Proceedings of the 35th European Safety and Reliability & the 33rd Society for Risk Analysis Europe Conference
Edited by Eirik Bjorheim Abrahamsen, Terje Aven, Frederic Bouder, Roger Flage, Marja Ylönen
©2025 ESREL SRA-E 2025 Organizers. *Published by* Research Publishing, Singapore.
doi: 10.3850/978-981-94-3281-3_ESREL-SRA-E2025-P7937-cd

Condition-Based Maintenance for Large-Scale Fleets under Multiple Constraints: A Constrained MDP Model with Primal-Dual Solution

Zehui Xuan

Chair on Risk and Resilience of Complex Systems, Laboratoire Génie Industriel, CentraleSupélec, Université Paris-Saclay, France. E-mail: zehui.xuan@centralesupelec.fr

Yiping Fang

Chair on Risk and Resilience of Complex Systems, Laboratoire Génie Industriel, CentraleSupélec, Université Paris-Saclay, France. E-mail: yiping.fang@centralesupelec.fr

Adam Abdin

Laboratoire Génie Industriel, CentraleSupélec, Université Paris-Saclay, France.
E-mail: adam.abdin@centralesupelec.fr

Anne Barros

Chair on Risk and Resilience of Complex Systems, Laboratoire Génie Industriel, CentraleSupélec, Université Paris-Saclay, France. E-mail: anne.barros@centralesupelec.fr

Managing maintenance activities for large-scale fleets, such as wind farms with numerous wind turbines, presents a significant challenge in condition-based maintenance. In addition to the curse of dimensionality inherent to optimizing dynamic decisions for large systems, prior research has primarily concentrated on individual modeling challenges, such as limited maintenance resources or overall system performance requirements, without fully addressing the need for a comprehensive solution that accounts for both dimensions. In this article, we propose a novel approach in the context of condition-based maintenance planning that integrates all three critical factors: system scale, resource limitations, and performance constraints. Specifically, we develop a constrained multi-agent Markov Decision Process (MDP) model to tackle the maintenance planning problem for a multi-component system, and we solve it using a Primal-Dual algorithm. The system includes more than 50 components with known transition dynamics. At each time step, the planner must decide whether to replace each component, balancing limited maintenance resources with stringent availability requirements. The goal is to find an optimal policy that minimizes the expected discounted maintenance cost while adhering to these constraints. Finally, we compare our method's performance against baseline approaches, demonstrating its ability to achieve superior trade-offs between cost and constraint satisfaction.

Keywords: Condition-based maintenance, Constrained Markov Decision Process (CMDP), Large-scale fleet systems, Resource constraint, Availability constraint, Primal-Dual approach.

1. Introduction

In many industrial applications, numerous scenarios require the planning and coordination of maintenance activities on a large ensemble of independent and heterogeneous facilities, which can be conceptualized as a fleet, such as a wind farm with multiple turbines, a fleet of aircraft, a railway system or a data center with numerous servers. The failure of facilities can harm system performance and potentially endanger human lives. However, overly frequent maintenance can lead to high costs, highlighting the need for a well-balanced maintenance policy.

Managing the maintenance of such an extensive system is challenging, as the number of possible states and actions grows exponentially with the number of facilities, which makes it difficult for traditional methods to compute maintenance planning decisions within a reasonable time frame. Furthermore, these decisions should be made subject to multiple constraints, such as limited maintenance resources and system-level performance requirements, further complicating the decision problem. An adapted framework is, therefore, needed to model and optimize maintenance planning for large-scale fleet systems effectively.

We propose to model this problem in the Markov Decision Processes (MDP) framework to address this challenge because of its sequential decision-making nature. Numerous studies (Zhou et al., 2022; Xu et al., 2024; Arcieri et al., 2024) have developed methods based on MDP for condition-based maintenance (CBM) problems. The first article that details Constrained MDP (CMDP) is Altman (1999). The author pointed out that such a model facilitates the modeling of sequential decision-making problems with multiple objectives, which is ideal for obtaining maintenance decisions that reduce maintenance costs while balancing maintenance resources and fleet performance in a dynamic setting.

Limited research has focused on CBM problems utilizing CMDP. Xu et al. (2022) propose a risk-aware maintenance model, which guarantees the system's safety level when optimizing the maintenance strategy by introducing risk metrics (e.g., Value-at-Risk and Conditional Value-at-Risk) as constraints. However, this model treats the system as an indivisible whole, so how it describes the states and actions of the system does not apply to the large, heterogeneous fleet. This model also does not take into account the maintenance resource constraints.

