

Timely condition-based maintenance planning for multi-component systems

Verbert, Kim; De Schutter, Bart; Babuska, Robert

DOI

[10.1016/j.res.2016.10.032](https://doi.org/10.1016/j.res.2016.10.032)

Publication date

2017

Document Version

Accepted author manuscript

Published in

Reliability Engineering & System Safety

Citation (APA)

Verbert, K., De Schutter, B., & Babuska, R. (2017). Timely condition-based maintenance planning for multi-component systems. *Reliability Engineering & System Safety*, 159, 310-321. <https://doi.org/10.1016/j.res.2016.10.032>

Important note

To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.

Timely Condition-Based Maintenance Planning for Multi-Component Systems

K. Verbert^{a,*}, B. De Schutter^a, R. Babuška^a

^a*Delft Center for Systems and Control (DCSC), Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands*

Abstract

Last-minute maintenance planning is often undesirable, as it may cause downtime during operational hours, may require rescheduling of other activities, and does not allow to optimize the management of spare parts, material, and personnel. In spite of the aforementioned drawbacks of last-minute planning, most existing methods plan maintenance activities at the last minute. In this paper, we propose a new strategy for timely maintenance planning in multi-component systems. As a first step, we determine for each system component independently the most appropriate maintenance planning strategy. This way, the maintenance decisions can be tailored to the specific situations. For example, conservative maintenance decisions can be taken when the risk tolerance is low, and maintenance decisions can be made timely when we can accurately predict future degradation behavior. In the second step, we optimize the maintenance plan at the system level. Here, we account for economic and structural dependence with the aim to profit from spreading or combining various maintenance activities. The applicability of the method is demonstrated on a railway case. It is shown how the different cost functions (e.g. costs of maintenance, downtime, and failure) influence the maintenance decisions.

Keywords: Condition-based maintenance; Optimization; Sequential decision making; Multi-component systems; Economic dependence; Structural dependence; Dynamic maintenance grouping; Railway networks.

1. Introduction

For many systems, like manufacturing and transportation systems, maintenance activities have a major influence on the availability, safety, and operational costs of the system. The ideal maintenance strategy prevents failures without resorting to over-maintenance. Such a strategy depends on the current and the future health of the system, which are never completely known in practice. However, if the right system variables are measured and processed adequately, good estimates and predictions of the system health can be obtained, based on which the maintenance can be planned. This is the motivation behind *condition-based maintenance*.

Although much research has been devoted to maintenance planning based on real-time condition monitoring, most existing methods use only diagnostic information, consider planning of individual components, and focus on last-minute planning. Such approaches are sufficient for systems for which it is convenient to perform maintenance shortly after the decision to do so has been made. Last-minute maintenance planning is however often undesirable as it may cause downtime during operational hours, may

require rescheduling of other activities, and does not allow to optimize the management of spare parts, material, and personnel. Furthermore, in multi-component systems, like road and railway networks and wind farms, it may be beneficial to combine or spread maintenance activities. This is not possible when maintenance needs are known just in time.

Motivated by the shortcomings of last-minute maintenance planning, we propose a two-stage bottom-up approach¹ for timely maintenance decision making based on real-time condition monitoring in multi-component systems. A bottom-up approach is preferred over a top-down (aggregate) approach, because of its applicability to heterogeneous systems, i.e. systems consisting of multiple types of components [1–4].

The *first stage* consists of determining the need for maintenance on each of the individual system components. If maintenance is required, based on the nature and urgency of the problem, the most appropriate type and time of maintenance have to be decided. As information regarding the system health is available in real time, it is not obvious when to settle on the decision regarding the time and type of maintenance. Generally, the more data are available, the better we can estimate the current and future system health, allowing a better decision on the time

*Corresponding author

Email addresses: k.a.j.verbert@tudelft.nl (K. Verbert),
b.deschutter@tudelft.nl (B. De Schutter),
r.babuska@tudelft.nl (R. Babuška)

¹In a bottom-up approach, optimal maintenance strategies for the individual system components are determined first.

and type of maintenance. However, it also holds that the more data are available, the further the system fault has already developed, reducing the freedom to efficiently plan the maintenance. This trade-off implies that the timing of the final maintenance decision is an important decision variable. To the authors' best knowledge, the timing of maintenance decisions, i.e. trading off accuracy and timeliness, has not been previously considered in the context of condition-based maintenance decision making. However, similar kinds of problems have been studied in economics as the *intertemporal choice problem* [5] and in probability theory as the *optimal stopping problem* [6, 7]. In this work, we extend the intertemporal choice problem to the maintenance domain.

