be relevant to include into the dynamic grid, we adopt a two-phase optimization procedure. The idea is to alternate between a phase a) of value iteration and a phase b) of simulations. During phase a), we run the value iteration algorithm to improve the approximation of $V(b, k, \xi)$ on each point $b \in \hat{\mathcal{B}}_{dyn}$. Then, phase b) consists in running several simulations, with the current policy, in order to identify the *reachable* belief points not yet included into $\hat{\mathcal{B}}_{dyn}$. Because it would be computationally too expensive to add them all, we need to select only a few ones. Such selection is performed heuristically, and aims at identifying the most promising candidates. The heuristic procedure is not detailed here, but it basically consists in looking for the reachable belief points that will most likely improve the approximation of the value function, which would in the end result in improving the lower bound.

4. NUMERICAL RESULTS

4.1 Case study

Our case study is a rather simple maintenance problem, but we think it is a good opportunity to illustrate the gain offered by advanced PBVI solving techniques for POMDPs compared to the approximate model proposed by Byon (2010). The numerical case study is very much inspired from the magnitudes of order found in Yildirim (2017), from the wind farms industry.

Table 1. Cost parameters

$cost_{CM}$	$cost_{PM}$	$cost_I$	$cost_{OP}$
20 k€	5 k€	1 k€	1.9 k€/ failed time step

The transition matrix P is defined as follows:

$$\begin{bmatrix} 0.968 & 0.03 & 0 & 0 & 0.002 \\ 0 & 0.92 & 0.06 & 0 & 0.02 \\ 0 & 0 & 0.91 & 0.02 & 0.07 \\ 0 & 0 & 0 & 0.87 & 0.13 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Finally, we study two imperfect monitoring systems, defined by their monitoring matrix Q. It enables us to highlight the fact that the good performance and properties remain valid for different kinds of condition monitoring quality. As the contribution of this paper is mostly methodological, the transition and monitoring matrices are not derived from a specific real-world use case, but serve the purpose of illustrating the interest of the proposed framework and methods.

- Low monitoring performance

$$Q_1 = \begin{bmatrix} 0.8 & 0.13 & 0.06 & 0.01 \\ 0.17 & 0.6 & 0.2 & 0.03 \\ 0.07 & 0.21 & 0.62 & 0.1 \\ 0.01 & 0.03 & 0.2 & 0.76 \end{bmatrix}$$

- Good monitoring performance

$$Q_2 = \begin{bmatrix} 0.925 & 0.05 & 0.022 & 0.003 \\ 0.062 & 0.85 & 0.075 & 0.13 \\ 0.025 & 0.08 & 0.858 & 0.037 \\ 0.002 & 0.013 & 0.075 & 0.91 \end{bmatrix}$$

For our numerical use case, we use a discount equivalent of 20% per year, which, if we consider that one time step represents one day, leads us to define $\gamma \approx 0.99939$.

4.2 Approximate sample paths (ASP) model

Through this work, we want to analyze the performance of the approximate model (ASP) proposed by Byon (2010). To simplify the computation, the authors proposed to approximate the Bellman equation: value function V is expressed as if no monitoring were to be performed in the future. Consequently, the future evolution of the belief b is only driven by maintenance decisions and the transition matrix P; the computation of the value iteration algorithm is then much easier since the value function should only be evaluated along some specific *sample paths*. Said differently, this ASP model is an approximate model in the sense that when computing the policy, the expected cost resulting from each decision does not take into account future data acquisitions via the monitoring system. Nevertheless, the information collected by imperfect condition monitoring is still partially taken into account in the simulation procedure, where the current belief on the state of the system is periodically updated.

4.3 Comparison of the two approaches

Motivation Although the proposed approach is very much inspired from Kurniawati (2008) and the SARSOP algorithm, it cannot be really considered state-of-the-art, as we implemented a slightly simpler version. However, we think the comparison with the ASP model is of interest because it illustrates the difficult choice researchers and practitioners often have to make between an approximate model with exact solving and an exact model with approximate solving. If our PBVI algorithm were to outperform the ASP model, it would be even more true for the SARSOP algorithm, and such work would then contribute to illustrate the efficiency of nowadays state-of-the-art POMDP solvers compared to approximate models such as ASP.