Other works proposed frameworks for maintenance
optimization problems for large multi-component systems that consider other constraints. Bansal et al. (2025) propose a Component-Wise MDP and Adjusted Component-Wise MDP approach for solving condition-based maintenance problems for large heterogeneous systems with economic dependencies. However, the authors do not consider the system's performance and resource constraints. Some other works (Glazebrook et al., 2005; Cho et al., 2015; Abbou and Makis, 2019; Ruiz-Hernández et al., 2020; Demirci et al., 2024) have used Restless Multi-Armed Bandit (RMAB), which allows one to consider hard constraints on the available maintenance resources. In this study, the soft constraint we apply to maintenance resources corresponds to the relaxed form of the hard constraint in the RMAB framework. However, these papers do not consider the system's performance requirements.

In this work, we consider a sequential decision-making problem regarding the maintenance of a heterogeneous large fleet, accounting for the system performance requirements and the constraints of limited maintenance resources. We describe the problem as a Constrained MDP containing N sub-problems, each of which can be considered as an independent MDP, but system-level constraints couple the N sub-problems. We utilize the primal-dual approaches in Chen et al. (2024) and Moskovitz et al. (2023) to obtain an optimal policy for each sub-problem of the system. We demonstrate through a small numerical experiment that the policies obtained by the method in this paper are very close to the theoretical optimum. We then conduct a large-scale numerical experiment and compare the results with several classical CBM benchmarks to demonstrate the advantages of our approach. To the best of our knowledge, this study is the first to propose a maintenance optimization framework for large-scale fleets under resource and performance constraints in a tractable way.

This article presents the following structure. In section 2, we describe the maintenance problem and its formulation. Section 3 introduces the Primal-Dual Algorithms and a modified version with Optimistic Ascent-Descent. Section 4 illustrates the numerical experiments of two sizes of instances of fleet. Section 5 concludes our work and provides some ideas for future work.

2. Problem Description

2.1. *A multi-component System*

The system we study comprises N components, which may differ from one another, with perfect monitoring. Functionality and degradation processes are independent between components. According to the monitoring information, the system planner decides which components to replace at every time step. Replacing a component is an immediate process that engages one unit of maintenance resource for a single time step.

There are two constraints to this problem. (i) Limited maintenance resources: On average, M units of maintenance resources are available per time step. (ii) Performance requirement: The system's average number of functioning machines should be more than K.

A component's degradation is similar to the process described in Roux et al. (2022). Assuming that each component in the system has only one failure mode, we consider a discrete-time and finite-state Markov chain to describe the degradation level. We abstract the state of a component into L levels, $L \geq 2$. The set of component's state is $\{0, 1, ..., L-1\}$, where the smaller the number indicates, the lower the degradation level. 0 and $L-2$ stand for as-good-as-new state and the most degraded functioning state respectively, and $L-1$ stands for the failed state. A component may degrade and move to the next more degraded state within a given time step or go to the failure state; otherwise, it will remain in its current state. Moreover, there are two additional settings for this Markov chain. (i) The component cannot spontaneously transit from a more degraded state to a healthier state. (ii) The more degraded the component is, the higher the probability it will go directly to the failure state.

We aim to find a maintenance policy for this system to minimize the total maintenance cost while satisfying resource constraints and performance requirements.

2.2. *Formulation*

We model the system using the Constrained Markov Decision Processes (CMDP) framework, where each component functions as an independent MDP, interconnected by two soft constraints. Consider an MDP $(\mathcal{S}, \mathcal{A}, P^a, r, \{c_i\}_{i=1,2}, \gamma)$. Let $\mathcal{S} = \{0, 1, ..., L-1\}$ be the state space of the component. $\mathcal{A} = \{0, 1\}$ is the action space of the component, where 1 represents the active action (replacement), while 0 represents the passive action (do nothing). P^a is the transition probability matrix under action $a \in \mathcal{A}$. Let $p(s'|s, a)$ denote a transition probability for a component from state s to s' under action a, where $s, s' \in \mathcal{S}$. Then we have $P^a = [p(s'|s, a)]_{s,s' \in \mathcal{S}}$. For example, when $L = 3$, we have

$$P^0 = \begin{bmatrix} 1 - p_{01} - p_{02} & p_{01} & p_{02} \\ 0 & 1 - p_{12} & p_{12} \\ 0 & 0 & 1 \end{bmatrix},$$

$$P^1 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}.$$

In addition, we have $p_{02} < p_{12}$ to meet the second degradation setting described in the previous section (2.1). r is a reward function of a state-action pair, which can be defined as follows.