In the *second stage*, we determine, based on the previously determined maintenance strategies for the individual components, the system-level maintenance strategy that minimizes total costs. So far, most work on system-level maintenance optimization has been focused on incorporating budget constraints [3, 8]. In this work, we incorporate economic and structural dependence with the aim to minimize costs by spreading or combining maintenance activities. As already pointed out in [2, 4] in the context of infrastructure management, the benefits of combining maintenance depend on the (spatial) relationships among the assets. Combining maintenance activities on multiple system components in road and railway networks is usually advantageous when the components form a series configuration, as the particular section is not available anyway. For system components arranged in a parallel structure, simultaneous maintenance is often undesirable, as it further reduces network capacity. An analogous reasoning holds for multi-component systems in general: in deciding whether to combine or spread maintenance, a trade-off between economics of scale and loss of functionality has to be made. The exact form of this trade-off depends on the structural dependencies between the considered components. In this work, we propose a systematic way for incorporating these dependencies in the system-level optimization.

In summary, the contributions of this paper are:

- We propose a component-level approach to maintenance planning in heterogeneous systems that trades timeliness for accuracy (Section 5);
- We propose a system-level maintenance optimization method that allows trading loss of functionality for cost reduction by incorporating economic and structural dependence in a fundamental way (Section 6).

To demonstrate both approaches, we apply them in a case study concerning maintenance planning for a railway network (Section 7) as a railway network is a typical example of a system subject to heterogeneity and interdependence for which last-minute maintenance planning is undesirable.

2. Literature review

Over the past years, various papers have been published on condition-based maintenance planning. Most of these works (e.g. [9–14]) base their maintenance decisions on just diagnostic information.

Currently, only a few papers have been published that consider prognostic information for maintenance optimization [15–19]. The advantage of including prognostic information is demonstrated in [18]. In [16], an optimal condition-based replacement policy is proposed for systems for which degradation conforms to an inverse Gaussian process. The authors of [15, 19] optimize maintenance planning for complex multi-component systems, assuming that component degradation follows a gamma distribution. The authors of [17] consider maintenance optimization for systems the degradation behavior of which is described by a random coefficient auto-regressive model.

Two drawbacks of the aforementioned methods are: 1. they only address the question whether or not to perform maintenance at a particular decision time instant; 2. except for [18], they rely on a specific degradation model. In contrast, the method proposed in this work is not restricted to a specific degradation model, but uses the diagnosis and prognosis result to predict both the *required time* and *optimal time* of maintenance.

Maintenance planning for *multi-component systems* has been treated e.g. in [15, 18–24]. Because in general interactions² exist among system components, the optimal maintenance strategy for a multi-component system is not simply the set of optimal component-level solutions [22]. Consequently, recently methods have been proposed that account for (some of) these interactions, most of them focusing on economic dependence [20–22, 24]. Stochastic dependence is e.g. considered in [18], and the authors of [19] propose a method that incorporates both economic, stochastic, and structural dependence. In this work, we incorporate both economic and structural dependence. Like in [19], we model structural dependence as economic dependence. We do not explicitly consider stochastic dependence as we assume that this type of dependence is accounted for in the diagnosis and prognosis step. The main difference compared to the method proposed in [19] is that in [19] the maintenance schedule is updated each time new information becomes available without considering when to finalize the maintenance decision. In contrast, our method optimizes the time instant at which to

²A distinction can be made between economic, structural, and stochastic dependence. *Economic dependence* implies that maintenance costs decrease or increase when components are jointly maintained instead of separately. *Structural dependence* applies if components structurally form a part, so that maintenance of a failed component implies decommissioning of the other components as well. *Stochastic dependence* occurs if the state of a component influences the lifetime distribution of another component, or if there are causes outside the system that correlate the lifetimes of the components (common-cause failures) [25].

settle on the maintenance decision. We aim to settle on the maintenance decision in an early stage, so as to avoid last minute re-scheduling of other activities and to allow for the optimization of spare parts, material, and personnel. Especially in intensively-used systems, like railway networks, it is desirable to plan the maintenance in time. Moreover, in contrast to [19], we account for different types of maintenance activities.

