



HAL
open science

A Decomposition Method by Interaction Prediction for the Optimization of Maintenance Scheduling

Thomas Bittar, Pierre Carpentier, Jean-Philippe Chancelier, Jérôme Lonchamp

► **To cite this version:**

Thomas Bittar, Pierre Carpentier, Jean-Philippe Chancelier, Jérôme Lonchamp. A Decomposition Method by Interaction Prediction for the Optimization of Maintenance Scheduling. 2020. hal-02489304v1

HAL Id: hal-02489304

<https://hal.science/hal-02489304v1>

Preprint submitted on 24 Feb 2020 (v1), last revised 4 May 2021 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Decomposition Method by Interaction Prediction for the Optimization of Maintenance Scheduling

T. Bittar^{*,‡} P. Carpentier[†] J-Ph. Chancelier^{*} J. Lonchamp[‡]

Abstract

Optimizing maintenance scheduling is a major issue to improve the performance of hydropower plants. We study a system of several physical components (turbines, alternators, generators) sharing a common stock of spare parts. Components experience random failures that occur according to known failure distributions. We seek a deterministic preventive maintenance strategy that minimizes an expected cost depending on maintenance and forced outages of the system. The Interaction Prediction Principle is used to decompose the original large-scale optimization problem into a sequence of independent subproblems of smaller dimension. Each subproblem consists in optimizing the maintenance on a single component. The resulting algorithm iteratively solves the subproblems with a blackbox algorithm and coordinates the components. The maintenance optimization problem is a mixed-integer problem. However, decomposition methods are based on variational techniques, therefore we have to relax the dynamics of the system and the cost functions. Relaxation parameters have an important influence on the optimization and must be appropriately chosen. We apply the decomposition method on a system with 80 components. It outperforms the reference blackbox algorithm applied directly on the original problem.

1 Introduction

In industry, maintenance aims at improving the availability of physical assets and therefore impacts the overall performance of a system. There exists two main kind of maintenance: corrective and preventive. Corrective maintenance (CM) is performed in reaction to a breakdown. Preventive maintenance (PM) consists in repairing or replacing a component before a failure. Maintenance policies have an important economic impact and are therefore studied in various areas such as the electricity sector [14], the manufacturing industry [13] or civil engineering [26]. In the electricity sector, maintenance optimization plays a major role in ensuring a reliable and competitive electricity production.

In this work, we consider components of hydroelectric power plants such as turbines, transformers or generators. We study a system of a given type of components that share a common stock of spare parts. The time horizon is 40 years. Over time, components experience random failures that occur according to known failure distributions. Thus, the dynamics of the system is stochastic. A preventive strategy consists in choosing the dates of replacement for each component of the system. The goal is to find a preventive strategy that minimizes an expected cost depending on maintenance and on the occurrences of forced outages of the system. Operational constraints impose to only look for deterministic maintenance strategies. This means that the dates of PM are chosen at the beginning of the time horizon with only a statistical knowledge of the future dates of failure: this is referred as an open-loop strategy. This differs from condition-based maintenance [21] where maintenance decisions

^{*}CERMICS, École des Ponts ParisTech, 6 et 8 avenue Blaise Pascal, 77455 Marne la Vallée Cedex 2, France

[†]Unité de Mathématiques Appliquées, ENSTA Paris, 828 Boulevard des Maréchaux, 91762 Palaiseau Cedex, France

[‡]EDF R&D PRISME, 6 quai Watier, 78400 Chatou, France

are taken given the degradation state of the components. Indeed, as the decisions depend on online observations, a condition-based maintenance strategy is stochastic. Our problem is modelled as a non-linear mixed integer program. The numerical experiments should involve systems constituted of up to 80 components in order to model the most demanding industrial case. This leads to optimization problems in high dimension that are numerically challenging.

We give a brief overview of the literature on the optimization methods for multi-component maintenance scheduling problems. As noted in [12], there is a great diversity in the modelling of maintenance problems because the objective and constraints vary from one study to another. Nevertheless, many reviews exist on optimal maintenance scheduling [1, 8, 14, 20] and give in particular a summary of the optimization techniques from the literature. They can be split in two main categories: mathematical programming approaches and heuristic methods [14].

Mixed integer programming is used in [15] for a periodic maintenance optimization problem. A linear relaxation of the problem is solved using column generation. In [17] a shutdown planning for the refueling of nuclear power plants is designed with a Benders decomposition coupled with various heuristics. A joint maintenance and spare part ordering optimization is studied in [22] using a Markov decision process and the value iteration algorithm. However, when the problem size grows, the resolution becomes intractable with these methods.

Many studies consider time-based [24, 25] or age-based [7, 23] maintenance policies. Such strategies are only defined with one decision variable per component: either the periodicity of maintenance (time-based) or the age at which a component is replaced (age-based). In this paper, more general strategies are considered as we can decide whether or not to perform a PM at each time step for each component. Suppose that there are T time steps and n components, then our maintenance strategy is defined by nT decision variables instead of n variables for time-based or age-based strategies. The effort is justified as we consider a system on a long-term horizon where the cost incurred by forced outages are of the order of millions of euros. Then, even a minor improvement in the maintenance strategy generates important savings.

As highlighted in [14], a frontal resolution is impracticable for high-dimensional problems and resorting to a decomposition method is relevant. Previous works use a linear relaxation to apply decomposition techniques such as Benders [17] or Dantzig-Wolfe decomposition [15]. The originality of our work lies in the use of a continuous relaxation of the system on which we apply a decomposition-coordination method based on variational techniques [6]. To our knowledge such a decomposition scheme has not been applied for optimal maintenance scheduling.

Originated from the work of [2, 16, 19, 27], decomposition-coordination methods consist in splitting the original large-scale optimization problem into several independent subproblems of smaller size that can be solved efficiently. The subproblems are coordinated to ensure that the concatenation of solutions leads to an optimal solution of the original problem. Different types of decomposition-coordination schemes have been designed, by prices, by quantities or by prediction. They have been unified within the Auxiliary Problem Principle [9].

In our setting, each subproblem consists in optimizing the maintenance on a single component. The decomposition algorithm iteratively solves the subproblems with the blackbox algorithm MADS [4] and coordinates the components in order to reach a global optimum. The relaxation parameters have an important influence on the output of the optimization by decomposition and must be appropriately chosen. We apply the decomposition method on relaxed systems with up to 80 components. The most demanding case takes around 20 hours of computation time. We show that in high dimension the decomposition method outperforms the blackbox algorithm applied directly on the original problem.

The paper is organized as follows: in Section 2, we describe the industrial system and formulate the maintenance optimization problem. The Auxiliary Problem Principle and the decomposition by prediction are introduced in Section 3. The application of the decomposition method to the

maintenance optimization problem and the relaxation of the system are presented in Section 4. Section 5 contains numerical results showing the efficiency of the method in high dimension. Finally, in Section 6, we conclude and give directions for future research.

2 System modelling and maintenance optimization problem

In this section, we describe the model of the studied industrial system and formulate the maintenance optimization problem.

2.1 Notations

In the following, for any vector $v = (v_1, \dots, v_n)$, we denote the first k components of v by:

$$v_{1:k} = (v_1, \dots, v_k). \quad (2.1)$$

The notation $\langle \cdot, \cdot \rangle$ represents the inner product in a Hilbert space and $\|\cdot\|$ is the induced norm. Random variables are denoted with capital bold letters. The indicator function is defined as follows.

Definition 2.1. Let $p \in \mathbb{N}$, $\mathcal{A} \subset \mathbb{R}^p$ and $x \in \mathbb{R}^p$. The indicator function of the set \mathcal{A} is

$$\mathbf{1}_{\mathcal{A}}(x) = \begin{cases} 1 & \text{if } x \in \mathcal{A}, \\ 0 & \text{if } x \notin \mathcal{A}. \end{cases} \quad (2.2)$$

Random variables are used in the description of the system. Hence, we introduce the following notation for the space of random variables.

Definition 2.2. Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space and $(\mathbb{Y}, \mathcal{Y})$ be a measurable space. The set of measurable functions from $(\Omega, \mathcal{F}, \mathbb{P})$ to $(\mathbb{Y}, \mathcal{Y})$ is denoted by $\mathcal{Y} = L^0(\Omega, \mathcal{F}, \mathbb{P}; \mathbb{Y})$, where we omit the σ -algebra \mathcal{Y} in the notation.

Any random variable $\mathbf{Y} : \Omega \rightarrow \mathbb{Y}$ is an element of \mathcal{Y} .

2.2 Description of the system

We consider a system of $n \in \mathbb{N}^*$ physical components of a hydropower plant (generators, turbines or transformers) sharing a common stock of spare parts. A sketch of the system with $n = 2$ components is represented in Figure 1. A corrective maintenance (CM) consists in the replacement of a component after a failure. A preventive maintenance (PM) is a planned replacement of a component before a failure.

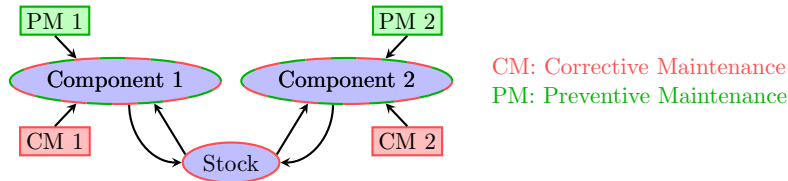


Figure 1: System of two components sharing the same stock of spare parts

We study this system on a horizon $T \in \mathbb{N}^*$. In the sequel, $i \in \mathbb{I} = \{1, \dots, n\}$ denotes a component index, $t \in \mathbb{T} = \{0, \dots, T\}$ denotes a time step and we use the notation $\mathbb{T}_{:-1}$ for the set $\{0, \dots, T-1\}$.

2.2.1 Characterization of the stock and the components

The stock over time is characterized by the sequence of random variables

$$\mathbf{S} = (\mathbf{S}_0, \dots, \mathbf{S}_T) \in \mathcal{S}, \quad (2.3)$$

defined on a given probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and where \mathbf{S}_t is the random variable representing the number of available spare parts at time t . We have $\mathcal{S} = L^0(\Omega, \mathcal{F}, \mathbb{P}; \mathbb{S})$ the set of all random variables taking values in $\mathbb{S} = \{0, \dots, s\}^{T+1}$. The parameter $s \in \mathbb{N}$ is the maximum number of spare parts. The value of the initial stock is set to $\mathbf{S}_0 = s$. The replenishment delay for the parts, that is, the time from order to delivery of a part, is known and denoted by $D \in \mathbb{N}$.

At time t , component i is characterized by random variables representing:

- its regime

$$\mathbf{E}_{i,t} = \begin{cases} 0 & \text{if the component is broken,} \\ 1 & \text{if the component is healthy.} \end{cases} \quad (2.4)$$

A component has only two regimes: in the healthy regime, it runs in its nominal operating point. In the broken regime, it stops working completely. Initially all components are healthy i.e. $\mathbf{E}_{i,0} = 1$ for all $i \in \mathbb{I}$.

- its age (if healthy) or the time for which it has failed (if broken) denoted by the real-valued random variable $\mathbf{A}_{i,t}$. Initially the components are new i.e. $\mathbf{A}_{i,0} = 0$ for all $i \in \mathbb{I}$.
- the time elapsed since its last D failures

$$\mathbf{P}_{i,t} = (\mathbf{P}_{i,t}^1, \dots, \mathbf{P}_{i,t}^D), \quad (2.5)$$

where D is the number of time steps for the supply of spare parts. For $d \in \{1, \dots, D\}$, $\mathbf{P}_{i,t}^d$ is the number of time steps elapsed since the d -th undiscarded failure of component i . $\mathbf{P}_{i,t}^d$ takes a default value δ if the component has failed fewer than d times. Hence, $\mathbf{P}_{i,t}^d$ takes values in $\{\delta\} \cup \mathbb{R}_+$ and $\mathbf{P}_{i,0} = (\delta, \dots, \delta)$. The random vector $\mathbf{P}_{i,t}$ is useful to compute the dates of replenishment of the stock. It is enough to store at most the dates of the last D failures to describe the supply of the stock. More details are given in §2.3.1.

The characteristics of component i at time t are gathered in

$$\mathbf{X}_{i,t} = (\mathbf{E}_{i,t}, \mathbf{A}_{i,t}, \mathbf{P}_{i,t}) \in \mathcal{X}_{i,t}, \quad (2.6)$$

where $\mathcal{X}_{i,t} = L^0(\Omega, \mathcal{F}, \mathbb{P}; \mathbb{X}_{i,t})$ with $\mathbb{X}_{i,t} = \{0, 1\} \times \mathbb{R}_+ \times (\{\delta\} \cup \mathbb{R}_+)^D$. The state of the system is then described at t by $(\mathbf{X}_{1,t}, \dots, \mathbf{X}_{n,t}, \mathbf{S}_t)$. Finally, to describe the components over the whole study period we introduce

$$\mathbf{X} = (\mathbf{X}_1, \dots, \mathbf{X}_n) = ((\mathbf{X}_{1,0}, \dots, \mathbf{X}_{1,T}), \dots, (\mathbf{X}_{n,0}, \dots, \mathbf{X}_{n,T})) \in \mathcal{X}, \quad (2.7)$$

where $\mathcal{X} = \prod_{i=1}^n \mathcal{X}_i$ and $\mathcal{X}_i = \prod_{t=0}^T \mathcal{X}_{i,t}$, for all $i \in \mathbb{I}$. In order to emphasize that \mathbf{X} depends on all the components of the system, we sometimes use the notation $\mathbf{X}_{1:n}$ instead of \mathbf{X} .

2.2.2 Preventive maintenance (PM) strategy

A PM consists in repairing a component although it is in the healthy regime. The dates of PM can be different for each component. They define the preventive maintenance strategy of the system. Operational constraints impose to look for deterministic strategies. This means that the dates of PM

are chosen without any knowledge on the state of the system after the beginning of the time horizon and cannot be changed during the study. The maintenance strategy is defined by a vector

$$u = (u_1, \dots, u_n) = ((u_{1,0}, \dots, u_{1,T}), \dots, (u_{n,0}, \dots, u_{n,T})) \in \mathbb{U} = [0, 1]^{n(T+1)}, \quad (2.8)$$

where $u_{i,t}$ characterizes the PM for component i at time t . More precisely, we set a threshold $0 < \nu < 1$: a control $u_{i,t} \geq \nu$ corresponds to a rejuvenation of the component proportional to $u_{i,t}$ and a value $u_{i,t} < \nu$ corresponds to not performing a maintenance. We consider that the maintenance operation is instantaneous and that it does not use parts from the stock. The reason is that PMs are planned in advance, hence it is possible to order the parts so that they arrive just on time for the maintenance operation. Note that the modelling of the PM strategy uses a continuous decision variable u , this choice is justified in §2.6.

2.2.3 Failures of the components

In our study, the failure distribution of component i is a known Weibull distribution of parameters (β_i, λ_i) denoted by $\text{Weib}(\beta_i, \lambda_i)$. The probability of failure of a component at a given time step only depends on its age and its failure distribution.