$$r(s, a) = \begin{cases} 0 & \text{if } a = 0 \text{ and } s \in \mathcal{S} \backslash \{L-1\} \\ -c_{OP} & \text{if } a = 0 \text{ and } s = L-1 \\ -c_R & \text{if } a = 1 \end{cases}.$$

$\{c_i\}_{i=1,2}$ is a collection of auxiliary cost functions of a state-action pair in the constraints. More precisely, c_1 corresponds to the constraint of limited maintenance resources,

$$c_1(s, a) = a.$$

c_2 corresponds to the performance requirement, which indicates whether a component is in the failed state,

$$c_2(s, a) = \mathbb{1}_{\{s = L-1\}}.$$

γ is the discount factor. Let π denote a stochastic policy at the component level. The occupancy measure of a state-action pair induced by a policy π is defined as follows.

$$x^\pi(s, a) = (1-\gamma) \cdot \mathbb{E}_{\tau \sim \pi} \left[\sum_{t=0}^{\infty} \gamma^t \mathbb{1}_{\{S_t = s, A_t = a\}} \right],$$

where τ is the trajectory of the component.

At the system level, let $\mathscr{S} = \times_{n=1}^{N} \mathcal{S}_n$ denote a state space of the system and $\boldsymbol{s}_t = (s_{1,t}, ..., s_{N,t}) \in \mathscr{S}$ denote a state of the system at time step t. Let $\mathscr{A} = \times_{n=1}^{N} \mathcal{A}_n$ denote the action space of the system, and $\boldsymbol{a}_t = (a_{1,t}, ..., a_{N,t}) \in \mathscr{A}$ denote an action of the system at time step t. The reward of the system at a specific time step t is the sum of the reward of each component, i.e. $\boldsymbol{r}(\boldsymbol{s}_t, \boldsymbol{a}_t) = \sum_{n=1}^{N} r(s_{n,t}, a_{n,t})$. Furthermore, the transition probability at the system level is the product of the transition probability of components: $\boldsymbol{P}^{\boldsymbol{a}} = \left[\prod_{n=1}^{N} p(s'_n|s_n, a_n)\right]_{\boldsymbol{s},\boldsymbol{s}'\in\mathscr{S}}$. d_i is the bound of the constraint i. Specifically, $d_1 = M$, $d_2 = N - K$. Let $\boldsymbol{\pi} = (\pi_1, ..., \pi_N)$ be a stochastic policy at the system level.

Let $J_{\boldsymbol{r}}(\boldsymbol{\pi})$ be the total expected discounted reward induced by policy $\boldsymbol{\pi}$. The following expression gives its definition.

$$J_{\boldsymbol{r}}(\boldsymbol{\pi}) \\ = (1-\gamma)\cdot \mathbb{E}_{\tau\sim\boldsymbol{\pi}}\left[\sum_{t=0}^{\infty}\gamma^t \sum_{n=1}^{N} r_n(S_{n,t}, A_{n,t})\right],$$

where τ is the system's trajectory, and $S_{n,t}$ and $A_{n,t}$ are the random variables of state and action of component n, respectively, at time step t. Similarly, the following expression gives the i-th total expected discounted auxiliary cost under policy $\boldsymbol{\pi}$.

$$J_{\boldsymbol{c}_i}(\boldsymbol{\pi}) \\ = (1-\gamma)\cdot \mathbb{E}_{\tau\sim\boldsymbol{\pi}}\left[\sum_{t=0}^{\infty}\gamma^t \sum_{n=1}^{N} c_{i,n}(S_{n,t}, A_{n,t})\right].$$

The auxiliary cost is related to the model's maintenance resources and performance requirements. The $J_{\boldsymbol{c}_1}(\boldsymbol{\pi})$ and $J_{\boldsymbol{c}_2}(\boldsymbol{\pi})$ represent the average number of resources used by the system and the average number of failed components, respectively.