Although the two-stage bottom-up approach that we consider for maintenance optimization is based on the approach proposed in [3], the two strategies are clearly different: while [3] focuses on infrastructure management, we use it for condition-based maintenance planning, resulting in other objective functions. Moreover, [3] only accounts for budget constraints in the system-level optimization, while we include economic and structural dependence in the system-level optimization.

In summary, compared to existing methods, the proposed method adds the following:

1. Next to deciding whether maintenance is needed, we optimize the required type of maintenance and the time to perform the maintenance;
2. We optimize the time to settle on the aforementioned maintenance decisions, hereby trading accuracy for timeliness;
3. We decouple the maintenance optimization from the diagnosis and prognosis process. This way, we are able to exploit both diagnostic and prognostic information for maintenance optimization without restricting ourselves to a particular degradation model;
4. We provide a systematic framework for incorporating economic and structural dependencies among system components in the system-level optimization.

3. Assumptions

For clarity and to limit the scope of this paper, the following assumptions are adopted:

- A₁ We have fixed monitoring time instants at which diagnosis, prognosis, and maintenance planning are carried out.
- A₂ Only major maintenance is considered. After maintenance is carried out, the component is in an as-good-as new state.
- A₃ The different, application-specific, cost functions (e.g. costs of maintenance, downtime, and system failure) are assumed to be known.
- A₄ The costs of performing a maintenance action are independent of the actual system health.
- A₅ Only economic dependence is explicitly modeled. Structural dependence is modeled as economic dependence. In addition, we assume that stochastic dependence is accounted for in the diagnosis and prognosis process.

Moreover, as the focus of this paper is on maintenance optimization, we assume that adequate diagnosis and prognosis results are available. State-of-the art diagnosis and prognosis methods can e.g. be found in [26–32]. Below, we specify how we assume the diagnosis and prognosis result to be specified.

3.1. Diagnosis result

The health state of component i at time τ is captured by a discrete variable $H_i(\tau) \in \{h, f_{i,1}, f_{i,2}, \dots, f_{i,\ell_i}\}$, where h represents the healthy state and $f_{i,1}$ through f_{i,ℓ_i} denote the possible fault types for component i . Since it is generally not possible to determine the health state with complete certainty, the diagnosis result is a probability mass function over the current ($\tau = \tau_c$) health state: $P(H_i(\tau_c))$.

3.2. Prognosis result

The prognosis result (see Figure 1) of component i includes the current value of the degradation measure³, $d_i(\tau_c)$, as well as its predicted evolution. We assume $d_i(\cdot)$ to be a continuous-time stochastic process. Since the prognosis result is a probability distribution over this process, we denote it as $p(d_i|\mathcal{D}_c)$, where \mathcal{D}_c represents all data available at time τ_c . The prognosis result can be used to predict the value $p(d_i(\tau)|\mathcal{D}_c)$ of the degradation measure at a given time $\tau > \tau_c$.

Since different fault types generally result in different time behaviors of $d_i(\cdot)$, we require a distinct prognostic model to be available for every possible fault type $f_{i,j}, \dots, f_{i,\ell_i}$. In the sequel, we assume that the prognostic models are captured by parametric models (see e.g. [28, 33]). So, for each component i we have ℓ_i parameterized models: $m_{i,1}(\cdot|\Theta_{i,1}), \dots, m_{i,\ell_i}(\cdot|\Theta_{i,\ell_i})$, each characterizing the expected temporal behavior of the degradation measure as a consequence of the corresponding fault type. Vector $\Theta_{i,j}$ denotes the model parameters, which might be stochastic.