Proposition 2.3. *Let $i \in \mathbb{I}$. Assume that component i has age $a \geq 0$ at time t . Let F_i be the cumulative distribution function of the failure distribution of component i . Then, the probability of failure of component i at time $t + \Delta t$ conditionally to the component being healthy at t is given by*

$$p_i(a) = \frac{F_i(a + \Delta t) - F_i(a)}{1 - F_i(a)}, \quad (2.9)$$

The proof of this result can be found in [18, Section 2.2.1]. We introduce the random sequence

$$\mathbf{W} = (\mathbf{W}_1, \dots, \mathbf{W}_n) = ((\mathbf{W}_{1,1}, \dots, \mathbf{W}_{1,T}), \dots, (\mathbf{W}_{n,1}, \dots, \mathbf{W}_{n,T})) \in \mathcal{W} \quad (2.10)$$

where $\mathcal{W} = L^0(\Omega, \mathcal{F}, \mathbb{P}; [0, 1]^{nT})$. The random process \mathbf{W} is an exogenous noise that affects the dynamics of the regime \mathbf{E} and the age \mathbf{A} . We assume that all $\mathbf{W}_{i,t}$ are independent random variables and follow a uniform distribution on $[0, 1]$. At time step t , component i has age $\mathbf{A}_{i,t}$. If $\mathbf{W}_{i,t+1} < p_i(\mathbf{A}_{i,t})$, component i fails at $t + 1$, otherwise no failure occurs.

2.3 Dynamics of the system

2.3.1 Dynamics of a component

Let $i \in \mathbb{I}$ and $t \in \mathbb{T}_{-1}$. The dynamics of component i between t and $t + 1$ is described as follows.

1. If component i is healthy i.e. $\mathbf{E}_{i,t} = 1$:

- (a) If $u_{i,t} \geq \nu$, then a PM is performed. After a PM, component i stays healthy and is rejuvenated so that

$$(\mathbf{E}_{i,t+1}, \mathbf{A}_{i,t+1}) = (1, (1 - u_{i,t})(\mathbf{A}_{i,t} + 1)). \quad (2.11)$$

Note that $u_{i,t} = 1$ makes the component as good as new: in this case we have $\mathbf{A}_{i,t+1} = 0$.

- (b) If $u_{i,t} < \nu$, then no PM is performed. Component i fails with probability $p_i(\mathbf{A}_{i,t})$:

$$(\mathbf{E}_{i,t+1}, \mathbf{A}_{i,t+1}) = \begin{cases} (0, 0) & \text{if } \mathbf{W}_{i,t+1} < p_i(\mathbf{A}_{i,t}), \\ (1, \mathbf{A}_{i,t} + 1) & \text{otherwise.} \end{cases} \quad (2.12)$$

2. If component i is broken i.e. $\mathbf{E}_{i,t} = 0$:

(a) If a spare is available in the stock, a CM is performed to replace the component. We get:

$$(\mathbf{E}_{i,t+1}, \mathbf{A}_{i,t+1}) = (1, 0) . \quad (2.13)$$

(b) If no spare part is available, the defective component stays in the broken regime:

$$(\mathbf{E}_{i,t+1}, \mathbf{A}_{i,t+1}) = (0, \mathbf{A}_{i,t} + 1) . \quad (2.14)$$

The unit is shut down until the arrival of a spare part and the execution of the CM. Such a situation is a forced outage. During the shut down no electricity is produced.

We have to express formally that a spare part is available for the replacement of component i . At time t , suppose that the stock has $\mathbf{S}_t = r$ parts and that m components are broken. If $r \geq m$ then all components can be replaced immediately. When $r < m$, we must choose which components to replace. Our modelling choice is to replace the broken components following the order of their index: if $i_1 \leq \dots \leq i_r \leq \dots \leq i_m$ are the indices of the broken components, we replace only the components with index i_1, \dots, i_r , the others stay in the broken regime and wait for new available parts. Using this choice, the availability of a spare part for component i corresponds to the condition

$$\mathbf{S}_t \geq \sum_{j=1}^i \mathbf{1}_{\{0\}}(\mathbf{E}_{j,t}) . \quad (2.15)$$

The right hand side of (2.15) simply counts the number of broken components with index smaller or equal than i .

To completely describe the dynamics of a component, we have to specify the dynamics of the vector $\mathbf{P}_{i,t}$. It has been introduced in §2.2.1 to store the dates of failures of the component and compute the dates for the replenishment of the stock.

- If $\mathbf{P}_{i,t} = (t_1, \dots, t_d, \delta, \dots, \delta)$ with $t_1, \dots, t_d \geq 0$, meaning that component i has undergone $d < D$ failures so far, then

$$\mathbf{P}_{i,t+1} = \begin{cases} (t_1 + 1, \dots, t_d + 1, 0, \delta, \dots, \delta) & \text{if failure at } t + 1 , \\ (t_1 + 1, \dots, t_d + 1, \delta, \delta, \dots, \delta) & \text{otherwise .} \end{cases} \quad (2.16a)$$

$$(2.16b)$$

- If $\mathbf{P}_{i,t} = (t_1, \dots, t_D)$ with $t_1, \dots, t_D \geq 0$, meaning that component i has undergone at least D failures so far, then

$$\mathbf{P}_{i,t+1} = \begin{cases} (t_2 + 1, \dots, t_D + 1, 0) & \text{if failure at } t + 1 , \\ (t_1 + 1, \dots, t_D + 1) & \text{otherwise .} \end{cases} \quad (2.17a)$$

$$(2.17b)$$

In (2.17a), note that t_1 is discarded. As $\mathbf{P}_{i,t} = (t_1, \dots, t_D)$ and $t_1 > \dots > t_D \geq 0$, we get that $t_1 \geq D - 1$. At time step $t + 1$, the part ordered from the failure at t_1 has arrived. Then, storing t_1 is not useful anymore. So if a failure occurs at $t + 1$, we can discard t_1 to make room for the new date of failure. This proves that it is enough to have $\mathbf{P}_{i,t}$ of size D to compute the replenishment of the stock as stated in §2.2.1. Note that the dates are not discarded if there is no failure, so it is possible to have $t_d > D$ for some $d \in \{1, \dots, D\}$. Such variables have no influence on the dynamics of the system.

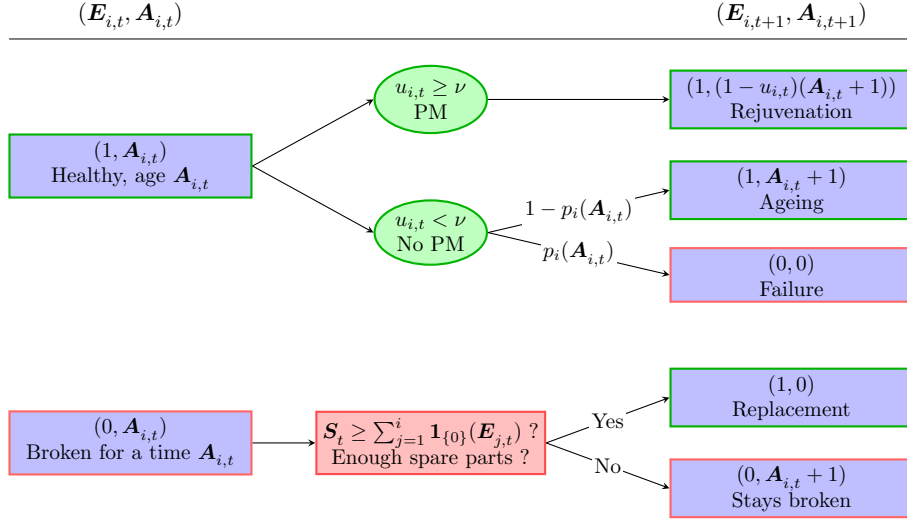


Figure 2: Dynamics of component i

Figure 2 describes the dynamics of component i from t to $t + 1$. Recall that we have $\mathbf{X}_{i,t} = (\mathbf{E}_{i,t}, \mathbf{A}_{i,t}, \mathbf{P}_{i,t})$. We write the *dynamics of component i* on the whole time horizon as:

$$\Theta_i(\mathbf{X}_{1:i}, \mathbf{S}, u_i, \mathbf{W}_i) = 0, \quad (2.18)$$

where $\Theta_i = \{\Theta_{i,t}\}_{t \in \mathbb{T}}$ such that:

$$\begin{cases} \Theta_{i,t+1}(\mathbf{X}_{1:i}, \mathbf{S}, u_i, \mathbf{W}_i) = \mathbf{X}_{i,t+1} - f_i(\mathbf{X}_{1:i,t}, \mathbf{S}_t, u_{i,t}, \mathbf{W}_{i,t+1}), & t \in \mathbb{T}_{:-1} \\ \Theta_{i,0}(\mathbf{X}_{1:i}, \mathbf{S}, u_i, \mathbf{W}_i) = \mathbf{X}_{i,0} - x_i, \end{cases} \quad (2.19)$$

with $x_i = (1, 0, \delta, \dots, \delta)^\top$ and f_i represents the dynamics we just described for component i . An explicit expression of f_i is given in Appendix A.

Note that there is a coupling between the dynamics of component i and the stock. There is also a coupling with components $j < i$. This is due to the choice (2.15) of replacing the broken components with the smallest indices first if there are not enough spare parts.

2.3.2 Dynamics of the stock

For the stock, the initial number of spare parts is $\mathbf{S}_0 = s$. As PMs can be anticipated, we consider that the needed spares are ordered so that they arrive just on time for the scheduled maintenance. Therefore, they do not appear in the dynamics of the stock. A part is used for each CM and a new part is ordered only after the failure of a component. The number of time steps for the supply of a part is D . Hence, the part ordered after the d -th undiscarded failure of component i arrives in the stock at $t + 1$ if $\mathbf{P}_{i,t+1}^d = D$. This is equivalent to $\mathbf{P}_{i,t}^d = D - 1$. On the other hand, the number of broken components is $\sum_{i=1}^n \mathbf{1}_{\{0\}}(\mathbf{E}_{i,t})$ and we replace as many of them as possible given the current level of stock \mathbf{S}_t . Thus, we have:

$$\mathbf{S}_{t+1} = \mathbf{S}_t + \sum_{i=1}^n \sum_{d=1}^D \mathbf{1}_{\{D-1\}}(\mathbf{P}_{i,t}^d) - \min \left\{ \mathbf{S}_t, \sum_{i=1}^n \mathbf{1}_{\{0\}}(\mathbf{E}_{i,t}) \right\}, \quad t \in \mathbb{T}_{:-1}. \quad (2.20)$$

We write the *dynamics of the stock* in compact form as:

$$\Theta_{\mathbf{S}}(\mathbf{X}_{1:n}, \mathbf{S}) = 0, \quad (2.21)$$

where $\Theta_{\mathcal{S}} = \{\Theta_{\mathcal{S},t}\}_{t \in \mathbb{T}}$ such that:

$$\begin{cases} \Theta_{\mathcal{S},t+1}(\mathbf{X}_{1:n}, \mathcal{S}) = \mathcal{S}_{t+1} - f_{\mathcal{S}}(\mathbf{X}_{1:n,t}, \mathcal{S}_t), & t \in \mathbb{T}_{:-1} \\ \Theta_{\mathcal{S},0}(\mathbf{X}_{1:n}, \mathcal{S}) = \mathcal{S}_0 - s, \end{cases}$$

with $f_{\mathcal{S}}$ corresponding to the right-hand side of (2.20). Note that \mathcal{S}_{t+1} depends on on the current level of stock \mathcal{S}_t but also on $\mathbf{X}_{i,t}$ for all $i \in \mathbb{I}$. The stock is coupling all the components of the system.

Finally, the *dynamics of the whole system* is summarized by the almost sure equality constraint $\Theta(\mathbf{X}, \mathcal{S}, u, \mathbf{W}) = 0$, where we have $\Theta : \mathcal{X} \times \mathcal{S} \times \mathbb{U} \times \mathcal{W} \rightarrow \mathcal{L}$, with $\Theta = \{\{\Theta_i\}_{i \in \mathbb{I}}, \Theta_{\mathcal{S}}\}$ and $\mathcal{L} = (\prod_{i=1}^n \mathcal{L}_i) \times \mathcal{L}_{\mathcal{S}}$ where $\mathcal{L}_i = L^0(\Omega, \mathcal{F}, \mathbb{P}; \mathbb{R}^{(D+2)(T+1)})$ for $i \in \mathbb{I}$ and $\mathcal{L}_{\mathcal{S}} = L^0(\Omega, \mathcal{F}, \mathbb{P}; \mathbb{R}^{(T+1)})$.

We have now completely described the dynamics of the system. In the next part we specify the costs associated to the system.

2.4 Costs generated by the system

The costs generated by the system are due to PMs, CMs and forced outages of the unit. In practice as PMs are scheduled in advance, they are cheaper than unpredictable CMs. A forced outage of the unit induces a loss of production. It is characterized by a yearly cost which is higher than that of a PM or a CM. We consider a discount rate τ meaning that a cost c occurring at time t will be valued $\eta_t c$ with the discount factor $\eta_t := \frac{1}{(1+\tau)^t}$. We introduce the following notations:

- $j_{i,t}^P(u_{i,t})$ is the PM cost incurred at time t for component i . Let C_i^P be the cost of a PM operation on component i . We set

$$j_{i,t}^P(u_{i,t}) = \eta_t C_i^P u_{i,t}^2. \quad (2.22)$$

We use a quadratic cost as it is strongly convex and should favor numerical convergence. In particular, in the case when $0 < u_{i,t} < \nu$, which models a situation where no PM is performed, we have $j_{i,t}^P(u_{i,t}) > 0$.

- $j_{i,t}^C(\mathbf{X}_{i,t})$ is the CM cost. It is due at the time of the failure of a component, even if there is no spare part to perform the operation immediately. Hence it only occurs when $(\mathbf{E}_{i,t}, \mathbf{A}_{i,t}) = (0, 0)$. Let C_i^C be the cost of a CM operation on component i . We have

$$j_{i,t}^C(\mathbf{X}_{i,t}) = \eta_t C_i^C \mathbf{1}_{\{0\}}(\mathbf{E}_{i,t}) \mathbf{1}_{\{0\}}(\mathbf{A}_{i,t}). \quad (2.23)$$

- $j_t^F(\mathbf{X}_{1:n,t})$ is the forced outage cost. The unit is in forced outage when at least one component is in a failed state and the CM has not occurred immediately because of a lack of spare part. Let C^F be the forced outage cost per time unit. We have

$$j_t^F(\mathbf{X}_{1:n,t}) = \eta_t C^F \min \left\{ 1, \sum_{i=1}^n \mathbf{1}_{\{0\}}(\mathbf{E}_{i,t}) \mathbf{1}_{\mathbb{R}_+^*}(\mathbf{A}_{i,t}) \right\}. \quad (2.24)$$

In order to consider the previous costs over the whole study period we introduce:

- the *total maintenance cost* (preventive and corrective) generated by component $i \in \mathbb{I}$ on the studied period:

$$j_i(\mathbf{X}_i, u_i) = \sum_{t=0}^T j_{i,t}^P(u_{i,t}) + j_{i,t}^C(\mathbf{X}_{i,t}), \quad (2.25)$$

- the *total forced outage cost* generated by the system during the studied period:

$$j^F(\mathbf{X}_{1:n}) = \sum_{t=0}^T j_t^F(\mathbf{X}_{1:n,t}), \quad (2.26)$$

2.5 Formulation of the maintenance optimization problem

The dynamics of the system is stochastic as it depends on the failure of the components, modelled by the random vector \mathbf{W} . The cost function is then stochastic as well. The objective is to find the deterministic maintenance strategy $u \in \mathbb{U}$ that minimizes an expected cost over all failure scenarios:

$$\begin{aligned} \min_{(\mathbf{X}, \mathbf{S}, u) \in \mathcal{X} \times \mathcal{S} \times \mathbb{U}} \mathbb{E} \left(\sum_{i=1}^n j_i(\mathbf{X}_i, u_i) + j^F(\mathbf{X}_{1:n}) \right) \\ \text{s.t. } \Theta(\mathbf{X}, \mathbf{S}, u, \mathbf{W}) = 0. \end{aligned} \quad (2.27)$$

Note that the term $\mathbb{E}(\sum_{i=1}^n j_i(\mathbf{X}_i, u_i)) = \sum_{i=1}^n \mathbb{E}(j_i(\mathbf{X}_i, u_i))$ is additive with respect to the components whereas $\mathbb{E}(j^F(\mathbf{X}_{1:n}))$ induces a non-additive coupling between the components. In the theoretical part on the Auxiliary Problem Principle (Section 3), we will see that these two terms are treated in a different way for the design of a decomposition-coordination algorithm.