Then, we can express our problem as

$$\begin{aligned} &\max_{\boldsymbol{\pi}} J_{\boldsymbol{r}}(\boldsymbol{\pi}) \\ \text{s.t.}\quad & J_{\boldsymbol{c}_i}(\boldsymbol{\pi}) \le d_i, \quad i = 1, 2. \end{aligned} \tag{1}$$

2.3. *Lagrangian Relaxation*

In order to solve this constrained optimization problem, we perform a Lagrangian relaxation on 1 (Adelman and Mersereau, 2008). We put the two constraints into the objective function and get the Lagrangian:

$$\begin{aligned} L(\boldsymbol{\pi}, \lambda) &= J_{\boldsymbol{r}}(\boldsymbol{\pi}) + \sum_{i=1}^{2}\lambda_i(d_i - J_{\boldsymbol{c}_i}(\boldsymbol{\pi})) \\ &= \sum_{i=1}^{2}\lambda_i d_i + \sum_{n=1}^{N} J_n^{\lambda}(\pi_n), \end{aligned} \tag{2}$$

where

$$J_n^{\lambda}(\pi) = (1-\gamma)\cdot\mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t r_n^{\lambda}(s,a)\right], \tag{3}$$

and

$$r_n^{\lambda}(s,a) = r_n(s,a) - \sum_{i=1}^{2}\lambda_i c_{i,n}(s,a). \tag{4}$$

Thus, the problem 1 takes the following equivalent form:

$$\inf_{\lambda}\sup_{\boldsymbol{\pi}} L(\boldsymbol{\pi}, \lambda). \tag{5}$$

Then, the original problem splits into N subproblems, each of which, under a fixed λ, can be seen as a modified unconstrained MDP with instantaneous reward r_n^{λ}.

3. Primal-Dual Approach

3.1. *Primal-Dual Algorithm to CMDPs*

Classical methods for solving a CMDP rely on linear programming; however, it is intractable for

Algorithm 1 Primal-Dual Algorithm to CMDPs (Chen et al., 2024)

```
Input: step size $\eta$, projection area $\Lambda_M$.
  Initialize $\pi_{n,0}$ for $n \in \{1,...,N\}$; $\lambda_{i,0}$ for $i \in \{1,2\}$.
  for $t = 1, 2, ..., T$ do
      Compute the occupancy measure $x^{\pi_{n,t-1}}, n \in \{1,...,N\}$ by solving the linear system 6.
      Compute the total expected discounted auxiliary costs $J_{\boldsymbol{c}_i}(\boldsymbol{\pi}_{t-1}), i \in \{1,2\}$ via equation 7.
      Compute the $Q^{\pi_{n,t-1},\lambda_{t-1}}$ via solving the linear system based on equation 8.
      Update $\lambda$ via equation 10.
      Update $\boldsymbol{\pi}$ via equation 9.
  end for
  $\bar{\lambda} = \frac{1}{T}\sum_{t=1}^{T}\lambda_t$.
  $\bar{x}_n = \frac{1}{T}\sum_{t=1}^{T}x^{\pi_{n,t}}, \forall n \in \{1,...,N\}$.
  $\bar{\pi}_n(\cdot|s) = \frac{\bar{x}_n(s,\cdot)}{\sum_{a\in\mathcal{A}}\bar{x}_n(s,a)}, \forall n \in \{1,...,N\}, s \in \mathcal{S}$.
Output: average policy $\bar{\boldsymbol{\pi}}$, average dual variables $\bar{\lambda}$.
```

a CMDP consisting of many sub-problems as the state space and action space are too large. Therefore, we use the primal-dual approach proposed in Chen et al. (2024), which allows the two groups of variables in $L(\boldsymbol{\pi}, \lambda)$ to be updated alternately and eventually find the saddle point by averaging the trajectory of primal and dual variables. We update λ using traditional gradient descent and update $\boldsymbol{\pi}$ with a single Regularized Policy Iteration (RPI) step. Since the update in the value of λ in each iteration changes the equivalent N modified unconstrained MDPs of the inner problem in 5, performing only one step of the RPI instead of the complete RPI reduces the computational burden. It is worth mentioning that we cannot simply average the resulting policies at each iteration. Instead, we first average the occupancy measures corresponding to each policy in each iteration and then derive the policy from the resulting average occupancy measure.