The applicable parametric model is indicated by the diagnosis result, and its parameters are estimated based on the available data. Since the diagnosis result is a probability mass function over the possible health states, the predicted degradation process constructed at time τ_c evaluated at τ is a mixture of all applicable degradation models:

$$d_i(\tau) \approx \sum_{j=1}^{\ell_i} P(H_i(\tau_c) = f_{i,j}) \cdot m_{i,j}(\tau|\Theta_{i,j})$$

Based on this predicted degradation process and the failure threshold λ_i , which is the value of $d_i(\cdot)$ above which failure occurs, the Remaining Useful Life (RUL) distribution can be determined (see e.g. [33, 34] for more information on determining RUL distributions).

³The degradation measure is a continuous variable that can be computed from sensor information, and captures a component's degree of degradation [33].

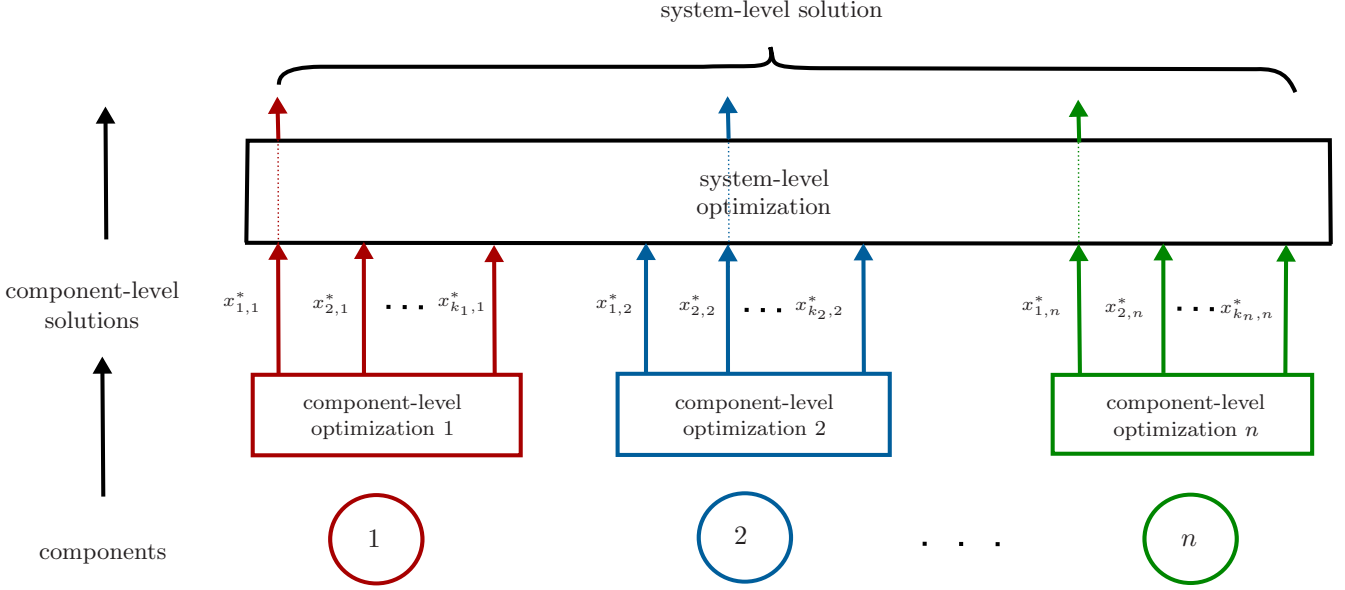


Figure 2: Two-stage bottom-up approach (adapted from [3]). First stage: optimal and near-optimal maintenance strategies $x_{1,i}$ till $x_{k_i,i}$ are determined for each component i in need of maintenance. Second stage: the optimal combination of component-level maintenance strategies is determined at the system level.