Method Both approaches are evaluated on different parameters scenarios and their performances are compared via Monte-Carlo simulations. Because we are optimizing a total discounted cost over infinite time horizon, we need to simulate the maintenance policies over a sufficiently long time horizon. We choose to run each simulation on 15 000 time steps, which roughly corresponds to 40 years (1 time step = 1 day). It should be noted that both methods are not comparable in terms of simulation time. For that reason, we were able to perform 120 000 simulation samples for each scenario with PBVI, whereas only 2 000 simulation samples (approximately) could be computed in the same amount of time for the ASP method. Thus, as the number of simulation samples may differ, we computed the 95% asymptotic confidence interval in order to compare the results.

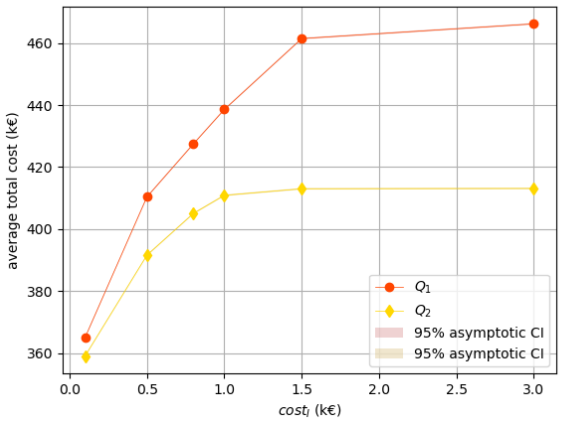

Fig. 1. Average total cost vs. $cost_I$: PBVI model

Sensitivity to the cost of inspection $cost_I$ From Fig. 1 and 2, we can observe two things. First, it was expected but the simulation confirms it, the cheaper the inspection cost and the lower the average total cost. It can be easily understood because with cheaper inspections, it

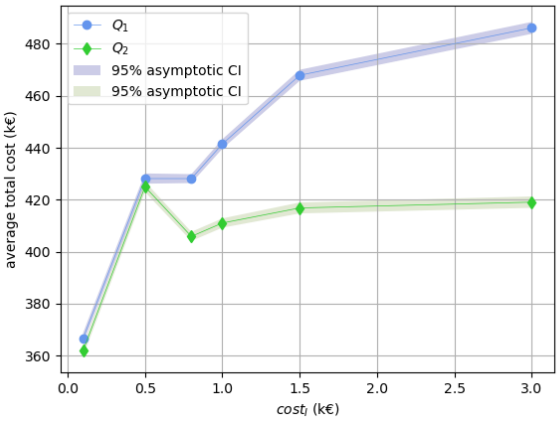

Fig. 2. Average total cost vs. $cost_I$: ASP model

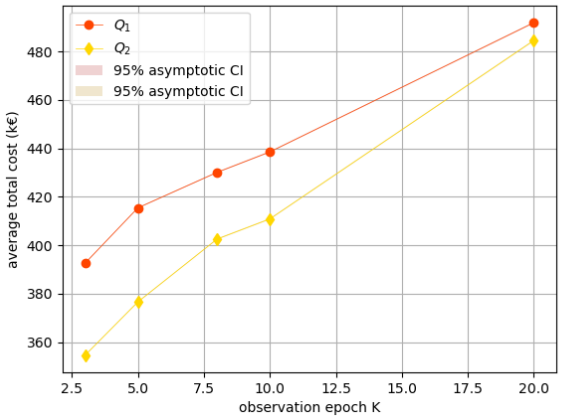

Fig. 3. Average total cost vs. K: PBVI model

becomes profitable to conduct more inspections, providing more accurate knowledge about the system condition, and leading to more effective maintenance decisions. Second, we observe that PBVI has a much more regular sensitivity to $cost_I$, whereas this total cost is not even monotonic for the ASP model. This is due to a better modeling of future monitoring observations in the value function via PBVI, leading to more accurate maintenance decisions. Said differently, for cheap values of $cost_I$, the ASP model tends to over-schedule inspections because it is myopic to future data acquisitions.

Table 2. Sensitivity to $cost_I$ (monitoring Q_1)

$cost_I$ (k€)	0.1	0.5	0.8	1	1.5	3
PBVI - average total cost (k€)	365.1	410.4	427.5	438.6	461.4	466.2
95% asympt. CI	± 0.3	± 0.3	± 0.3	± 0.3	± 0.3	± 0.3
ASP - average total cost (k€)	366.7	428.2	428.2	441.5	467.9	486.2
95% asympt. CI	± 2.0	± 2.0	± 1.7	± 1.8	± 2.1	± 2.3
Relative gap	+0.4%	+4.3%	+0.2%	+0.7%	+1.4%	+4.3%

Table 3. Sensitivity to $cost_I$ (monitoring Q_2)