2.6 Optimization with Mesh Adaptive Direct Search (MADS) and its limits

This work is motivated by a real industrial case of maintenance of components of hydroelectric power plants. Problem (2.27) models a simplified version of the real problem. The number of components n in the system can be up to 80 and T is 40 years so \mathbb{U} has dimension up to 3280 as we have one maintenance decision each year for each component (starting at year 0).

When the number of components is not too large, say $n < 10$, the maintenance problem (2.27) is solved efficiently by MADS [4]. MADS has been designed for continuous optimization. It uses the modelling of §2.2 and the cost functions described in §2.4. In particular, the PM strategies are modelled with a continuous decision variable.

MADS is a blackbox algorithm. This means that evaluation points are chosen iteratively without the need for the gradients of the objective function. This feature is particularly appealing as the cost function is not differentiable.

In practice, the objective function is costly to evaluate as the expectation is estimated using Monte-Carlo simulations. When the number of components is large ($n \geq 10$), MADS needs more iterations to explore the high-dimensional space of solutions and the objective function takes more time to evaluate. The algorithm may not be able to find a very effective maintenance strategy. To overcome the difficulty of MADS when dealing with large systems, we use a decomposition of the original optimization problem component by component. In the decomposition method, we use the same modelling for the system as in MADS. In particular, we use the continuous modelling of PM strategies given in §2.2.2.

3 The Auxiliary Problem Principle for decomposition

When performing optimization on a large scale system, that is, a system which is described by a large number of variables or constraints, the computation is often highly expensive either in time or in memory requirement (or both). The idea of decomposition is to formulate subproblems involving only smaller subsystems of the original large system. Each subproblem is easier to solve than the global optimization problem and provides its “local” solution. Then the goal of coordination is to ensure that gathering the local solutions leads to a global solution. Decomposition-coordination methods usually result in an iterative process alternating an optimization step on the subsystems and a coordination step that updates the subproblems.

The main advantage of decomposition is that the resolution of the small subproblems is faster than the original problem. More than that, the computational complexity of an optimization problem is often superlinear or even exponential in the size of the problem. Hence, the sum of the computational

efforts required for the resolution of all subproblems will be lower than for the global problem, even if the resolution of the subproblems must be carried out multiple times. Another feature of decomposition methods is that they are naturally adapted to parallelization as each subproblem is independent. This leads to a reduction of computation time.

In this section, we introduce the general framework of the Auxiliary Problem Principle (APP) that allows us to formulate a decomposition-coordination scheme for the optimization problem (2.27). The APP has first been introduced in [9] as a unified framework for decomposition methods but also for other classical iterative algorithms. This principle casts the resolution of an optimization problem into the resolution of a sequence of auxiliary problems whose solutions converge to the solution of the original problem. Appropriate choices for the auxiliary problems lead to decomposition-coordination schemes.

Based on [6], we present the main ideas of the APP. Consider the following problem, which we call the master problem:

$$\min_{u \in \mathbb{U}^{ad}} J_{\Sigma}(u) + J_{\Delta}(u) \quad \text{such that } \Theta(u) \in -C, \quad (3.1)$$

where

- \mathbb{U}^{ad} is a non-empty, closed, convex subset of a Hilbert space $\mathbb{U} = \mathbb{U}_1 \times \dots \times \mathbb{U}_N$ and is decomposable as $\mathbb{U}^{ad} = \mathbb{U}_1^{ad} \times \dots \times \mathbb{U}_N^{ad}$ where for all $i \in \mathbb{I}$, $\mathbb{U}_i^{ad} \subset \mathbb{U}_i$ is a closed convex set;
- C is a pointed closed convex cone of a Hilbert space $\mathcal{C} = \mathcal{C}_1 \times \dots \times \mathcal{C}_N$ and is decomposable as $C = C_1 \times \dots \times C_N$ where for all $i \in \mathbb{I}$, $C_i \subset \mathcal{C}_i$ is a closed convex cone;
- $J_{\Sigma} : \mathbb{U} \rightarrow \mathbb{R}$ and $J_{\Delta} : \mathbb{U} \rightarrow \mathbb{R}$ are convex and lower semi-continuous (l.s.c.). The function $J_{\Sigma} + J_{\Delta}$ is coercive on \mathbb{U}^{ad} . Moreover, J_{Δ} is differentiable and J_{Σ} is additive with respect to the decomposition of the admissible space, so that we have for $u = (u_1, \dots, u_N) \in \mathbb{U}^{ad}$ with $u_i \in \mathbb{U}_i^{ad}$ for all $i \in \mathbb{I}$:

$$J_{\Sigma}(u) = \sum_{i=1}^N J_{\Sigma,i}(u_i); \quad (3.2)$$

where $J_{\Sigma,i} : \mathbb{U}_i \rightarrow \mathbb{R}$.

- $\Theta : \mathbb{U} \rightarrow \mathcal{C}$ is differentiable and C -convex, meaning that:

$$\forall u, v \in \mathbb{U}, \forall \rho \in [0, 1], \rho\Theta(u) + (1 - \rho)\Theta(v) - \Theta(\rho u + (1 - \rho)v) \in C. \quad (3.3)$$

We write $\Theta(u) = (\Theta_1(u), \dots, \Theta_N(u))$ with $\Theta_i : \mathbb{U} \rightarrow \mathcal{C}_i$ for $i \in \mathbb{I}$.

The goal of the APP is to turn the resolution of the master problem on \mathbb{U}^{ad} into the resolution of subproblems on the sets \mathbb{U}_i^{ad} . In the master problem, there is a non-additive coupling in the cost due to J_{Δ} and a coupling in the constraint Θ .

We use the Interaction Prediction Principle [19], which can be seen as a special case of the APP, to construct a decomposable auxiliary problem:

1. The decomposition of the admissible space $\mathbb{U}^{ad} = \mathbb{U}_1^{ad} \times \dots \times \mathbb{U}_N^{ad}$ defines the subspace on which each subproblem is solved.
2. The decomposition of the cone $C = C_1 \times \dots \times C_N$ specifies which part of the constraint is assigned to each subproblem.

Definition 3.1. Let $\bar{u} = (\bar{u}_1, \dots, \bar{u}_N) \in \mathbb{U}$ and $\bar{p} = (\bar{p}_1, \dots, \bar{p}_N) \in C^*$ the dual cone of C . We recall that

$$C^* = \{p \in \mathcal{C}^*, \langle p, q \rangle \geq 0 \text{ for all } q \in \mathcal{C}\}, \quad (3.4)$$

with \mathcal{C}^* being the dual space of \mathcal{C} . Let K be an auxiliary cost and Φ be an auxiliary constraint satisfying the following properties:

- $K : \mathbb{U} \rightarrow \mathbb{R}$ is convex, l.s.c., differentiable and additive: $K(u) = \sum_{i=1}^N K_i(u_i)$;
- $\Phi : \mathbb{U} \rightarrow \mathcal{C}$ is differentiable and block-diagonal: $\Phi(u) = (\Phi_1(u_1), \dots, \Phi_N(u_N))$.

The auxiliary problem for the master problem (3.1) arising from the choice of K and Φ is given by:

$$\begin{aligned} \min_{u \in \mathbb{U}^{ad}} K(u) + J_\Sigma(u) + \langle \nabla J_\Delta(\bar{u}) - \nabla K(\bar{u}), u \rangle + \langle \bar{p}, (\Theta'(\bar{u}) - \Phi'(\bar{u})) \cdot u \rangle \\ \text{s.t. } \Phi(u) - \Phi(\bar{u}) + \Theta(\bar{u}) \in -C. \end{aligned} \quad (3.5)$$

The major interest in choosing K additive and Φ block-diagonal is that problem (3.5) decomposes in N independent subproblems which can be solved in parallel. For $i \in \mathbb{I}$, the i -th subproblem is given by

$$\begin{aligned} \min_{u_i \in \mathbb{U}_i^{ad}} K_i(u_i) + J_{\Sigma,i}(u_i) + \langle \nabla_{u_i} J_\Delta(\bar{u}) - \nabla K_i(\bar{u}_i), u_i \rangle \\ - \langle \bar{p}_i, \Phi'_i(\bar{u}_i) \cdot u_i \rangle + \sum_{j=1}^N \langle \bar{p}_j, \partial_{u_i} \Theta_j(\bar{u}) \cdot u_i \rangle \\ \text{s.t. } \Phi_i(u_i) - \Phi_i(\bar{u}_i) + \Theta_i(\bar{u}) \in -C_i. \end{aligned} \quad (3.6)$$

This subproblem only depends on $u_i \in \mathbb{U}_i^{ad}$ and inherits only the i -th component of the constraint.

Example 3.2. A canonical choice for the additive auxiliary cost function K is

$$K(u) = \sum_{i=1}^N K_i(u_i) \quad \text{with} \quad K_i(u_i) = J_\Delta(\bar{u}_{1:i-1}, u_i, \bar{u}_{i+1:n}). \quad (3.7)$$

Similarly a canonical choice for the block-diagonal auxiliary constraint Φ is

$$\Phi(u) = (\Phi_1(u_1), \dots, \Phi_N(u_N)) \quad \text{with} \quad \Phi_i(u_i) = \Theta_i(\bar{u}_{1:i-1}, u_i, \bar{u}_{i+1:n}). \quad (3.8)$$

The general idea is to construct the i -th term of the auxiliary function from the original function where only the i -th component is allowed to vary. \triangle

The following statement is the fundamental lemma for the theory of the APP.

Lemma 3.3. *Let u^\sharp be a solution of the auxiliary problem (3.5) and p^\sharp be an optimal multiplier for its constraint. If $(u^\sharp, p^\sharp) = (\bar{u}, \bar{p})$, then u^\sharp is a solution of the master problem (3.1) and p^\sharp is an optimal multiplier for its constraint.*

The proof consists in checking that if $(u^\sharp, p^\sharp) = (\bar{u}, \bar{p})$ then it solves the variational inequalities that are satisfied by an optimal solution and an optimal multiplier of the master problem (3.1). More details can be found in [6, Section 4]. Lemma 3.3 suggests to use the APP fixed-point Algorithm 1.

Algorithm 1 APP fixed-point algorithm

- 1: Start with $\bar{u} = u^0$, $\bar{p} = p^0$ and set $k = 0$.
 - 2: At iteration $k + 1$, solve the optimization problem (3.5) with $\bar{u} = u^k$ and $\bar{p} = p^k$. Let u^{k+1} be a solution and p^{k+1} an optimal multiplier for its constraint.
 - 3: If $\|u^{k+1} - u^k\| + \|p^{k+1} - p^k\|$ is “sufficiently small” then stop, else $k \leftarrow k + 1$ and go back to step 2.
-

A convergence result for Algorithm 1, given in [10, Theorem 5.1], is recalled now.

Theorem 3.4. [10, Theorem 5.1 with $\epsilon = 1$] Assume that

- the admissible space \mathbb{U}^{ad} is equal to the whole space \mathbb{U} ;
- the constraints are equality constraints, that is $C = \{0\}$;
- $K(u) = \frac{1}{2} \langle u, Ku \rangle$, $J_\Sigma(u) = 0$, $J_\Delta(u) = \frac{1}{2} \langle u, Ju \rangle + \langle j, u \rangle$ where K and J are linear self-adjoint strongly monotone and Lipschitz continuous operators, j is a vector in \mathbb{U} , and $2K - J$ is assumed to be strongly monotone;
- $\Phi(u) = Ou$, $\Theta(u) = Tu + t$ where the operators O and T are linear and surjective and t is a vector in \mathbb{C} ;
- (Geometric condition) the operator $TJ^{-1}O^\top + OJ^\top T^\top - 2TJ^{-1}KJ^{-1}T^\top$ is strongly monotone.

Then the sequence $\{(u^k, p^k)\}_{k \geq 0}$ generated by the APP fixed-point algorithm converges strongly to the unique optimal solution (u^\sharp, p^\sharp) of the original problem (3.1).

In this section, we have studied a master optimization problem with coupling coming both from the part J_Δ of the cost function and the constraint Θ . The APP allows to turn the resolution of the master problem into the iterative resolution of an auxiliary problem that is decomposable into independent subproblems of smaller size. In the next section, we present the application of the APP to solve the maintenance optimization problem described in Section 2.

4 Application of the APP for the maintenance problem (2.27)

In order to apply the APP to solve Problem (2.27), also referred as original problem in the sequel, we decompose the admissible space and then construct an auxiliary problem adapted to the proposed decomposition.

4.1 Decomposition of the space by component

Considering the physical nature of the industrial system composed of n components and a stock, we choose to decompose the problem in $n+1$ subproblems and call this decomposition a *decomposition by component*. More precisely, for $i \in \mathbb{I}$, the i -th subproblem is called *subproblem on component i* since it is solved on $\mathcal{X}_i \times \mathbb{U}_i$ and only involves the dynamics of component i . The $(n+1)$ -th subproblem is called *subproblem on the stock* since it is solved on \mathcal{S} and only involves the dynamics of the stock. This means that the space, $\mathcal{X} \times \mathcal{S} \times \mathbb{U}$, of Problem (2.27) is decomposed as a product of a $n+1$ subspaces:

$$\mathcal{X} \times \mathcal{S} \times \mathbb{U} = (\mathcal{X}_1 \times \mathbb{U}_1) \times \dots \times (\mathcal{X}_n \times \mathbb{U}_n) \times \mathcal{S}, \quad (4.1)$$

where, for $(\mathbf{X}, \mathbf{S}, u) = ((\mathbf{X}_1, \dots, \mathbf{X}_n), \mathbf{S}, (u_1, \dots, u_n)) \in \mathcal{X} \times \mathcal{S} \times \mathbb{U}$, we have

$$(\mathbf{X}_i, u_i) \in \mathcal{X}_i \times \mathbb{U}_i \text{ for all } i \in \mathbb{I} \text{ and } \mathbf{S} \in \mathcal{S}. \quad (4.2)$$

The constraints in Problem (2.27), that is $\Theta(\mathbf{X}, \mathbf{S}, u, \mathbf{W}) \in -C$ with $C = \{0\}_{\mathcal{L}}$ is decomposed through the following cone decomposition:

$$C = \{0\}_{\mathcal{L}} = \{0\}_{\mathcal{L}_1} \times \dots \times \{0\}_{\mathcal{L}_n} \times \{0\}_{\mathcal{L}_S} = C_1 \times \dots \times C_n \times C_S, \quad (4.3)$$

where for $i \in \mathbb{I}$, $\{0\}_{\mathcal{L}_i}$, $\{0\}_{\mathcal{L}_S}$ and $\{0\}_{\mathcal{L}}$ denote the null function of \mathcal{L}_i , \mathcal{L}_S and \mathcal{L} respectively. We recall that Θ_i takes values in \mathcal{L}_i and Θ_S takes values in \mathcal{L}_S .