More specifically, in each iteration of the primal-dual algorithm, we first compute the occupation measure of each component corresponding to the current policy by solving a linear system.

$$\begin{cases} \dfrac{x^{\pi}(s,a)}{\sum_{a' \in \mathcal{A}} x^{\pi}(s,a')} = \pi(a|s), \\ \forall s \in \mathcal{S}, a \in \mathcal{A}; \\ \displaystyle\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} x^{\pi}(s,a)(\mathbb{1}_{\{s=s'\}} - \gamma P(s'|s,a)) \\ = (1-\gamma)\mu_0(s'), \forall s' \in \mathcal{S}. \end{cases} \tag{6}$$

Then, we compute the value of the total expected discounted auxiliary costs according to the following equation.

$$J_{\boldsymbol{c}_i}(\boldsymbol{\pi}) = \sum_{n=1}^{N} \sum_{s\in\mathcal{S}} \sum_{a\in\mathcal{A}} x^{\pi_n}(s,a) c_{i,n}(s,a) \tag{7}$$

We also compute the $Q^{\pi,\lambda}$ under the current policy, which is the state-action value of each of the N modified unconstrained MDPs equivalent to the inner problem of Problem 5. It is defined as follows:

$$\begin{aligned} &Q^{\pi,\lambda}(s,a) \\ &= (1-\gamma) \cdot \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty} \gamma^t r^{\lambda}(S_t, A_t)|s,a\right]. \end{aligned}$$

We compute the value of $Q^{\pi,\lambda}$ by solving the following linear system.

$$\begin{aligned} &Q^{\pi,\lambda}(s,a) \\ &= (1-\gamma) r^{\lambda}(s,a) \\ &+ \gamma \sum_{s'\in\mathcal{S}} \sum_{a'\in\mathcal{A}} P(s'|s,a)\pi(a'|s')Q^{\pi,\lambda}(s',a'), \\ &\forall s \in \mathcal{S}, a \in \mathcal{A}. \end{aligned} \tag{8}$$

After calculating the necessary values, the next step is to update the policy and dual variable. Let t be the current iteration number, and $\boldsymbol{\pi}_{t-1}$ and λ_{t-1} be the results of the previous iteration. We update the policy with the following equation.

$$\begin{aligned} \pi_{n,t}(\cdot|s) \leftarrow & Z_{t-1}^{-1} \pi_{n,t-1}(\cdot|s) \exp Q^{\pi_{n,t-1},\lambda_{t-1}}(s,\cdot), \\ & \forall n \in \{1,...,N\}, s \in \mathcal{S} \end{aligned} \tag{9}$$

where $Z_{t-1} = \sum_{a\in\mathcal{A}} \pi_{n,t-1}(a|s) \cdot \exp Q^{\pi_{n,t-1},\lambda_{t-1}}(s,a)$ is a normalizing factor.

The update of λ is as follows.

$$\lambda_t \leftarrow \text{Proj}_{\Lambda_M}\{\lambda_{t-1} - \eta(\partial_\lambda L(\pi_{t-1}, \lambda_{t-1}))\}, \tag{10}$$

where $\Lambda_M = \{\lambda | \lambda \in \mathbb{R}_+^2, \|\lambda\| \le M\}$ is a projection area which guarantees that the dual variable λ is bounded and positive. $[\partial_\lambda L(\pi, \lambda)]_i = d_i - J_{\boldsymbol{c}_i}(\boldsymbol{\pi})$ is the distance between the bound and the total expected discounted auxiliary cost in the constraints.

After completing all primal-dual updates, we output the average dual variable and the average policy derived from the trajectory. The complete procedure is presented in Algorithm 1.

3.2. *Policy-based Reinforcement Learning with Optimistic Ascent-Descent (ReLOAD)*

Although Algorithm 1 can finally get the optimal policy, it needs to average the trajectory, which means that the policy obtained by the last iteration is not necessarily optimal. Next, we take the idea of Optimistic Optimization from Moskovitz et al. (2023) to achieve last-iterate convergence.

We achieve this change in each iteration by replacing the gradient g_t with the optimistic gradient $\tilde{g}_t$, where $\tilde{g}_t = 2g_t - g_{t-1} = g_t + (g_t - g_{t-1})$.