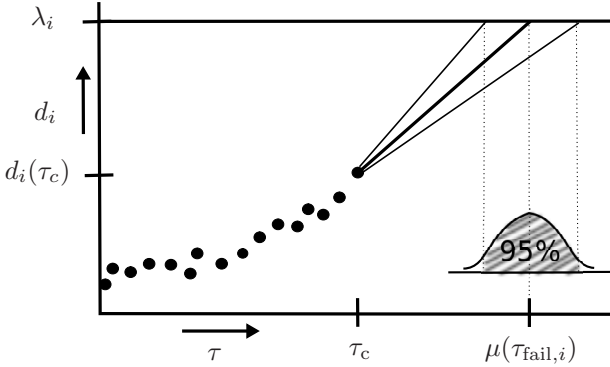


Figure 1: Illustration of the prognosis result at $\tau = \tau_c$. For $\tau \leq \tau_c$, the degradation measure $d_i(\tau)$ is known. For $\tau > \tau_c$ we predict the behavior of the degradation measure based on its previous values and the fault-specific parametric model (linear here). From this prediction, the distribution of the estimated failure time, $p(\tau_{\text{fail}})$, can be determined.

4. Problem definition

In this work, condition-based maintenance planning for a heterogeneous system consisting of n independently and continuously monitored components is considered. In defining the problem, we make a distinction between decision making at the component level and decision making at the system level (see Figure 2).

4.1. Component-level optimization

At this level, we aim to find a set of optimal and near-optimal maintenance strategies for each system component in need of maintenance. The optimal maintenance strategy is defined by:

1. the required type of maintenance;
2. the optimal time of performing maintenance.

The *required type of maintenance* refers to the maintenance action that brings the system to an as-good-as-new condition. As only major maintenance is considered, the optimal maintenance action depends only on the system health state H_i .

The *optimal time of performing maintenance* refers to the maintenance time that minimizes total “costs”, i.e. the time of maintenance is chosen such that the component’s lifetime is maximized, while accounting for the following additional objectives:

- prevention of failure;
- minimization of downtime during operational hours.

The optimal time depends on the expected degradation over time and on the applicable cost functions and risk tolerances.

As information regarding the system health is available in real time, it is not obvious when to settle on a decision regarding the time and type of maintenance. On the one hand, it is desirable to make the maintenance decisions as early as possible, as the creation of an effective maintenance schedule requires that the maintenance needs are known in time, e.g.:

- To prevent a failure, the right decision should be made at least z_i time units before component i fails, where z_i refers to the time needed to get the personnel and material at the maintenance location and to maintain component i .
- Moreover, to prevent system downtime during operational hours, the right decision should be made at least $z_i + q_i$ time units before component i fails, with q_i the maximum possible time gap between any two consecutive out-of-service periods of component i . Indeed, when the system will fail just before an out-of-service period, maintenance has to be performed in a previous out-of-service period to avoid failure and system downtime during operational hours.
- To optimize (system-level) maintenance planning: The earlier the maintenance requirements are known, the more freedom there is in maintenance scheduling, and the more cost efficient the resulting maintenance schedule will be.

On the other hand, it is desirable to plan the maintenance based on reliable and accurate predictions of the system health. This is done to avoid that maintenance is performed too late, resulting in sudden failures, or that maintenance is done too early, leading to over-maintenance. Since the prediction accuracy increases over time, a trade-off between accuracy and timeliness needs to be made. Clearly, early predictions that are not accurate at all are of no use. Perfect predictions less than z_i time units before a functional failure of component i are of no use either. In such a case, a failure could not be prevented.

In summary, next to determining the required type of maintenance and optimal time to perform the maintenance, the component-level optimization comprises the timing of the maintenance decision, i.e. trading off accuracy and timeliness.

4.2. System-level optimization

At this level, we search for the optimal maintenance strategy from the system-level point of view. In the simplest case, the system-level solution coincides with the set containing the optimal strategy for each component. However, if budget or resource constraints are binding or system dependence applies, the system-level solution may differ from the set of optimal component-level solutions.