4.2 Construction of an auxiliary problem

Problem (2.27) is not directly decomposable by component because of couplings, highlighted in Section 2, that we recall now.

- The maintenance cost $j^M = \sum_{i=1}^n j_i$ is additive with respect to the decomposition by component. The expectation of j^M can be identified with J_Σ in Section 3.
- The forced outage cost j^F induces a non-additive coupling between the components. The expectation of j^F can be identified with J_Δ in Section 3.
- The dynamics Θ_i of component i induces a coupling with the stock and all components with index $j < i$. The stock dynamics, Θ_S , is coupling the stock with all components.

In order to obtain a decomposition of Problem (2.27) by component, we use the canonical technique from Example 3.2. We define an additive mapping K and a block diagonal mapping Φ so that the resulting auxiliary problem is decomposable. We also choose to augment the auxiliary cost K with a strongly convex term in order to ease the numerical convergence of the method. Let $\bar{\mathbf{X}} = (\bar{\mathbf{X}}_1, \dots, \bar{\mathbf{X}}_n) \in \mathcal{X}$, $\bar{\mathbf{S}} \in \mathcal{S}$, $\bar{\boldsymbol{\Lambda}} \in C^*$ and $\gamma_x, \gamma_s, \gamma_u > 0$, we consider:

- an auxiliary cost function $K : \mathcal{X} \times \mathcal{S} \times \mathbb{U} \rightarrow \mathbb{R}$:

$$K(\mathbf{X}, \mathbf{S}, u) = \sum_{i=1}^n K_i(\mathbf{X}_i, u_i) + K_S(\mathbf{S}), \quad (4.4)$$

with

$$\begin{aligned} K_i(\mathbf{X}_i, u_i) &= \mathbb{E} \left(j^F(\bar{\mathbf{X}}_{1:i-1}, \mathbf{X}_i, \bar{\mathbf{X}}_{i+1:n}) + \frac{\gamma_x}{2} \|\mathbf{X}_i\|^2 + \frac{\gamma_u}{2} \|u_i\|^2 \right), \quad i \in \mathbb{I} \\ K_S(\mathbf{S}) &= \mathbb{E} \left(\frac{\gamma_s}{2} \|\mathbf{S}\|^2 \right), \end{aligned} \quad (4.5)$$

- an auxiliary dynamics mapping $\Phi : \mathcal{X} \times \mathcal{S} \times \mathbb{U} \times \mathcal{W} \rightarrow \mathcal{L}$ defined by

$$\Phi(\mathbf{X}, \mathbf{S}, u, \mathbf{W}) = (\Phi_1(\mathbf{X}_1, u_1, \mathbf{W}_1), \dots, \Phi_n(\mathbf{X}_n, u_n, \mathbf{W}_n), \Phi_S(\mathbf{S})), \quad (4.6)$$

with

$$\begin{aligned} \Phi_i(\mathbf{X}_i, u_i, \mathbf{W}_i) &= \Theta_i(\bar{\mathbf{X}}_{1:i-1}, \mathbf{X}_i, \bar{\mathbf{S}}, u_i, \mathbf{W}_i), \quad i \in \mathbb{I} \\ \Phi_S(\mathbf{S}) &= \Theta_S(\bar{\mathbf{X}}_{1:n}, \mathbf{S}), \end{aligned} \quad (4.7)$$

with $\bar{\mathbf{X}}_{1:0}$ being by convention an empty vector.

We can now write an auxiliary problem for (2.27). Assume that the original dynamics Θ is differentiable. Then Φ is differentiable and the auxiliary problem writes:

$$\begin{aligned} \min_{(\mathbf{X}, \mathbf{S}, u) \in \mathcal{X} \times \mathcal{S} \times \mathbb{U}} \mathbb{E} & \left(\sum_{i=1}^n \left(j_i(\mathbf{X}_i, u_i) + j^F(\bar{\mathbf{X}}_{1:i-1}, \mathbf{X}_i, \bar{\mathbf{X}}_{i+1:n}) \right) \right. \\ & \left. + \frac{\gamma_x}{2} \|\mathbf{X} - \bar{\mathbf{X}}\|^2 + \frac{\gamma_s}{2} \|\mathbf{S} - \bar{\mathbf{S}}\|^2 + \frac{\gamma_u}{2} \|u - \bar{u}\|^2 \right. \\ & \left. + \langle \bar{\boldsymbol{\Lambda}}, (\Theta'(\bar{\mathbf{X}}, \bar{\mathbf{S}}, \bar{u}, \mathbf{W}) - \Phi'(\bar{\mathbf{X}}, \bar{\mathbf{S}}, \bar{u}, \mathbf{W})) \cdot (\mathbf{X}, \mathbf{S}, u) \rangle \right) \\ \text{s.t. } & \Phi(\mathbf{X}, \mathbf{S}, u, \mathbf{W}) = 0. \end{aligned} \quad (4.8)$$

By construction the auxiliary problem (4.8) is decomposable with respect to decompositions (4.1) and (4.3).

4.3 Relaxation of the system

In the original problem (2.27), Θ takes values in the space of measurable functions $\mathcal{L} = L^0(\Omega, \mathcal{F}, \mathbb{P}; \mathbb{R}^q)$ with $q = (n(D+2) + 1)(T+1)$. This space is not well-suited to ensure the convergence of the APP fixed-point algorithm to an optimal solution of the original problem. As Theorem 3.4 applies in a Hilbert space, we can restrict Θ to take values in $L^2(\Omega, \mathcal{F}, \mathbb{P}; \mathbb{R}^q)$. This remark is made for theoretical purpose and does not influence the implementation of the APP fixed-point algorithm.

However, even with the restriction to the space $L^2(\Omega, \mathcal{F}, \mathbb{P}; \mathbb{R}^q)$, the assumptions of Theorem 3.4 are far from being satisfied in our industrial case. Indeed, the forced outage cost j^F is not quadratic, nor even convex, the dynamics Θ is not linear, and it is hard to check if the geometric condition is satisfied.

Moreover, the APP relies on variational techniques and requires the mappings Θ and Φ to be differentiable as the gradients Θ' and Φ' appear in Problem (4.8). However, the dynamics Θ in Problem (2.27) involves integer variables so the gradient Θ' is not defined. We propose a continuous relaxation of the system with relaxed cost and dynamics that are differentiable almost everywhere. It is possible to use a differentiable relaxation of the system but this requires more implementation efforts. As we are far from the conditions of convergence of the algorithm, nothing ensures that a differentiable relaxation would give better results than the simple one that is defined below.

4.3.1 State variable relaxation

Let $i \in \mathbb{I}$ and $t \in \mathbb{T}$. Recall from §2.2 that

$$\begin{cases} \mathbf{X}_{i,t} = (\mathbf{E}_{i,t}, \mathbf{A}_{i,t}, \mathbf{P}_{i,t}) \text{ takes values in } \{0, 1\} \times \mathbb{R} \times (\{\delta\} \cup \mathbb{R}_+)^D, \\ \mathbf{S}_t \text{ takes values in } \mathbb{N}. \end{cases} \quad (4.9)$$

We relax the integrity constraint on $\mathbf{E}_{i,t}$ and \mathbf{S}_t , so we allow the random variables

$$\begin{cases} \mathbf{X}_{i,t} = (\mathbf{E}_{i,t}, \mathbf{A}_{i,t}, \mathbf{P}_{i,t}) \text{ to take values in } [0, 1] \times \mathbb{R} \times (\{\delta\} \cup \mathbb{R}_+)^D, \\ \mathbf{S}_t \text{ to take values in } \mathbb{R}. \end{cases} \quad (4.10)$$

Remark 4.1. We lose the physical interpretation of the relaxed variables.

- If $0 < \mathbf{E}_{i,t} < 1$, we could think that component i is in a degraded regime where the closer $\mathbf{E}_{i,t}$ is to 1 the healthier it is. This interpretation is however not exact as the health of a component is only characterized by its age $\mathbf{A}_{i,t}$. The probability of failure of a component is indeed only a function of $\mathbf{A}_{i,t}$.
- A value $\mathbf{S}_t \in \mathbb{R}$ means that there can be a non-integer number of parts in the stock.

◇

4.3.2 Relaxation of the dynamics

The dynamics of the original system has been described in §2.3 and an explicit expression is given in Appendix A. This expression involves indicator functions $\mathbf{1}_{\mathcal{A}}$ for some set \mathcal{A} . The dynamics is then non-continuous. Replacing the original indicator function $\mathbf{1}_{\mathcal{A}}$ with a continuous relaxed version allows to define a relaxed dynamics for the system.

Definition 4.2. Let $\mathcal{A} \subset \mathbb{R}^p$, $x \in \mathbb{R}^p$ and $\alpha > 0$. We define a continuous relaxation $\mathbf{1}_{\mathcal{A}}^\alpha$ with parameter α of the indicator function $\mathbf{1}_{\mathcal{A}}$ as:

$$\mathbf{1}_{\mathcal{A}}^\alpha(x) = \begin{cases} 1 - 2\alpha d(\mathcal{A}, x) & \text{if } d(\mathcal{A}, x) \leq \frac{1}{2\alpha}, \\ 0 & \text{if } d(\mathcal{A}, x) > \frac{1}{2\alpha}, \end{cases} \quad (4.11)$$

where $d(\mathcal{A}, x)$ is the euclidean distance between x and the set \mathcal{A} .

Figure 3 illustrates the relaxation of the indicator function.

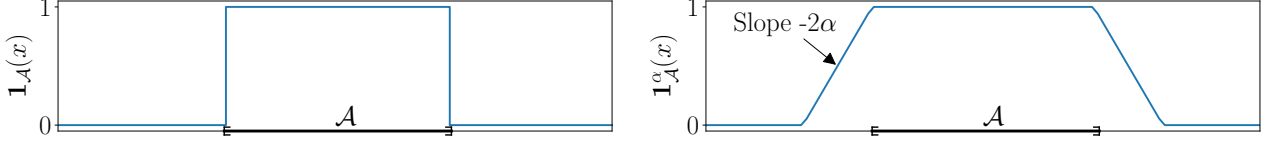


Figure 3: Illustration of the relaxation of the indicator function

The parameter α quantifies the stiffness of the relaxation. As $\alpha \rightarrow +\infty$, the relaxed indicator $\mathbf{1}_{\mathcal{A}}^{\alpha}$ converges towards the original indicator $\mathbf{1}_{\mathcal{A}}$. As $\alpha \rightarrow 0$, the relaxed indicator $\mathbf{1}_{\mathcal{A}}^{\alpha}$ converges pointwise towards the constant function 1. Note that for all $\alpha > 0$ the relaxed indicator $\mathbf{1}_{\mathcal{A}}^{\alpha}$ is continuous and differentiable almost everywhere.

A continuous relaxation of the system dynamics of parameter α is obtained by replacing the occurrences of the indicator function by its relaxed version. For instance, the relaxed dynamics of the stock is obtained from (2.20):

$$\mathbf{S}_{t+1} = \mathbf{S}_t + \sum_{i=1}^n \sum_{d=1}^D \mathbf{1}_{\{D-1\}}^{\alpha}(\mathbf{P}_{i,t}^d) - \min\left(\mathbf{S}_t, \sum_{i=1}^n \mathbf{1}_{\{0\}}^{\alpha}(\mathbf{E}_{i,t})\right). \quad (4.12)$$

Additional technical details and an explicit expression for the relaxed dynamics of the components are given in Appendix B. The relaxed dynamics and relaxed auxiliary dynamics are denoted in the sequel by Θ^{α} and Φ^{α} respectively.

4.3.3 Cost relaxation

The relaxation of the maintenance cost and forced outage cost is constructed using the same technique as for the dynamics. Let $i \in \mathbb{I}$, $t \in \mathbb{T}$ and $\alpha > 0$ be given,

- the relaxed maintenance cost for component i at time t with parameter α is defined as

$$j_{i,t}^{\alpha}(\mathbf{X}_{i,t}, u_{i,t}) = \eta_t C_i^P u_{i,t}^2 + \eta_t C_i^C \mathbf{1}_{\{0\}}^{\alpha}(\mathbf{E}_{i,t}) \mathbf{1}_{\{0\}}^{\alpha}(\mathbf{A}_{i,t}), \quad (4.13)$$

- the relaxed forced outage cost at time t with parameter α is defined as

$$j_t^{F,\alpha}(\mathbf{X}_{1:n,t}) = \eta_t C^F \min\left\{1, \sum_{i=1}^n \mathbf{1}_{\{0\}}^{\alpha}(\mathbf{E}_{i,t}) \mathbf{1}_{\mathbb{R}_+^*}^{\alpha}(\mathbf{A}_{i,t})\right\}. \quad (4.14)$$

Similarly, as in (2.25) and (2.26), we set

$$j_i^{\alpha}(\mathbf{X}_i, u_i) = \sum_{t=0}^T j_{i,t}^{\alpha}(\mathbf{X}_{i,t}, u_{i,t}) \quad \text{and} \quad j^{F,\alpha}(\mathbf{X}_{1:n}) = \sum_{t=0}^T j_t^{F,\alpha}(\mathbf{X}_{1:n,t}). \quad (4.15)$$

Using the relaxed dynamics and relaxed cost functions in the auxiliary problem (4.8), we get that the objective function and the dynamics are continuous and differentiable almost everywhere with respect to $(\mathbf{X}, \mathbf{S}, u)$. Note that the choice of the relaxation parameter α plays an important role. We can guess that choosing a low value for α leads to a problem that may be easier to solve numerically than with a higher value of α . However, with a low value of α the relaxed dynamics and cost do not represent well the original problem whereas with a high α the original and relaxed dynamics are close. More details on the influence of the parameter α and how it is chosen in practice are given in §5.1.

4.4 Decomposition of the auxiliary problem into independent subproblems

By construction, the auxiliary problem (4.8) can be decomposed into independent subproblems on the components and a subproblem on the stock. The APP fixed-point algorithm is applied on the relaxed system, so we use the relaxed dynamics and cost in the writing of the subproblems. Formally, the gradients of Θ^α and Φ^α are not defined everywhere. By abuse of notation, the subproblems are given as if Θ^α and Φ^α were differentiable. Appendix D gives details on how we handle the points where the relaxed indicator is not differentiable.