$cost_I$ (k€)	0.1	0.5	0.8	1	1.5	3
PBVI - average total cost (k€)	359.1	391.6	405.1	410.9	413.0	413.1
95% asympt. CI	± 0.2	± 0.3	± 0.3	± 0.3	± 0.3	± 0.3
ASP - average total cost (k€)	361.9	425.0	405.9	411.1	416.9	419.1
95% asympt. CI	± 2.0	± 2.0	± 1.7	± 1.8	± 2.1	± 2.2
Relative gap	+0.8%	+8.5%	+0.2%	+0.0%	+0.9%	+1.4%

With Tables 2 and 3, we can notice that PBVI systematically outperforms ASP, but with a varying gap. When looking at the sensitivity to the cost of inspection, a threshold effect around $cost_I = 0.8$k€ seems to have a strong impact on the performance of the ASP method. This is a direct consequence of the sub-optimality of that approach, which ignores future monitoring data when scheduling maintenance interventions, and in particular inspections.

Sensitivity to the observation epoch K When varying the observation epoch, it can be noted that PBVI produces, once again, a much more regular curve (Fig. 3 vs. Fig. 4). There seems to be a threshold effect around $K = 5$ (Fig. 4) in the ASP model, which can be explained by the sub-optimality of such model in a context of small observation

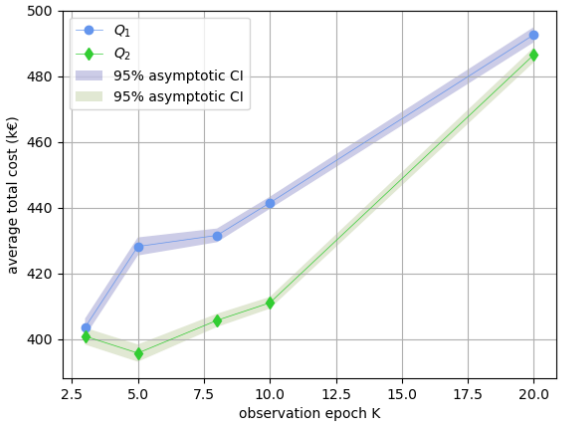

Fig. 4. Average total cost vs. K: ASP model

epoch. In particular, ASP will tend to even more over-schedule inspections when the monitoring is good, which is observed in curve Q_2 of Fig. 4. This is furthermore confirmed when looking at Tables 4 and 5, where it can be observed that the gap between the two methods increases both with the monitoring quality and small values of K.

Table 4. Sensitivity to K (monitoring Q_1)

observation epoch K	3	5	8	10	20
PBVI - average total cost (k€)	392.6	415.5	430.1	438.6	492.0
95% asympt. CI	± 0.3	± 0.3	± 0.3	± 0.3	± 0.3
ASP - average total cost (k€)	403.6	428.2	431.6	441.5	492.6
95% asympt. CI	± 2.7	± 2.7	± 2.1	± 1.8	± 2.3
Relative gap	+2.8%	+3.1%	+0.3%	+0.7%	+0.1%

Table 5. Sensitivity to K (monitoring Q_2)

observation epoch K	3	5	8	10	20
PBVI - average total cost (k€)	354.6	376.6	402.6	410.9	484.6
95% asympt. CI	± 0.2	± 0.3	± 0.3	± 0.3	± 0.3
ASP - average total cost (k€)	400.9	395.8	405.7	411.1	486.6
95% asympt. CI	± 2.5	± 2.6	± 2.0	± 1.8	± 2.3
Relative gap	+13.1%	+5.1%	+0.8%	+0.0%	+0.4%

4.4 Satisfying precision given by bounds

The advantage of our proposed PBVI method compared to ASP is that it provides a lower bound (LB). Consequently, in addition with the upper bound (UB) estimated from Monte-Carlo simulations, the PBVI method can provide an estimation of the gap between the current solution and the optimal solution. Fig. 5 and 6 show the relative gap between the LB and UB estimation obtained for the two sensitivity analysis. In all the tested parameters scenarios, the proposed method has a very good performance, providing a policy that is associated to an expected cost at most 0.6% more expensive than the expected cost of the optimal policy (and on average, this relative gap is about +0.1%).