With this substitution, the trajectory in the primal-dual algorithm no longer oscillates back and forth within the same interval. Instead, the oscillation region gradually shrinks toward the optimum and eventually converges.

Specifically, we need to change 10 and 9 as follows.

$$\begin{aligned}\lambda_t \leftarrow & \mathrm{Proj}_{\Lambda_M}\{\lambda_{t-1} - \eta(2\partial_\lambda L(\pi_{t-1}, \lambda_{t-1}) - \\ & \partial_\lambda L(\pi_{t-2}, \lambda_{t-2}))\},\end{aligned} \tag{11}$$

$$\begin{aligned}\pi_{n,t}(\cdot|s) \leftarrow & Z_{t-1}^{-1}\pi_{n,t-1}(\cdot|s)\exp(2Q^{\pi_{n,t-1},\lambda_{t-1}}(s,\cdot) \\ & - Q^{\pi_{n,t-2},\lambda_{t-2}}(s,\cdot)), \\ & \forall n \in \{1,...,N\}, s \in \mathcal{S},\end{aligned} \tag{12}$$

where $Z_{t-1} = \sum_{a\in\mathcal{A}} \pi_{n,t-1}(a|s) \cdot \exp(2Q^{\pi_{n,t-1},\lambda_{t-1}}(s,a) - Q^{\pi_{n,t-2},\lambda_{t-2}}(s,a))$ is a normalizing factor. Upon completing all primal-dual updates, we output the dual variable and policy of the last iteration. The entire procedure is illustrated in Algorithm 2.

4. Numerical results

4.1. *A 3-component system*

First, we consider a fleet comprising three different components with three state levels. The degradation transition matrix of the three components is as follows:

$$P_1^0 = \begin{bmatrix} 0.5 & 0.3 & 0.2 \\ 0 & 0.7 & 0.3 \\ 0 & 0 & 1 \end{bmatrix}, \quad P_2^0 = \begin{bmatrix} 0.5 & 0.3 & 0.2 \\ 0 & 0.5 & 0.5 \\ 0 & 0 & 1 \end{bmatrix},$$
$$P_3^0 = \begin{bmatrix} 0.3 & 0.3 & 0.4 \\ 0 & 0.5 & 0.5 \\ 0 & 0 & 1 \end{bmatrix}. \tag{13}$$

Algorithm 2 Policy-based Reinforcement Learning with Optimistic Ascent-Descent (ReLOAD) (Moskovitz et al., 2023)

```
Input: step size $\eta$, projection area $\Lambda_M$.
  Initialize $\pi_{n,-1}, \pi_{n,0}$ for $n \in \{1,...,N\}$; $\lambda_{i,-1}, \lambda_{i,0}$ for $i \in \{1,2\}$
  for $t = 1, 2, ..., T$ do
      Compute the occupancy measure $x^{\pi_{n,t-1}}, n \in \{1,...,N\}$ by solving the linear system 6.
      Compute the total expected discounted auxiliary costs $J_{\boldsymbol{c}_i}(\boldsymbol{\pi}_{t-1}), i \in \{1,2\}$ via equation 7.
      Compute the $Q^{\pi_{n,t-1},\lambda_{t-1}}$ via solving the linear system based on equation 8.
      Update $\lambda$ via equation 11.
      Update $\boldsymbol{\pi}$ via equation 12.
  end for
Output: policy $\boldsymbol{\pi}_T$, dual variables $\lambda_T$.
```

Table 1 lists the other parameters used in the experiments.

Table 1. Experiment Parameters

Description	Value
Opportunity cost c_{OP}	2
Replacement cost c_R	1
Maintenance resource M	1.5
Performance requirement K	2.4

We used linear programming-based methods to obtain theoretical values for dual variables and occupation measures. Next, we ran 3×10^5 iterations using the primal-dual and ReLOAD algorithms.

Figure 1 illustrates the trajectory of the second dual variable associated with the performance requirements over the first 1.5×10^5 iterations. We do not show the dual variable associated with the maintenance resource constraint because, in this instance, the maintenance resource is relatively sufficient, so the corresponding total expected discounted auxiliary cost does not reach the set bound. We can see that the trajectory of the second dual variable in the primal-dual oscillates in the same interval all the time. Therefore, the results of dual variables must be averaged before output. The trajectory of the dual variable obtained by ReLOAD is gradually converging. The output of both algorithms eventually converges to the theoretical value.