The subproblem on component $i \in \mathbb{I}$ is

$$\begin{aligned} \min_{(\mathbf{X}_i, u_i) \in \mathcal{X}_i \times \mathcal{U}_i} \mathbb{E} & \left(j_i^\alpha(\mathbf{X}_i, u_i) + j^{F, \alpha}(\bar{\mathbf{X}}_{1:i-1}, \mathbf{X}_i, \bar{\mathbf{X}}_{i+1:n}) \right. \\ & + \frac{\gamma_x}{2} \|\mathbf{X}_i - \bar{\mathbf{X}}_i\|^2 + \frac{\gamma_u}{2} \|u_i - \bar{u}_i\|^2 \\ & + \left\langle \bar{\Lambda}_S, \partial_{\mathbf{X}_i} \Theta_S^\alpha(\bar{\mathbf{X}}_{1:n}, \bar{\mathbf{S}}) \cdot \mathbf{X}_i \right\rangle \\ & \left. + \sum_{j=i+1}^n \left\langle \bar{\Lambda}_j, \partial_{\mathbf{X}_i} \Theta_j^\alpha(\bar{\mathbf{X}}_{1:j}, \bar{\mathbf{S}}, \bar{u}_j, \mathbf{W}_j) \cdot \mathbf{X}_i \right\rangle \right) \\ \text{s.t. } & \Phi_i^\alpha(\mathbf{X}_i, u_i, \mathbf{W}_i) = 0. \end{aligned} \quad (4.16)$$

The subproblem on the stock is

$$\begin{aligned} \min_{\mathbf{S} \in \mathcal{S}} \mathbb{E} & \left(\frac{\gamma_s}{2} \|\mathbf{S} - \bar{\mathbf{S}}\|^2 + \sum_{i=1}^n \left\langle \bar{\Lambda}_i, \partial_{\mathbf{S}} \Theta_i^\alpha(\bar{\mathbf{X}}_{1:i}, \bar{\mathbf{S}}, \bar{u}_i, \mathbf{W}_i) \cdot \mathbf{S} \right\rangle \right) \\ \text{s.t. } & \Phi_S^\alpha(\mathbf{S}) = 0. \end{aligned} \quad (4.17)$$

4.5 The APP fixed-point algorithm for Problem (2.27)

We can now solve the original maintenance optimization problem using Algorithm 1. At each iteration, the resolution of the auxiliary problem consists in the parallel resolution of the independent subproblems (4.16)–(4.17).

The subproblems on the components are solved with the blackbox algorithm Mesh Adaptive Direct Search (MADS) [4]. At iteration k , MADS only outputs a primal solution $(\mathbf{X}_i^{k+1}, u_i^{k+1})$ of subproblem $i \in \mathbb{I}$. The optimal multiplier Λ_i^{k+1} is computed afterwards using the adjoint state. This technical point is addressed in Appendix C.

The subproblem on the stock (4.17) is very easy to solve numerically. The constraint $\Phi_S^\alpha(\mathbf{S}) = 0$ represents the dynamics of the stock with $\bar{\mathbf{X}} = (\bar{\mathbf{X}}_1, \dots, \bar{\mathbf{X}}_n)$ being fixed. The value of $\bar{\mathbf{X}}$ completely determines the dynamics of the stock. Hence, solving the subproblem on the stock just boils down to simulate its dynamics. The optimal multiplier is also computed using the adjoint state.

The features of the subproblem on the stock suggest to change the fully parallel strategy for the subproblem resolution into a mixed parallel/sequential strategy. At iteration k , the n subproblems on the components are solved in parallel with MADS. This yields a solution $(\mathbf{X}_i^{k+1}, u_i^{k+1}, \Lambda_i^{k+1})$ for each subproblem $i \in \mathbb{I}$. Then we can solve the subproblem on the stock at iteration k while immediately using the output of the subproblems on the components. This amounts to set before solving the subproblem on the stock at iteration k :

$$(\bar{\mathbf{X}}_1, \dots, \bar{\mathbf{X}}_n) = (\mathbf{X}_1^{k+1}, \dots, \mathbf{X}_n^{k+1}) \quad \text{and} \quad (\bar{\Lambda}_1, \dots, \bar{\Lambda}_n) = (\Lambda_1^{k+1}, \dots, \Lambda_n^{k+1}). \quad (4.18)$$

With this strategy we expect to reduce the number of iterations for convergence without penalizing the computation time per iteration as the subproblem on the stock can be solved in negligible time.

This results in an overall speed up of the algorithm. The APP fixed-point algorithm with the mixed parallel/sequential strategy is presented in Algorithm 2. The termination criteria is a maximum number of iterations M . This is the version that is used for numerical experiments.

Algorithm 2 APP fixed-point algorithm with a mixed parallel/sequential strategy

```

1: Start with  $(\bar{\mathbf{X}}, \bar{\mathbf{S}}, \bar{u}) = (\mathbf{X}^0, \mathbf{S}^0, u^0)$ ,  $\bar{\boldsymbol{\Lambda}} = \boldsymbol{\Lambda}^0$ 
2: for  $k = 0, \dots, M - 1$  do
3:   for  $i = 1, \dots, n$  do
4:     Solve the subproblem (4.16) on component  $i$ .
5:     Let  $(\mathbf{X}_i^{k+1}, u_i^{k+1})$  be a solution and  $\boldsymbol{\Lambda}_i^{k+1}$  be an optimal multiplier.
6:   end for
7:   Set  $(\bar{\mathbf{X}}, \bar{u}) = ((\mathbf{X}_1^{k+1}, \dots, \mathbf{X}_n^{k+1}), (u_1^{k+1}, \dots, u_n^{k+1}))$ 
8:   Set  $(\bar{\boldsymbol{\Lambda}}_1, \dots, \bar{\boldsymbol{\Lambda}}_n) = (\boldsymbol{\Lambda}_1^{k+1}, \dots, \boldsymbol{\Lambda}_n^{k+1})$ 
9:   Solve the subproblem (4.17) on the stock.
10:  Let  $\mathbf{S}^{k+1}$  be a solution and  $\boldsymbol{\Lambda}_{\mathbf{S}}^{k+1}$  be an optimal multiplier.
11:  Set  $\bar{\mathbf{S}} = \mathbf{S}^{k+1}$ 
12:  Set  $\bar{\boldsymbol{\Lambda}}_{\mathbf{S}} = \boldsymbol{\Lambda}_{\mathbf{S}}^{k+1}$ 
13: end for
14: return the maintenance strategy  $u^M$ 

```

5 Numerical results

In this part, we present the results of the decomposition methodology applied to Problem (2.27). The expectation in (2.27) cannot be evaluated exactly, so we solve a Monte-Carlo approximation of the problem with $Q = 100$ fixed failure scenarios $\omega_1, \dots, \omega_Q \in \Omega$:

$$\begin{aligned}
& \min_{(\mathbf{X}, \mathbf{S}, u) \in \mathcal{X} \times \mathcal{S} \times \mathcal{U}} \frac{1}{Q} \sum_{q=1}^Q \left(\sum_{i=1}^n j_i(\mathbf{X}_i(\omega_q), u_i) + j^F(\mathbf{X}_{1:n}(\omega_q)) \right) \\
& \text{s.t. } \Theta(\mathbf{X}(\omega_q), \mathbf{S}(\omega_q), u, \mathbf{W}(\omega_q)) = 0, \quad q \in \{1, \dots, Q\}.
\end{aligned} \tag{5.1}$$

The reference algorithm is the blackbox algorithm MADS applied directly on the original optimization problem (5.1). When running the optimization, the reference algorithm uses the original dynamics as it does not use gradient information. The APP fixed-point algorithm uses the relaxed dynamics. The maintenance strategies given by the two algorithms are then evaluated on a set of 10^5 failure scenarios, distinct from those used for the optimization. For the two strategies, the evaluation is done with the original dynamics of the system in order to ensure a fair comparison. For the numerical experiments, we consider a system with the characteristics given in Table 1.

5.1 Parameter tuning

Several parameters have to be tuned in order to apply the APP fixed-point algorithm. The parameters $\gamma_x, \gamma_s, \gamma_u$ appear in the auxiliary problem and α characterizes the relaxation of the system. These parameters are used in the APP fixed-point algorithm but not in the reference algorithm as the latter uses the original dynamics of the system to perform the optimization.

As the maintenance threshold ν is used both in the reference algorithm and the decomposition method, it is fixed to a value giving good performance with the reference algorithm. The same value

Parameter	Value		
Number of components n	80		
Initial number of spare parts \mathbf{S}_0	16		
Horizon T	40 years		
Time step Δt	1 year		
Number of time steps for supply D	2		
Discount rate τ	0.08		
Maintenance threshold ν	0.9		
Yearly forced outage cost C^F	10000 k€/ year		
	Comp. 1	Comp. 2	Comp. $i \geq 3$
PM cost C^P	50 k€	50 k€	50 k€
CM cost C^C	100 k€	250 k€	200 k€
Failure distribution	Weib(2.3, 10)	Weib(4, 20)	Weib(3, 10)
Mean time to failure	8.85 years	18.13 years	8.93 years

Table 1: Characteristics of the industrial system

of ν is used in the APP fixed-point algorithm. We do not consider changing the value of ν as it would mean that the reference algorithm changes for each different value of ν , making a fair comparison harder and the results less clear to analyze.

5.1.1 Description of the parameters

In the auxiliary problem (4.8), the value of $\gamma = (\gamma_x, \gamma_s, \gamma_u)$ influences the numerical behavior of the algorithm. We choose to increase the values of γ_x, γ_s and γ_u at each iteration of the APP fixed-point algorithm. The insight is that we can use low values of γ in the first iterations to get close to a good solution. Then, we use high values of γ to avoid oscillations of the solution of the auxiliary problem. Indeed, with a large value of γ , the solution $(\mathbf{X}^{k+1}, \mathbf{S}^{k+1}, u^{k+1})$ of the auxiliary problem at iteration $k+1$ is close to the previous solution $(\mathbf{X}^k, \mathbf{S}^k, u^k)$. The value of γ_u evolves from iteration k to $k+1$ of the APP fixed-point algorithm with a step $\Delta\gamma > 0$ so that

$$\gamma_u^{k+1} = \gamma_u^k + \Delta\gamma. \quad (5.2)$$

Then, γ_x and γ_s are chosen to be proportional to γ_u with ratios $r_x > 0$ and $r_s > 0$ respectively so that

$$\gamma_x^{k+1} = \gamma_u^{k+1}/r_x \quad \text{and} \quad \gamma_s^{k+1} = \gamma_u^{k+1}/r_s. \quad (5.3)$$

The motivation for this choice is that the vectors \mathbf{X} , \mathbf{S} and u do not have the same dimension so their norms may not have the same order of magnitude. The parameters $\gamma_u^0, r_x, r_s, \Delta\gamma$ have to be tuned.

The other parameter that requires attention is the relaxation parameter α . Similarly as for γ , we choose to increase the value of α at each iteration. A low value of α makes the problem easier to solve numerically but does not represent well the real problem. As α increases, the relaxed problem is closer and closer to the real one but becomes harder to solve. We use the solution of the auxiliary problem at the previous iteration as a warm start for the current iteration. The value of α varies from iteration k to $k+1$ of the APP fixed-point algorithm with a step $\Delta\alpha > 0$ so that

$$\alpha^{k+1} = \alpha^k + \Delta\alpha. \quad (5.4)$$

The values of α^0 and $\Delta\alpha$ have to be tuned. In the following, we call

$$p = (\gamma_u^0, r_x, r_s, \Delta\gamma, \alpha^0, \Delta\alpha) \in \mathbb{R}^6 \quad (5.5)$$

the vector of parameters that have to be adjusted for the algorithm.

5.1.2 Tuning methodology

Choosing a good value for p is difficult in practice, so we use the following methodology for the tuning. We start by defining bounds on the value of the parameters, they are given in Table 2. Then, 200 samples of p are drawn with an optimized Latin Hypercube Sampling [11]. We use a

Parameter	γ_u^0	r_x	r_s	$\Delta\gamma$	α^0	$\Delta\alpha$
Bounds	[1, 100]	[1, 10^4]	[1, 10^3]	[0, 100]	[2, 200]	[0, 200]

Table 2: Bounds for the parameters of the of the fixed-point algorithm

uniform distribution between the specified bounds for the sampling. Then, we run the APP fixed-point algorithm 200 times on a smaller system (10 components) than the one described in Table 1, each time with a different value of p from the sample. The 200 optimization processes are launched in parallel and are executed in about 4 hours. The value of p which gives the best result on the small system is used for the case with 80 components. The chosen value is

$$p = (17.32, 7434, 815.3, 1.360 \times 10^{-1}, 46.51, 135.5). \quad (5.6)$$

We give the value of p with 4 significant digits to emphasize that the performance of the APP fixed-point algorithm is very sensitive to this value. A sensitivity analysis has been performed using a Morris method [5] but no clear pattern for a good choice of p has been identified.

5.2 Comparison of the decomposition method with MADS

We run MADS and the decomposition method for the system of 80 components described in Table 1 and compare the performance of the maintenance strategies given by the two algorithms. The APP fixed-point algorithm uses the parameter p in (5.6) for the construction of the auxiliary problem and the relaxation of the system. Parameters of the computation are given in Table 3.

	Decomposition	MADS
Fixed-point iterations	50	/
Cost function calls	10^3 /subproblem/iteration	8×10^5
Cost and dynamics	Relaxed	Original
Processor model	Intel® Xeon® Processor E5-2680 v4, 2.4 GHz	
Computation time	18h24min	22h30min

Table 3: Parameters of the computation for the two algorithms

Remark 5.1. The APP fixed-point algorithm solves a decomposable auxiliary problem at each iteration, this algorithm is designed to be parallelized. It runs on 80 processors so that the subproblems on the components are solved in parallel. The reference algorithm MADS runs only on one processor. Note that it is also possible to parallelize MADS [3], although the implementation is not as immediate as for the decomposition method. The parallel version of MADS has not been tested. \diamond

The maintenance strategy computed by both algorithms is evaluated on a common set of 10^5 failure scenarios. These scenarios are distinct from the ones used for the optimization. The two algorithms give a maintenance strategy with $u_{i,t} \in [0, 1]$ for $(i, t) \in \mathbb{I} \times \mathbb{T}$. From the operational perspective, PMs make the components as good as new. Hence, for the evaluation of the strategy, the controls are projected on $\{0, 1\}$: we consider that if $u_{i,t} \geq \nu$ then the PM makes the component as good as new, otherwise no PM is performed. The comparison between the two maintenance strategies is fair as we use the same procedure for their evaluation.

The mean cost is 12902 k€ with MADS and 11483 k€ with the decomposition which represents a gain of 11%. The values of some quantiles are gathered in Table 4 and the distribution of the cost is represented on Figure 4.