As only major maintenance is considered, the only optimization variable at the system level is the time of performing maintenance. At the system-level, we aim to max-

imizing the cost benefits resulting from combining or spreading maintenance activities. In general, direct maintenance costs can be reduced by combining maintenance on nearby system components. This way, part of the costs, e.g. setup work and transportation costs, can be shared between the simultaneously maintained components. Although the direct maintenance costs generally decrease when maintenance on nearby components is combined, the indirect costs (e.g. costs related to downtime) do not necessarily decrease. The effect of combining maintenance on the indirect costs depends on the extent to which the additional maintenance of a component influences the functionality of the whole system. When the whole system is out of service during the maintenance of component A , simultaneously maintaining an arbitrary component B has no negative impact on the functionality of the system, and so on the indirect costs. However, when the system is still (partly) functional when only component A is maintained, but no longer (or less) functional when component A and B are maintained simultaneously, combining maintenance on components A and B may have a negative effect on the indirect maintenance costs. In this case, the potential reduction in direct costs (economics of scale) must be traded against the potential increase of indirect costs (loss of functionality). To handle this trade-off, the influences of combining maintenance on both the direct costs and the indirect costs need to be clear. Here, we assume that the potential reduction in direct costs depends on:

1. the number of simultaneously maintained components;
2. the similarity between the maintenance activities;

and the potential reduction in indirect costs depends on:

1. the reduction of downtime when combining maintenance;
2. the structural dependencies between the components.

5. Decision making at the component level

At the component-level, for each component i in need of maintenance, the optimal and near-optimal⁴ maintenance strategies are determined. The focus is on the individual components and dependencies among system components are not yet taken into account. The determination of the (near-)optimal maintenance strategies is done in two steps (see Figure 3). At each monitoring time instant, we first determine, based on the currently available information, optimal and near optimal maintenance strategies. Next, we determine whether we want to plan the maintenance according to the determined maintenance strategy or to postpone the maintenance decision to a later time, when more data are available.

⁴The near-optimal strategies $x_{2,i}^*, \dots, x_{k_i,i}^*$ serve as an input for the system-level maintenance optimization (see Figure 2).

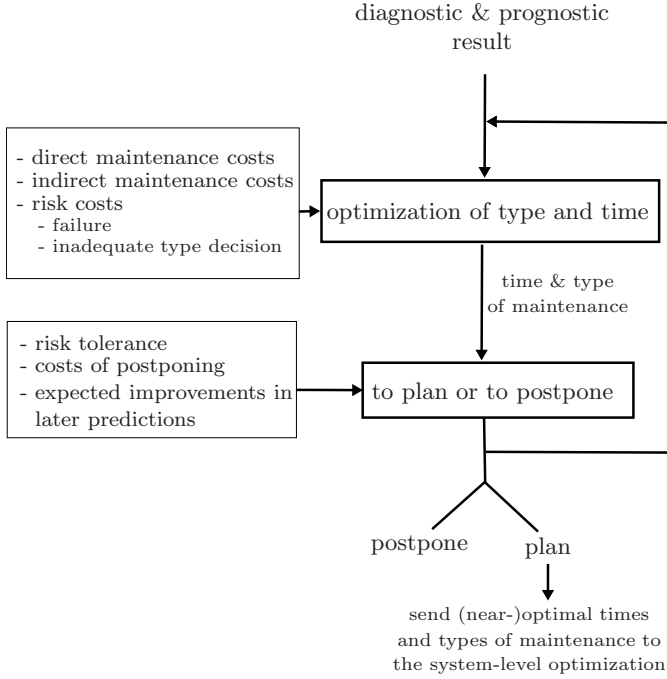


Figure 3: Component-level maintenance optimization. First, based on the currently available information, the optimal type and time of maintenance are determined. Next, it is decided whether to plan the maintenance according to the previously determined strategy or to postpone the decision to a later time (when more data are available).

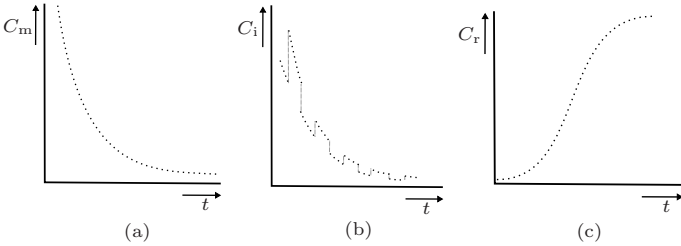


Figure 4: Example relations between the different cost components and the time of maintenance. (a) life-time averaged direct maintenance costs; (b) lifetime-averaged indirect maintenance costs; (c) costs related to risk.