Figure 2 shows the heat map of the theoretical value of the occupancy measure and the computation results of the two algorithms. We can see that the output occupancy measure of the two algorithms is very close to the theoretical value.

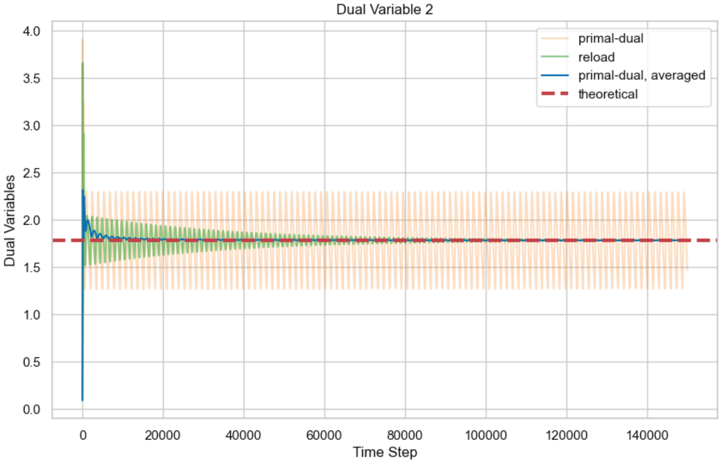

Fig. 1. Trajectories of 2nd dual variable

4.2. *A 50-component system*

Next, we consider a fleet consisting of five types of components, each with a distinct transition matrix and ten components per type. Each component has five possible states. The transition matrices and maintenance costs are randomly generated while adhering to the specifications described in Section 2.1. We set the constraint bounds such that the average usage of maintenance resources does not exceed 25 and the average number of failed machines does not exceed 10.

Table 2. Results

Policy	J_r	J_{c_1} (≤ 25)	J_{c_2} (≤ 10)
PD	-20.57	*14.77*	*9.53*
ReLOAD	-20.57	*14.77*	*9.53*
T1	-19.25	*13.34*	10.66
T2	-17.53	*12.28*	11.97
T3	-17.34	*12.17*	12.13
T4	-17.32	*12.16*	12.16

Note: PD and ReLOAD represent the stochastic policy obtained by the primal-dual and ReLOAD algorithms. T1, T2, T3, and T4 represent the policy with state thresholds of 1, 2, 3, and 4, respectively. Italicized numbers indicate that the result satisfies the constraint. The upper bounds of the constraints are in parentheses.

We computed 3×10^5 iterations using the primal-dual algorithm and ReLOAD algorithm, respectively. In this example, the system has a state space of size $5^{50} = 8.88 \times 10^{34}$ and an action space of size $2^{50} = 1.13 \times 10^{15}$. Solving this problem using LP would require 10^{50} variables and 8.88×10^{34} constraints, making it computationally intractable. Therefore, we adopt a threshold-based heuristic policy for comparison. Expressly, we set the threshold to $1, ..., L - 1$, and a component is immediately replaced once its state reaches or exceeds the specified threshold. We simulated 100 episodes of length 200 with different random seeds for each policy, calculated their total empirical discount reward and total empirical discount auxiliary costs, and averaged the results. Table 2 presents the results. As can be seen from the results, both algorithms used in this paper produce policies that can conform to both constraints. However, the threshold-based algorithm cannot do so.

5. Conclusion

This paper investigated the condition-based maintenance (CBM) problem for a heterogeneous fleet under limited maintenance resources and specific performance requirements. To address this challenge, we proposed a Constrained Markov Decision Process (CMDP)-based model that efficiently

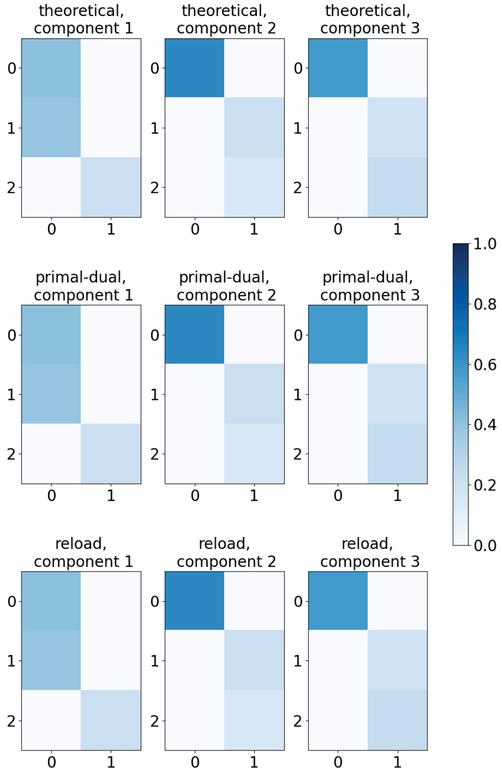

Fig. 2. Theoretical occupancy measure versus occupancy measure computed by primal-dual and ReLOAD algorithms