	1%	5%	25%	50%	75%	95%	99%
Decomposition	10385	10683	11136	11472	11809	12326	12713
MADS	11858	12151	12588	12894	13211	13674	14010

Table 4: Quantiles of the cost of the two maintenance strategies (k€)

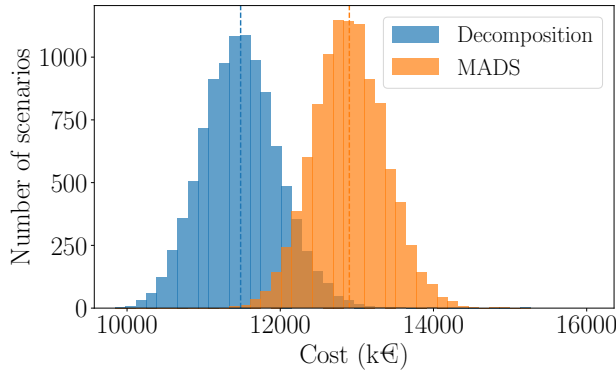


Figure 4: Distribution of the cost for the two maintenance strategies

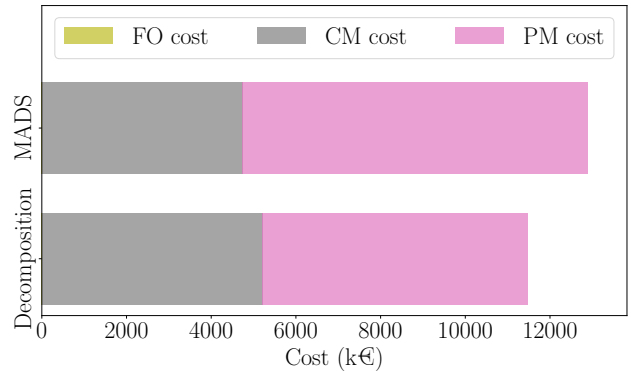


Figure 5: Part of the PM, CM and forced outage cost in the total expected cost

Figure 5 outlines that the average CM cost is higher with the decomposition strategy. However, a much lower PM cost makes the decomposition more efficient than MADS. This is due to the fact that fewer PMs are performed with the decomposition strategy than with MADS strategy (Table 5). The counterpart is that failures and forced outages occur more often with the decomposition strategy (Table 5). The forced outage cost is not visible on Figure 5 as it represents 0.05 k€ for MADS strategy (3.9×10^{-6} of the total expected cost) and 4.09 k€ for the decomposition strategy (3.5×10^{-4} of the total expected cost). There are more forced outages with the decomposition strategy (63 occurrences in 10^5 failure scenarios versus 1 for MADS) but they almost all occur in the last two time steps of the study horizon. Therefore the cost of forced outages is low because of the discount factor.

The cumulative number of PMs can be visualized on Figure 6. As already noticed there are fewer PMs with the decomposition strategy. A striking feature with the decomposition strategy is that there are almost no PM in the first three years. This exploits the fact that the components are new. The reference algorithm MADS applied directly on the original problem does not detect this feature. In fact, the region of the space corresponding to not doing any PM in first three years jointly for all components is a very small subset of the admissible space of the original problem and is not explored by MADS. On the other hand, the subproblems in the APP fixed-point algorithm act on an individual component, it is then easier to figure out that doing no PM in the first three years is profitable.

	Decomposition	MADS
Total number of PMs	447	558
Mean number of PMs/component	5.6	7.0
Mean time between PMs	6.1 years	5.0 years
Mean number of failures/component	1.40	1.18
Number of forced outages/Number of scenarios	63/10000	1/10000

Table 5: Overview of the number of PMs, failures and forced outages for each strategy

There is also a significant reduction of the number of PMs in the last five years of the study horizon. It is indeed useless to invest money to repair a component for the last few years. Moreover, the discount factor at the end of the horizon greatly reduces the incurred cost so that a forced outage is not too penalizing. This is why some forced outages occur with the decomposition strategy at the end of the study period.

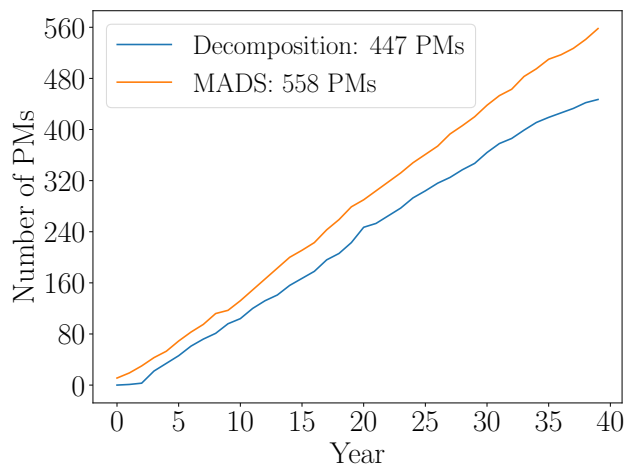


Figure 6: Cumulative number of PMs

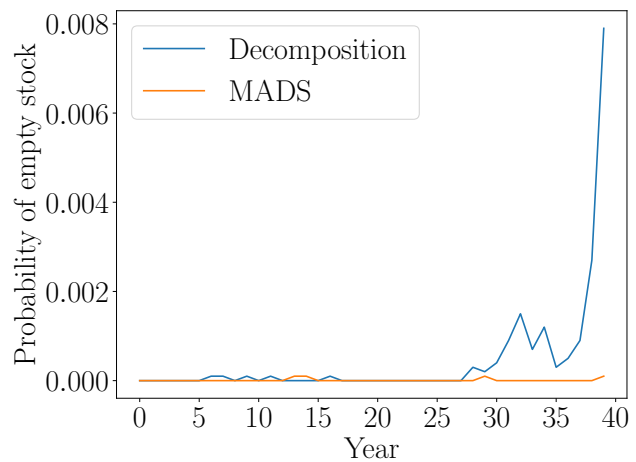


Figure 7: Evolution of the probability of having an empty stock

Another indicator that is monitored by decision makers is the level of stock. A necessary condition for the occurrence of a forced outage is that the stock is empty. Hence, we look at the probability of having an empty stock. The higher this probability is, the higher is the probability of forced outage. The probability of having an empty stock is very low for both strategies in the first 30 years and then increases for the decomposition strategy (Figure 7). Again, because of the discount factor, forced outages in the last few years do not have important financial consequences. It is then more profitable to do fewer PMs and allow for a higher risk of failure. This is what the decomposition strategy does.

Overall the strategy obtained by decomposition is more cost effective than MADS strategy. For a decision maker the decomposition strategy requires less investment as we do fewer PMs. It also has the best expected cost. Even in the case of extreme events, it is more robust than MADS strategy, as shown by the 99% quantile in Table 4. Indeed the forced outages may occur only at the end of the horizon.

6 Conclusion

In this work we study a maintenance scheduling optimization problem for hydropower plants management. We set up a decomposition method to find a deterministic preventive maintenance strategy

for a system of physical components sharing a common stock of spare parts. The decomposition relies on the Auxiliary Problem Principle. We construct a sequence of auxiliary problems that are solved iteratively. The auxiliary problems are decomposable into independent subproblems of smaller dimension that are solved in parallel. Each subproblem involves only one component of the system or the stock. A relaxation of the system is necessary as the gradients of the dynamics and the cost are required to compute the coordination terms.

On the industrial system, the decomposition method outperforms the blackbox algorithm MADS applied directly on the full problem. The strategy given by the decomposition involves fewer PMs especially at the beginning and the end of the time horizon, hence considerably reducing the investment. More forced outages occur but only at the end of the time horizon so without heavy financial consequences. It is also robust to extreme events as the 99% quantile is better for the decomposition strategy than for MADS.

This work proves the interest of the modelling effort needed to apply the decomposition method. Some challenges still remain for an application in an operational context. Here, the dynamics is simulated with a time step of one year. A smaller time step must be used for an accurate evaluation of the costs. This will not increase the complexity of the problem as maintenance decisions are always made on a yearly basis, so that the space of admissible maintenance strategies is still of the same dimension. However, the time needed for the evaluation of the cost function will increase. It is also possible to model more complex systems, by adding a control on the time of the order of spare parts or dependence between the failures of the components for instance. We could also consider imperfect preventive maintenance. A balance must be found between the simplicity of the model and its adequation to reality given the industrial application in mind.

A Explicit expression of the dynamics of the original system

In this part, we give an explicit expression for the dynamics f_i of component $i \in \mathbb{I}$ that appears in (2.19). We can write

$$\mathbf{X}_{i,t+1} = \begin{pmatrix} \mathbf{E}_{i,t+1} \\ \mathbf{A}_{i,t+1} \\ \mathbf{P}_{i,t+1} \end{pmatrix} = \begin{pmatrix} f_{i,\mathbf{E}}(\mathbf{X}_{1:i,t}, \mathbf{S}_t, u_{i,t}, \mathbf{W}_{i,t+1}) \\ f_{i,\mathbf{A}}(\mathbf{X}_{1:i,t}, \mathbf{S}_t, u_{i,t}, \mathbf{W}_{i,t+1}) \\ f_{i,\mathbf{P}}(\mathbf{X}_{1:i,t}, \mathbf{S}_t, u_{i,t}, \mathbf{W}_{i,t+1}) \end{pmatrix}, \quad (\text{A.1})$$

so that $f_i = (f_{i,\mathbf{E}}, f_{i,\mathbf{A}}, f_{i,\mathbf{P}})$. We give an explicit formula for $f_{i,\mathbf{E}}$, $f_{i,\mathbf{A}}$ and $f_{i,\mathbf{P}}$.

A.1 Dynamics of the regime $\mathbf{E}_{i,t}$

Using Figure 2, we can write

$$\begin{aligned} \mathbf{E}_{i,t+1} &= f_{i,\mathbf{E}}(\mathbf{X}_{1:i,t}, \mathbf{S}_t, u_{i,t}, \mathbf{W}_{i,t+1}) \\ &= \mathbf{1}_{\mathbb{R}_+}(\mathbf{S}_t - \sum_{j=1}^i \mathbf{1}_{\{0\}}(\mathbf{E}_{j,t})) \mathbf{1}_{\{0\}}(\mathbf{E}_{i,t}) \\ &\quad + (\mathbf{1}_{\mathbb{R}_+}(u_{i,t} - \nu) + \mathbf{1}_{\mathbb{R}_+}(\mathbf{W}_{i,t+1} - p_i(\mathbf{A}_{i,t})) \mathbf{1}_{\mathbb{R}_+^*}(\nu - u_{i,t})) \mathbf{1}_{\{1\}}(\mathbf{E}_{i,t}). \end{aligned} \quad (\text{A.2})$$

The first part of (A.2) means that if the component is broken at t and we have enough spares to repair it, it is then functioning at $t+1$. The second part means that if the component is functioning at t and we do a PM, it is still functioning at $t+1$. Finally, if we do not do a PM, the regime depends on the occurrence of a failure between t and $t+1$.

A.2 Dynamics of the age $\mathbf{A}_{i,t}$

Again using Figure 2, we can write

$$\begin{aligned} \mathbf{A}_{i,t+1} &= f_{i,A}(\mathbf{X}_{1:i,t}, \mathbf{S}_t, u_{i,t}, \mathbf{W}_{i,t+1}) \\ &= (\mathbf{A}_{i,t} + 1) \left[\mathbf{1}_{\mathbb{R}_+^*} \left(\sum_{j=1}^i \mathbf{1}_{\{0\}}(\mathbf{E}_{j,t}) - \mathbf{S}_t \right) \mathbf{1}_{\{0\}}(\mathbf{E}_{i,t}) + ((1 - u_{i,t}) \mathbf{1}_{\mathbb{R}_+}(u_{i,t} - \nu) \right. \\ &\quad \left. + \mathbf{1}_{\mathbb{R}_+}(\mathbf{W}_{i,t+1} - p_i(\mathbf{A}_{i,t})) \mathbf{1}_{\mathbb{R}_+^*}(\nu - u_{i,t}) \right) \mathbf{1}_{\{1\}}(\mathbf{E}_{i,t}) \right]. \end{aligned} \quad (\text{A.3})$$

If the component is broken at t , it stays broken if there are not enough spares in the stock. In this case the time $\mathbf{A}_{i,t}$ increases by 1. If the component is healthy at t , it ages if no PM is done and no failure occurs. In the case of a PM the component is rejuvenated. If there is a failure, we have $\mathbf{A}_{i,t+1} = 0$.

A.3 Dynamics of the vector of times since last failures $\mathbf{P}_{i,t}$

The expression of the dynamics of $\mathbf{P}_{i,t} = (\mathbf{P}_{i,t}^1, \dots, \mathbf{P}_{i,t}^D)$ is more complex. We write

$$\mathbf{P}_{i,t+1} = \begin{pmatrix} \mathbf{P}_{i,t+1}^1 \\ \vdots \\ \mathbf{P}_{i,t+1}^D \end{pmatrix} = \begin{pmatrix} f_{i,\mathbf{P}}^1(\mathbf{X}_{1:i,t}, \mathbf{S}_t, u_{i,t}, \mathbf{W}_{i,t+1}) \\ \vdots \\ f_{i,\mathbf{P}}^D(\mathbf{X}_{1:i,t}, \mathbf{S}_t, u_{i,t}, \mathbf{W}_{i,t+1}) \end{pmatrix} = f_{i,\mathbf{P}}(\mathbf{X}_{1:i,t}, \mathbf{S}_t, u_{i,t}, \mathbf{W}_{i,t+1}), \quad (\text{A.4})$$

so that $f_{i,\mathbf{P}} = (f_{i,\mathbf{P}}^1, \dots, f_{i,\mathbf{P}}^D)$. We give the expression of $f_{i,\mathbf{P}}^d$ for $d \in \{1, \dots, D\}$:

$$\begin{aligned} \mathbf{P}_{i,t+1}^d &= \left((\mathbf{P}_{i,t}^d + 1) \mathbf{1}_{\mathbb{R}_+}(\mathbf{P}_{i,t}^d) + \delta \mathbf{1}_{\{\delta\}}(\mathbf{P}_{i,t}^d) \right) \left(1 - \mathbf{1}_{\{1\}}(\mathbf{E}_{i,t}) \mathbf{1}_{\{0\}}(\mathbf{E}_{i,t+1}) \right) \\ &\quad + \left((\mathbf{P}_{i,t}^d + 1) \mathbf{1}_{\mathbb{R}_+}(\mathbf{P}_{i,t}^d) \mathbf{1}_{\{\delta\}}(\mathbf{P}_{i,t}^D) + \delta \mathbf{1}_{\{\delta\}}(\mathbf{P}_{i,t}^{d-1}) \mathbf{1}_{[2,D]}(d) \right) \\ &\quad + \left((\mathbf{P}_{i,t}^{d+1} + 1) \mathbf{1}_{\mathbb{R}_+}(\mathbf{P}_{i,t}^D) \mathbf{1}_{[1,D-1]}(d) \right) \mathbf{1}_{\{1\}}(\mathbf{E}_{i,t}) \mathbf{1}_{\{0\}}(\mathbf{E}_{i,t+1}). \end{aligned} \quad (\text{A.5})$$

The first line represents the case where there is no failure. Then $\mathbf{P}_{i,t}$ increases by one if it is different from δ , otherwise it keeps the value δ . When there is a failure, if component i has undergone fewer than D failures, the evolution of $\mathbf{P}_{i,t}$ is described by (2.16a). This case is represented by the second line of (A.5). When the component has already undergone D failures, the evolution of $\mathbf{P}_{i,t}$ is described by (2.17a). This case is represented by the third line of (A.5). Note that this expression of $\mathbf{P}_{i,t+1}^d$ depends on $\mathbf{E}_{i,t+1}$. It is possible to express $\mathbf{P}_{i,t+1}^d$ only with variables describing component i at time t , this can be done by replacing $\mathbf{E}_{i,t+1}$ by its expression (A.2).