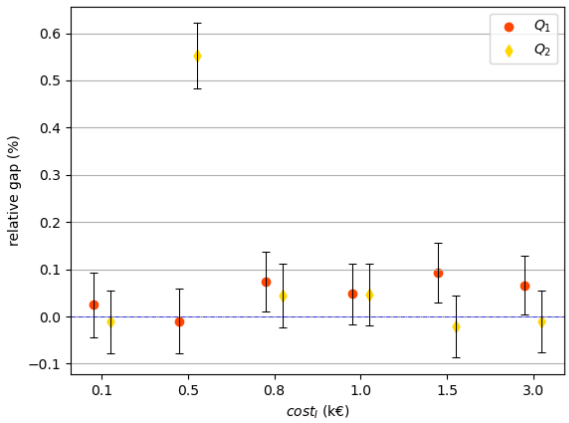

Fig. 5. Relative gap between LB and UB estimation (with 95% asympt. CI): sensitivity to $cost_I$

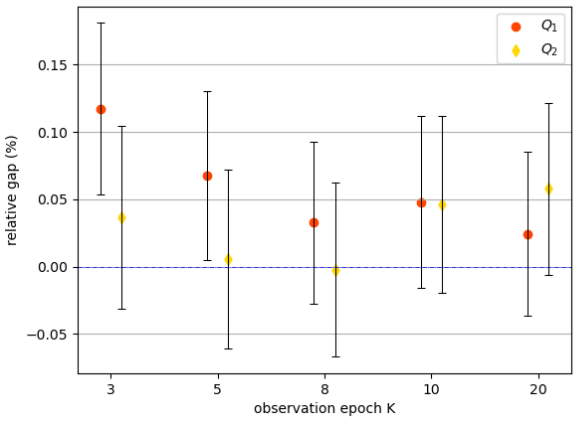

Fig. 6. Relative gap between LB and UB estimation (with 95% asympt. CI): sensitivity to K

5. CONCLUSION

This work applied POMDP to a very classical maintenance planning problem, showing that it constitutes a quite efficient framework to deal with imperfect monitoring information. Moreover, we successfully extended it to include additional constraints limiting the frequency of observation or restraining the availability of maintenance resources. The proposed point-based value iteration algorithm coupled with approximate interpolated value functions gave very good results. One of the advantages of the proposed method is that it provides a lower bound to estimate the quality of the results. Eventually, the comparison with the approximate *sample paths* (ASP) model proposed by Byon (2010) showed that, in a large variety of parameters scenarios, the use of that approximation is clearly sub-optimal and the PBVI algorithm can provide better results with guarantees of performance. As a conclusion, we estimate that efficient POMDP solving algorithms exist now for single-item systems, and aligned with the suggestions from de Jonge (2020), we suggest to extend the proposed work to optimize condition-based policies for multi-items systems imperfectly monitored.

REFERENCES

Barlow, R., & Hunter, L. (1960). Optimum preventive maintenance policies. *Operations research*, 8(1), 90–100.

Bajestani, M. A., & Banjevic, D. (2016). Calendar-based age replacement policy with dependent renewal cycles. *IIE Transactions*, 48(11), 1016–1026.

Alaswad, S., & Xiang, Y. (2017). A review on condition-based maintenance optimization models for stochastically deteriorating system. *Reliability engineering & system safety*, 157, 54–63.

de Jonge, B., & Scarf, P. A. (2020). A review on maintenance optimization. *European journal of operational research*, 285(3), 805–824.

Sondik, E. J. (1971). The optimal control of partially observable Markov processes. *Stanford University*.

Ghasemi, A., Yacout, S., & Ouali, M. S. (2007). Optimal condition based maintenance with imperfect information and the proportional hazards model. *International journal of production research*, 45(4), 989–1012.

Papakonstantinou, K. G., & Shinozuka, M. (2014a). Planning structural inspection and maintenance policies via dynamic programming and Markov processes. Part I: Theory. *Reliability Engineering & System Safety*, 130, 202–213.

Papakonstantinou, K. G., & Shinozuka, M. (2014b). Planning structural inspection and maintenance policies via dynamic programming and Markov processes. Part II: POMDP implementation. *Reliability Engineering & System Safety*, 130, 214–224.

Byon, E., & Ding, Y. (2010). Season-dependent condition-based maintenance for a wind turbine using a partially observed Markov decision process. *IEEE Transactions on Power Systems*, 25(4), 1823–1834.

Yildirim, M., Gebraeel, N. Z., & Sun, X. A. (2017). Integrated predictive analytics and optimization for opportunistic maintenance and operations in wind farms. *IEEE Transactions on power systems*, 32(6), 4319–4328.

Hauskrecht, M. (2000). Value-function approximations for partially observable Markov decision processes. *Journal of artificial intelligence research*, 13, 33–94.

Kurniawati, H., Hsu, D., & Lee, W. S. (2008). SARSOP: Efficient point-based POMDP planning by approximating optimally reachable belief spaces. In *Robotics: Science and systems*, 2008.