balances system performance and resource constraints. Moreover, we exploit the primal-dual approach to compute the optimal policy, overcoming the curse of dimensionality. The results demonstrate that, within our proposed framework, the obtained maintenance policy converges to the optimal value and satisfies all constraints. In contrast, the threshold-based heuristic approach cannot guarantee constraint satisfaction. In future work, we aim to recast the soft constraint of limited resources into a hard constraint and consider economic dependency between components.

Acknowledgement

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

References

Abbou, A. and V. Makis (2019, October). Group Maintenance: A Restless Bandits Approach. *INFORMS Journal on Computing 31*(4), 719–731.

Adelman, D. and A. J. Mersereau (2008, June). Relaxations of Weakly Coupled Stochastic Dynamic Programs. *Operations Research 56*(3), 712–727.

Altman, E. (1999). *Constrained Markov Decision Processes*. Stochastic Modeling. Boca Raton, Fla.: Chapman & Hall.

Arcieri, G., C. Hoelzl, O. Schwery, D. Straub, K. G. Papakonstantinou, and E. Chatzi (2024, May). POMDP inference and robust solution via deep reinforcement learning: An application to railway optimal maintenance. *Machine Learning*.

Bansal, V., Y. Chen, and S. Zhou (2025, February). Component-wise Markov decision process for solving condition-based maintenance of large multi-component systems with economic dependence. *IISE Transactions 57*(2), 158–171.

Chen, Y., J. Dong, Z. Wang, and C. Zhang (2024). A primal-dual approach to constrained markov decision processes with applications to queue scheduling and inventory management. forthcoming in Management Science.

Cho, P., V. Farias, J. Kessler, R. Levi, T. Magnanti, and E. Zarybnisky (2015). Maintenance and flight scheduling of low observable aircraft. *Naval Research Logistics (NRL) 62*(1), 60–80.

Demirci, E. Z., J. Arts, and G.-J. van Houtum (2024, June). A restless bandit approach for capacitated condition based maintenance scheduling. *Flexible Services and Manufacturing Journal*.

Glazebrook, K. D., H. M. Mitchell, and P. S. Ansell (2005, August). Index policies for the maintenance of a collection of machines by a set of repairmen. *European Journal of Operational Research 165*(1), 267–284.

Moskovitz, T., B. O'Donoghue, V. Veeriah, S. Flennerhag, S. Singh, and T. Zahavy (2023). ReLOAD: Reinforcement Learning with Optimistic Ascent-Descent for Last-Iterate Convergence in Constrained MDPs. In *Proceedings of the 40 Th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.*, Honolulu, Hawaii, USA.

Roux, M., Y. Fang, and A. Barros (2022). Maintenance planning under imperfect monitoring: an efficient pomdp model using interpolated value function. *IFAC-PapersOnLine 55*(16), 128–135. 18th IFAC Workshop on Control Applications of Optimization CAO 2022.

Ruiz-Hernández, D., J. M. Pinar-Pérez, and D. Delgado-Gómez (2020, July). Multi-machine preventive maintenance scheduling with imperfect interventions: A restless bandit approach. *Computers & Operations Research 119*, 104927.

Xu, J., B. Liu, X. Zhao, and X.-L. Wang (2024). Online reinforcement learning for condition-based group maintenance using factored Markov decision processes. *European Journal of Operational Research 315*(1), 176–190.

Xu, J., X. Zhao, and B. Liu (2022, November). A risk-aware maintenance model based on a constrained Markov decision process. *IISE Transactions 54*(11), 1072–1083.

Zhou, Y., B. Li, and T. Lin (2022). Maintenance optimisation of multicomponent systems using hierarchical coordinated reinforcement learning. *Reliability Engineering and System Safety 217*.