A.4 Dynamics of the stock \mathbf{S}_t

We recall the explicit dynamics of the stock that is already given in (2.20):

$$\mathbf{S}_{t+1} = \mathbf{S}_t + \sum_{i=1}^n \sum_{d=1}^D \mathbf{1}_{\{D-1\}}(\mathbf{P}_{i,t}^d) - \min \left\{ \mathbf{S}_t, \sum_{i=1}^n \mathbf{1}_{\{0\}}(\mathbf{E}_{i,t}) \right\}. \quad (\text{A.6})$$

The dynamics of the whole system has now been explicitly described.

B Explicit expression of the dynamics of the relaxed system

The expression of the relaxed dynamics of parameter α is obtained from Equations (A.2), (A.3), (A.5) and (A.6) by replacing the indicator function with its relaxed version.

We do not always substitute directly the indicator with its relaxation. The dynamics often involves conditions on complementary events. For example, the condition *if the component is broken* is represented by $\mathbf{1}_{\{0\}}(\mathbf{E}_{i,t})$. On the other hand, the condition *if the component is healthy* is represented by $\mathbf{1}_{\{1\}}(\mathbf{E}_{i,t})$. For the original dynamics as $\mathbf{E}_{i,t} \in \{0, 1\}$ we always have

$$\mathbf{1}_{\{0\}}(\mathbf{E}_{i,t}) + \mathbf{1}_{\{1\}}(\mathbf{E}_{i,t}) = 1. \quad (\text{B.1})$$

This relation is not true anymore using directly the relaxed version of the indicator with the relaxed variables. Take for example $\alpha = 2$, and suppose $\mathbf{E}_{i,t} = \frac{1}{2}$, then

$$\mathbf{1}_{\{0\}}^\alpha(\mathbf{E}_{i,t}) + \mathbf{1}_{\{1\}}^\alpha(\mathbf{E}_{i,t}) = 0. \quad (\text{B.2})$$

If we replace directly all indicator functions by their relaxation, the consequence would be in this case that $\mathbf{E}_{i,t+1} = 0$ no matter the control $u_{i,t}$. This means that even if we do a PM with $u_{i,t} = 1$, the component is down at $t + 1$. This does not represent the dynamics of the system as we would expect. To design a coherent relaxed dynamics, complementary conditions $\mathbf{1}_{\mathcal{A}}$ and $\mathbf{1}_{\mathcal{A}^c}$ are represented using the relaxed version $\mathbf{1}_{\mathcal{A}}^\alpha$ of the indicator function for the first condition and the function $1 - \mathbf{1}_{\mathcal{A}}^\alpha$ for the complementary condition.

B.1 Relaxed dynamics of the regime $\mathbf{E}_{i,t}$

The relaxed dynamics of the regime of parameter $\alpha > 0$ is given by

$$\begin{aligned} \mathbf{E}_{i,t+1} &= f_{i,\mathbf{E}}^\alpha(\mathbf{X}_{1:i,t}, \mathbf{S}_t, u_{i,t}, \mathbf{W}_{i,t+1}) \\ &= \mathbf{1}_{\mathbb{R}_+}^\alpha(\mathbf{S}_t - \sum_{j=1}^i \mathbf{1}_{\{0\}}^\alpha(\mathbf{E}_{j,t})) \mathbf{1}_{\{0\}}^\alpha(\mathbf{E}_{i,t}) + \left(\mathbf{1}_{\mathbb{R}_+}^\alpha(u_{i,t} - \nu) \right. \\ &\quad \left. + \mathbf{1}_{\mathbb{R}_+}^\alpha(\mathbf{W}_{i,t+1} - p_i(\mathbf{A}_{i,t}))(1 - \mathbf{1}_{\mathbb{R}_+}^\alpha(u_{i,t} - \nu)) \right) \left(1 - \mathbf{1}_{\{0\}}^\alpha(\mathbf{E}_{i,t}) \right). \end{aligned} \quad (\text{B.3})$$

We use the relaxed version of the indicator for $\mathbf{1}_{\{0\}}^\alpha(\mathbf{E}_{i,t})$ and $\mathbf{1}_{\mathbb{R}_+}^\alpha(u_{i,t} - \nu)$. We relax $\mathbf{1}_{\{1\}}^\alpha(\mathbf{E}_{i,t})$ and $\mathbf{1}_{\mathbb{R}_+}^*(\nu - u_{i,t})$ as $1 - \mathbf{1}_{\{0\}}^\alpha(\mathbf{E}_{i,t})$ and $1 - \mathbf{1}_{\mathbb{R}_+}^\alpha(u_{i,t} - \nu)$ respectively.

B.2 Relaxed dynamics of the age $\mathbf{A}_{i,t}$

The relaxed dynamics of the age of parameter $\alpha > 0$ is given by

$$\begin{aligned} \mathbf{A}_{i,t+1} &= f_{i,\mathbf{A}}^\alpha(\mathbf{X}_{1:i,t}, \mathbf{S}_t, u_{i,t}, \mathbf{W}_{i,t+1}) \\ &= (\mathbf{A}_{i,t} + 1) \left[\mathbf{1}_{\mathbb{R}_+}^{\alpha*} \left(\sum_{j=1}^i \mathbf{1}_{\{0\}}^\alpha(\mathbf{E}_{j,t}) - \mathbf{S}_t \right) \mathbf{1}_{\{0\}}^\alpha(\mathbf{E}_{i,t}) + ((1 - u_{i,t}) \mathbf{1}_{\mathbb{R}_+}^\alpha(u_{i,t} - \nu) \right. \\ &\quad \left. + \mathbf{1}_{\mathbb{R}_+}^\alpha(\mathbf{W}_{i,t+1} - p_i(\mathbf{A}_{i,t}))(1 - \mathbf{1}_{\mathbb{R}_+}^\alpha(u_{i,t} - \nu)) \right) (1 - \mathbf{1}_{\{0\}}^\alpha(\mathbf{E}_{i,t})) \right]. \end{aligned} \quad (\text{B.4})$$

B.3 Relaxed dynamics of the vector of last failures $\mathbf{P}_{i,t}$

The relaxed dynamics of parameter $\alpha > 0$ of the d -th element of the vector of last failures is given by

$$\begin{aligned} \mathbf{P}_{i,t+1}^d &= f_{i,\mathbf{P}}^{d,\alpha}(\mathbf{X}_{1:i,t}, \mathbf{S}_t, u_{i,t}, \mathbf{W}_{i,t+1}) \\ &= \left((\mathbf{P}_{i,t}^d + 1)(1 - \mathbf{1}_{\{\delta\}}^\alpha(\mathbf{P}_{i,t}^d)) + \delta \mathbf{1}_{\{\delta\}}^\alpha(\mathbf{P}_{i,t}^d) \right) \left(1 - \mathbf{1}_{\{1\}}^\alpha(\mathbf{E}_{i,t}) \mathbf{1}_{\{0\}}^\alpha(\mathbf{E}_{i,t+1}) \right) \\ &\quad + \left((\mathbf{P}_{i,t}^d + 1)(1 - \mathbf{1}_{\{\delta\}}^\alpha(\mathbf{P}_{i,t}^d)) \mathbf{1}_{\{\delta\}}^\alpha(\mathbf{P}_{i,t}^D) + \delta \mathbf{1}_{\{\delta\}}^\alpha(\mathbf{P}_{i,t}^{d-1}) \mathbf{1}_{[2,D]}(d) \right) \\ &\quad + (\mathbf{P}_{i,t}^{d+1} + 1)(1 - \mathbf{1}_{\{\delta\}}^\alpha(\mathbf{P}_{i,t}^D)) \mathbf{1}_{[1,D-1]}(d) \mathbf{1}_{\{1\}}^\alpha(\mathbf{E}_{i,t}) \mathbf{1}_{\{0\}}^\alpha(\mathbf{E}_{i,t+1}). \end{aligned} \quad (\text{B.5})$$

We do not relax $\mathbf{1}_{[2,D]}(d)$ and $\mathbf{1}_{[1,D-1]}(d)$. The reason is that these indicator functions do not arise from a discontinuity in the original dynamics. They are just used to take into account in the same equation the cases of $\mathbf{P}_{i,t}^1$ and $\mathbf{P}_{i,t}^D$ that have a slightly different expression than $\mathbf{P}_{i,t}^d$ for $1 < d < D$.

B.4 Relaxed dynamics of the stock \mathbf{S}_t

The relaxed dynamics of the stock of parameter $\alpha > 0$ is given by

$$\mathbf{S}_{t+1} = \mathbf{S}_t + \sum_{i=1}^n \sum_{d=1}^D \mathbf{1}_{\{D-1\}}^\alpha(\mathbf{P}_{i,t}^d) - \min \left\{ \mathbf{S}_t, \sum_{i=1}^n \mathbf{1}_{\{0\}}^\alpha(\mathbf{E}_{i,t}) \right\}. \quad (\text{B.6})$$

C Computation of optimal multipliers

At iteration k of the APP fixed-point algorithm, the subproblem on component $i \in \mathbb{I}$ is solved with the blackbox algorithm MADS [4]. MADS directly solves the constrained problem and outputs a primal solution $(\mathbf{X}_i^{k+1}, u_i^{k+1})$. Finding the primal solution \mathbf{S}^{k+1} of the subproblem on the stock just requires a simulation of the dynamics. For each subproblem we also have to compute optimal multipliers $\Lambda_1^{k+1}, \dots, \Lambda_n^{k+1}, \Lambda_{\mathbf{S}}^{k+1}$ to update the coordination term at the end of each iteration.

Suppose that the optimal solution and optimal multiplier of the auxiliary problem (4.8) are uniquely defined. As we know the primal solution, we can compute the optimal multiplier using the stationarity of the Lagrangian. In the following calculation, we use the relaxed cost and dynamics to be able to compute the different gradients that appear, however for the sake of readability we drop the superscript α .

The Lagrangian L of the auxiliary problem (4.8) is

$$\begin{aligned} L(\mathbf{X}, \mathbf{S}, u, \Lambda) &= \mathbb{E} \left(\sum_{i=1}^n \left(j_i(\mathbf{X}_i, u_i) + j^F(\bar{\mathbf{X}}_{1:i-1}, \mathbf{X}_i, \bar{\mathbf{X}}_{i+1:n}) \right) \right) \\ &\quad + \frac{\gamma_x}{2} \|\mathbf{X} - \bar{\mathbf{X}}\|^2 + \frac{\gamma_s}{2} \|\mathbf{S} - \bar{\mathbf{S}}\|^2 + \frac{\gamma_u}{2} \|u - \bar{u}\|^2 \\ &\quad + \langle \bar{\Lambda}, (\Theta'(\bar{\mathbf{X}}, \bar{\mathbf{S}}, \bar{u}, \mathbf{W}) - \Phi'(\bar{\mathbf{X}}, \bar{\mathbf{S}}, \bar{u}, \mathbf{W})) \cdot (\mathbf{X}, \mathbf{S}, u) \rangle \\ &\quad + \langle \Lambda, \Phi(\mathbf{X}, \mathbf{S}, u, \mathbf{W}) \rangle \Big). \end{aligned} \quad (\text{C.1})$$

At the saddle point $(\mathbf{X}^\sharp, \mathbf{S}^\sharp, u^\sharp, \Lambda^\sharp)$ of L we have:

$$\nabla L(\mathbf{X}^\sharp, \mathbf{S}^\sharp, u^\sharp, \Lambda^\sharp) = 0. \quad (\text{C.2})$$

Recall that for $i \in \{1, \dots, n, \mathbf{S}\}$, we have

$$\Lambda_i^{k+1} = (\Lambda_{i,0}^{k+1}, \dots, \Lambda_{i,T}^{k+1}). \quad (\text{C.3})$$

Using (C.2) and knowing the solution $(\mathbf{X}^\sharp, \mathbf{S}^\sharp, u^\sharp)$ of the auxiliary problem, we can update the multiplier Λ^\sharp with a backward recursion.

Proposition C.1. Let $i \in \mathbb{I}$. For the dynamics of component i in the auxiliary problem (4.8), the optimal multiplier $\Lambda_i^\# = (\Lambda_{i,0}^\#, \dots, \Lambda_{i,T}^\#)$ can be computed with the following backward recursion for $t \in \mathbb{T}$:

$$\begin{aligned}
\Lambda_{i,T}^\# &= -\nabla_{\mathbf{X}_{i,T}} j_{i,T}(\mathbf{X}_{i,T}^\#, u_{i,T}^\#) - \nabla_{\mathbf{X}_{i,T}} j_T^F(\bar{\mathbf{X}}_{1:i-1,T}, \mathbf{X}_{i,T}^\#, \bar{\mathbf{X}}_{i+1:n,T}) \\
&\quad - \gamma_x(\mathbf{X}_{i,T}^\# - \bar{\mathbf{X}}_{i,T}), \\
\Lambda_{i,t}^\# &= -\nabla_{\mathbf{X}_{i,t}} j_{i,t}(\mathbf{X}_{i,t}^\#, u_{i,t}^\#) - \nabla_{\mathbf{X}_{i,t}} j_t^F(\bar{\mathbf{X}}_{1:i-1,t}, \mathbf{X}_{i,t}^\#, \bar{\mathbf{X}}_{i+1:n,t}) \\
&\quad - \gamma_x(\mathbf{X}_{i,t}^\# - \bar{\mathbf{X}}_{i,t}) - \sum_{j=i+1}^n \partial_{\mathbf{X}_{i,t}} \Theta_{j,t+1}(\bar{\mathbf{X}}_{1:j}, \bar{\mathbf{S}}, \bar{u}_j, \mathbf{W}_j)^\top \cdot \bar{\Lambda}_{j,t+1} \\
&\quad - \partial_{\mathbf{X}_{i,t}} \Theta_{S,t+1}(\bar{\mathbf{X}}_{1:n}, \bar{\mathbf{S}})^\top \cdot \bar{\Lambda}_{S,t+1} - \partial_{\mathbf{X}_{i,t}} \Phi_{i,t+1}(\mathbf{X}_i^\#, u_i^\#, \mathbf{W}_i)^\top \cdot \Lambda_{i,t+1}^\#.
\end{aligned} \tag{C.4}$$

Proof. The gradient of the Lagrangian L with respect to $\mathbf{X}_{i,t}$, $t \in \mathbb{T}$ is given by

$$\begin{aligned}
\nabla_{\mathbf{X}_{i,T}} L(\mathbf{X}^\#, \mathbf{S}^\#, u^\#, \Lambda^\#) &= \nabla_{\mathbf{X}_{i,T}} j_{i,T}(\mathbf{X}_{i,T}^\#, u_{i,T}^\#) \\
&\quad + \nabla_{\mathbf{X}_{i,T}} j_T^F(\bar{\mathbf{X}}_{1:i-1,T}, \mathbf{X}_{i,T}^\#, \bar{\mathbf{X}}_{i+1:n,T}) \\
&\quad + \gamma_x(\mathbf{X}_{i,T}^\# - \bar{\mathbf{X}}_{i,T}) + \partial_{\mathbf{X}_{i,T}} \Phi_{i,T}(\mathbf{X}_i^\#, u_i^\#, \mathbf{W}_i)^\top \cdot \Lambda_{i,T}^\#, \\
\nabla_{\mathbf{X}_{i,t}} L(\mathbf{X}^\#, \mathbf{S}^\#, u^\#, \Lambda^\#) &= \nabla_{\mathbf{X}_{i,t}} j_{i,t}(\mathbf{X}_{i,t}^\#, u_{i,t}^\#) + \nabla_{\mathbf{X}_{i,t}} j_t^F(\bar{\mathbf{X}}_{1:i-1,t}, \mathbf{X}_{i,t}^\#, \bar{\mathbf{X}}_{i+1:n,t}) \\
&\quad + \gamma_x(\mathbf{X}_{i,t}^\# - \bar{\mathbf{X}}_{i,t}) + \partial_{\mathbf{X}_{i,t}} \Theta_{S,t+1}(\bar{\mathbf{X}}_{1:n}, \bar{\mathbf{S}})^\top \cdot \bar{\Lambda}_{S,t+1} \\
&\quad + \sum_{j=i+1}^n \partial_{\mathbf{X}_{i,t}} \Theta_{j,t+1}(\bar{\mathbf{X}}_{1:j}, \bar{\mathbf{S}}, \bar{u}_j, \mathbf{W}_j)^\top \cdot \bar{\Lambda}_{j,t+1} \\
&\quad + \partial_{\mathbf{X}_{i,t}} \Phi_{i,t}(\mathbf{X}_i^\#, u_i^\#, \mathbf{W}_i)^\top \cdot \Lambda_{i,t}^\# \\
&\quad + \partial_{\mathbf{X}_{i,t}} \Phi_{i,t+1}(\mathbf{X}_i^\#, u_i^\#, \mathbf{W}_i)^\top \cdot \Lambda_{i,t+1}^\#.
\end{aligned} \tag{C.5}$$

Using

$$\nabla_{\mathbf{X}_{i,t}} L(\mathbf{X}^\#, \mathbf{S}^\#, u^\#, \Lambda^\#) = 0, \quad t \in \mathbb{T}, \tag{C.6}$$

$$\partial_{\mathbf{X}_{i,t}} \Phi_{i,t}(\mathbf{X}_i^\#, u_i^\#, \mathbf{W}_i) = \mathbf{I}, \quad t \in \mathbb{T}, \tag{C.7}$$

where \mathbf{I} is the identity matrix of appropriate size, we get the formula (C.4).

Proposition C.2. The optimal multiplier $\Lambda_S^\# = (\Lambda_{S,0}^\#, \dots, \Lambda_{S,T}^\#)$ associated to the dynamics of the stock in the auxiliary problem (4.8) can be computed with the following backward recursion for $t \in \mathbb{T}$:

$$\begin{aligned}
\Lambda_{S,T}^\# &= -\gamma_s(\mathbf{S}_T^\# - \bar{\mathbf{S}}_T), \\
\Lambda_{S,t}^\# &= -\gamma_s(\mathbf{S}_t^\# - \bar{\mathbf{S}}_t) - \sum_{i=1}^n \partial_{\mathbf{S}_t} \Theta_{i,t+1}(\bar{\mathbf{X}}_{1:i}, \bar{\mathbf{S}}, \bar{u}_i, \mathbf{W}_i)^\top \cdot \bar{\Lambda}_{i,t+1} \\
&\quad - \partial_{\mathbf{S}_t} \Phi_{S,t+1}(\mathbf{S}^\#)^\top \cdot \Lambda_{S,t+1}^\#.
\end{aligned} \tag{C.8}$$

Proof. The gradient of the Lagrangian L with respect to \mathbf{S}_t , $t \in \mathbb{T}$ is given by

$$\begin{aligned}\nabla_{\mathbf{S}_T} L(\mathbf{X}^\#, \mathbf{S}^\#, u^\#, \Lambda^\#) &= \gamma(\mathbf{S}_T^\# - \bar{\mathbf{S}}_T) + \partial_{\mathbf{S}_T} \Phi_{\mathbf{S}, T}(\mathbf{S}^\#)^\top \cdot \Lambda_{\mathbf{S}, T}^\#, \\ \nabla_{\mathbf{S}_t} L(\mathbf{X}^\#, \mathbf{S}^\#, u^\#, \Lambda^\#) &= \gamma(\mathbf{S}_t^\# - \bar{\mathbf{S}}_t) \\ &\quad + \sum_{i=1}^n \partial_{\mathbf{S}_t} \Theta_{i, t+1}(\bar{\mathbf{X}}_{1:i}, \bar{\mathbf{S}}, \bar{u}_i, \mathbf{W}_i)^\top \cdot \bar{\Lambda}_{i, t+1} \\ &\quad + \partial_{\mathbf{S}_t} \Phi_{\mathbf{S}, t+1}(\mathbf{S}^\#)^\top \cdot \Lambda_{\mathbf{S}, t+1}^\# + \partial_{\mathbf{S}_t} \Phi_{\mathbf{S}, t}(\mathbf{S}^\#)^\top \cdot \Lambda_{\mathbf{S}, t}^\#.\end{aligned}\tag{C.9}$$

Using

$$\nabla_{\mathbf{S}_t} L(\mathbf{X}^\#, \mathbf{S}^\#, u^\#, \Lambda^\#) = 0, \quad t \in \mathbb{T}, \tag{C.10}$$

$$\partial_{\mathbf{S}_t} \Phi_{\mathbf{S}, t}(\mathbf{S}^\#) = 1, \quad t \in \mathbb{T}, \tag{C.11}$$

we get the backward recursion (C.8).

D Derivative of the relaxed indicator function

We give some details about the derivative of the relaxed indicator function. The relaxed indicator function $\mathbf{1}_{\mathcal{A}}^\alpha$, where $\alpha > 0$, appears in the dynamics and cost with three main cases for the set $\mathcal{A} \subset \mathbb{R}$. Note that $\mathbf{1}_{\mathcal{A}}^\alpha$ is not differentiable at $x \in \mathbb{R}$ if $d(\mathcal{A}, x) = \frac{1}{2\alpha}$. At such point, the derivative is taken to be 0. The following situations occur:

1. \mathcal{A} is a singleton $\{a\}$, then for $x \in \mathbb{R}$:

$$\mathbf{1}_{\{a\}}^\alpha(x) = \begin{cases} 1 - 2\alpha|x - a| & \text{if } |x - a| \leq \frac{1}{2\alpha}, \\ 0 & \text{if } |x - a| > \frac{1}{2\alpha}. \end{cases}\tag{D.1}$$

Hence the derivative $\mathbf{1}'_{\{a\}}^\alpha$ is given by:

$$\mathbf{1}'_{\{a\}}^\alpha(x) = \begin{cases} 2\alpha & \text{if } a - \frac{1}{2\alpha} < x < a, \\ -2\alpha & \text{if } a < x < a + \frac{1}{2\alpha}, \\ 0 & \text{otherwise.} \end{cases}\tag{D.2}$$

2. $\mathcal{A} = \mathbb{R}_+$ then for $x \in \mathbb{R}$ we have:

$$\mathbf{1}_{\mathbb{R}_+}^\alpha(x) = \begin{cases} 2\alpha x + 1 & \text{if } -\frac{1}{2\alpha} < x < 0, \\ 1 & \text{if } x \geq 0, \\ 0 & \text{if } x \leq -\frac{1}{2\alpha}. \end{cases} \quad \mathbf{1}'_{\mathbb{R}_+}^\alpha(x) = \begin{cases} 2\alpha & \text{if } -\frac{1}{2\alpha} < x < 0, \\ 0 & \text{otherwise.} \end{cases}\tag{D.3}$$

3. $\mathcal{A} = \mathbb{R}_+^*$: if we strictly apply Definition 4.2, we would have $\mathbf{1}_{\mathbb{R}_+^*}^\alpha = \mathbf{1}_{\mathbb{R}_+}^\alpha$. However with this definition we would not have pointwise convergence of $\mathbf{1}_{\mathbb{R}_+^*}^\alpha$ towards $\mathbf{1}_{\mathbb{R}_+^*}$ as α goes to 0. Indeed, for all $\alpha > 0$ we would have $\mathbf{1}_{\mathbb{R}_+^*}^\alpha(0) = 1$ but $\mathbf{1}_{\mathbb{R}_+^*}(0) = 0$. To overcome this issue we define $\mathbf{1}_{\mathbb{R}_+^*}^\alpha$ as follows:

$$\mathbf{1}_{\mathbb{R}_+^*}^\alpha(x) = \begin{cases} 2\alpha x & \text{if } 0 < x < \frac{1}{2\alpha}, \\ 1 & \text{if } x \geq \frac{1}{2\alpha}, \\ 0 & \text{if } x \leq 0. \end{cases} \quad \mathbf{1}'_{\mathbb{R}_+^*}^\alpha(x) = \begin{cases} 2\alpha & \text{if } 0 < x < \frac{1}{2\alpha}, \\ 0 & \text{otherwise.} \end{cases}\tag{D.4}$$

Using these formulas for the derivative of the relaxed indicator function and the explicit expressions of the relaxed cost function and relaxed dynamics given in §4.3.3 and Appendix B respectively, all the gradients that appear either in the objective function of the subproblems or in the backward recursion for the multiplier update can be computed.

References

- [1] A. ALRABGHI AND A. TIWARI, *State of the art in simulation-based optimisation for maintenance systems*, Computers & Industrial Engineering, 82 (2015), pp. 167–182.
- [2] K. J. ARROW AND L. HURWICZ, *Decentralization and Computation in Resource Allocation*, stanford university, department of economics ed., 1960.
- [3] C. AUDET, J. E. DENNIS, AND S. L. DIGABEL, *Parallel Space Decomposition of the Mesh Adaptive Direct Search Algorithm*, SIAM Journal on Optimization, 19 (2008), pp. 1150–1170.
- [4] C. AUDET AND J. E. DENNIS, JR, *Mesh Adaptive Direct Search Algorithms for Constrained Optimization*, SIAM Journal on Optimization, 17 (2006), pp. 188–217.
- [5] F. CAMPOLONGO, J. CARIBONI, AND A. SALTELLI, *An effective screening design for sensitivity analysis of large models*, Environmental Modelling & Software, 22 (2007), pp. 1509–1518.
- [6] P. CARPENTIER AND G. COHEN, *Décomposition-coordination en optimisation déterministe et stochastique*, vol. 81 of Mathématiques et Applications, Springer Berlin Heidelberg, Berlin, Heidelberg, 2017.
- [7] M.-C. CHEN, C.-M. HSU, AND S.-W. CHEN, *Optimizing joint maintenance and stock provisioning policy for a multi-echelon spare parts logistics network*, Journal of the Chinese Institute of Industrial Engineers, 23 (2006), pp. 289–302.
- [8] D. I. CHO AND M. PARLAR, *A survey of maintenance models for multi-unit systems*, European Journal of Operational Research, 51 (1991), pp. 1–23.
- [9] G. COHEN, *Optimization by decomposition and coordination: A unified approach*, IEEE Transactions on Automatic Control, 23 (1978), pp. 222–232.
- [10] G. COHEN, *Auxiliary problem principle and decomposition of optimization problems*, Journal of Optimization Theory and Applications, 32 (1980), pp. 277–305.
- [11] G. DAMBLIN, M. COUPLET, AND B. IOOSS, *Numerical studies of space-filling designs: optimization of Latin Hypercube Samples and subprojection properties*, Journal of Simulation, 7 (2013), pp. 276–289.
- [12] R. DEKKER, *Applications of maintenance optimization models: a review and analysis*, Reliability Engineering & System Safety, 51 (1996), pp. 229–240.
- [13] S.-H. DING AND S. KAMARUDDIN, *Maintenance policy optimization—literature review and directions*, The International Journal of Advanced Manufacturing Technology, 76 (2015), pp. 1263–1283.
- [14] A. FROGER, M. GENDREAU, J. E. MENDOZA, E. PINSON, AND L.-M. ROUSSEAU, *Maintenance Scheduling in the Electricity Industry: A Literature Review*, European Journal of Operational Research, 251 (2016), pp. 695–706.

- [15] A. GRIGORIEV, J. VAN DE KLUNDERT, AND F. C. SPIEKSMAN, *Modeling and solving the periodic maintenance problem*, European Journal of Operational Research, 172 (2006), pp. 783–797.
- [16] L. S. LASDON AND J. D. SCHOEFFLER, *A multi-level technique for optimization*, Joint Automatic Control Conference, 3 (1965), pp. 85–92.
- [17] R. LUSBY, L. F. MULLER, AND B. PETERSEN, *A solution approach based on Benders decomposition for the preventive maintenance scheduling problem of a stochastic large-scale energy system*, Journal of Scheduling, 16 (2013), pp. 605–628.
- [18] W. Q. MEEKER AND L. A. ESCOBAR, *Statistical methods for reliability data*, John Wiley & Sons, 2014.
- [19] M. D. MESAROVIĆ, D. MACKO, AND Y. TAKAHARA, *Theory of Hierarchical, Multilevel Systems*, vol. 68 of Mathematics in Science and Engineering, Academic Press, 1970.
- [20] R. P. NICOLAI AND R. DEKKER, *Optimal Maintenance of Multi-component Systems: A Review*, in Complex System Maintenance Handbook, Springer London, London, 2008, pp. 263–286.
- [21] M. C. OLDE KEIZER, S. D. P. FLAPPER, AND R. H. TEUNTER, *Condition-based maintenance policies for systems with multiple dependent components: A review*, European Journal of Operational Research, 261 (2017), pp. 405–420.
- [22] M. C. OLDE KEIZER, R. H. TEUNTER, AND J. VELDMAN, *Joint condition-based maintenance and inventory optimization for systems with multiple components*, European Journal of Operational Research, 257 (2017), pp. 209–222.
- [23] N. REZG, X. XIE, AND Y. MATI, *Joint optimization of preventive maintenance and inventory control in a production line using simulation*, International Journal of Production Research, 42 (2004), pp. 2029–2046.
- [24] O. ROUX, M. A. JAMALI, D. A. KADI, AND E. CHÂTELET, *Development of simulation and optimization platform to analyse maintenance policies performances for manufacturing systems*, International Journal of Computer Integrated Manufacturing, 21 (2008), pp. 407–414.
- [25] R. SARKER AND A. HAQUE, *Optimization of maintenance and spare provisioning policy using simulation*, Applied Mathematical Modelling, 24 (2000), pp. 751–760.
- [26] M. SÁNCHEZ-SILVA, D. M. FRANGOPOL, J. PADGETT, AND M. SOLIMAN, *Maintenance and Operation of Infrastructure Systems: Review*, Journal of Structural Engineering, 142 (2016), p. F4016004.
- [27] Y. TAKAHARA, *Multilevel Approach to Dynamic Optimization*, Tech. Report SRC-50- C-64-18, Case Western Reserve University, Systems Research Center, 1964.