For the sake of brevity, in the sequel we omit the subscript i when the explicit reference to a particular component i is not necessary. For the same reason, we omit the time argument τ whenever possible.

5.1. Optimization of maintenance type and time

In this step we search for the optimal maintenance action a^* , maintenance time t^* , and the associated costs C^* :

$$(a^*, t^*) = \arg \min_{a \in A, t \in T} C_m(a, t) + C_i(a, t) + C_r(a, t) \quad (1)$$

$$C^* = \min_{a \in A, t \in T} C_m(a, t) + C_i(a, t) + C_r(a, t) \quad (2)$$

with A the discrete set of possible maintenance activities, T the discrete set of available maintenance time instants, $C_m(a, t)$ the lifetime-averaged direct costs of performing maintenance activity a at time t , $C_i(a, t)$ the lifetime-averaged indirect costs of maintenance activity a at time

t , and $C_r(a, t)$ the costs associated with the risk of action a being inadequate or time t being too late, where all costs are expressed in terms of (virtual) monetary units. More specifically, we define the lifetime-averaged direct costs of maintenance $C_m(a, t)$ as:

$$C_m(a, t) = \frac{c_m(a)}{t - t_{\text{mnt}}} \quad (3)$$

with t_{mnt} the previous maintenance time and $c_m(a)$ the direct costs of performing maintenance activity a (e.g. material, personnel). So, the lifetime-average direct costs of maintenance $C_m(a, t)$ correspond to the costs $c_m(a)$ averaged over the lifetime $t - t_{\text{mnt}}$; the larger t is chosen, the lower are the lifetime-averaged direct costs of maintenance (see Figure 4(a)).

The lifetime-averaged indirect costs of maintenance $C_i(a, t)$ are defined as:

$$C_i(a, t) = \frac{c_i(a, t)}{t - t_{\text{mnt}}} \quad (4)$$

with $c_i(a, t)$ the indirect costs (e.g. the costs related to downtime) of maintenance activity a at time t . In contrast to the direct costs, the indirect costs depend on the time of maintenance. Indeed, for most systems, the cost of downtime depends on the time of the downtime. For example, for road or railway networks, the inconvenience of downtime is less during night than during day. So, given a particular action a , the life-time averaged indirect costs $C_i(a, t)$ intermittently decrease for increasing t (see Figure 4(b)).

Finally, we define the costs $C_r(a, t)$ related to risk as:

$$C_r(a, t) = \sum_{j=1}^{\ell} P(H = f_j) p(\tau_{\text{fail}} < t | H = f_j) C_{\text{fail}, j} + \sum_{j \in \{1, \dots, \ell\}} P(H = f_j) C_{\text{wrong}, j}(a) \quad (5)$$

The first term expresses the costs related to the risk of maintenance time t being too late to avoid a failure. The second term expresses the costs related to the risk of maintenance action a being not appropriate to repair the system. In (5), $C_{\text{fail}, j}$ represents the additional costs of a failure as a consequence of fault f_j and $C_{\text{wrong}, j}(a)$ represents the cost of an (inadequate) maintenance type decision a in case of fault f_j . So, given a particular maintenance action, and as long as no maintenance is done, the costs related to risk $C_r(a, t)$ increase for increasing t (see Figure 4(c)).

We conclude that the first two terms in (1), i.e. C_m and C_i , are minimized for t chosen as large as possible, while the last term, C_r , is minimized for t chosen as small as possible. The overall optimum depends, besides on the diagnosis result ($P(H = f_j)$) and prognosis result ($p(\tau_{\text{fail}} < t | H = f_j)$), on the cost functions $c_m(\cdot)$, $c_i(\cdot)$, $C_{\text{fail}, j}$, and $C_{\text{wrong}, j}(\cdot)$. These cost functions are application-specific and need to be defined by practitioners. The direct maintenance costs $c_m(a)$ are generally known. The other cost

functions have to be defined such that they reflect the relative importance of costs minimization, downtime minimization, and failure avoidance.

Besides the optimal maintenance strategy, alternative, near optimal, strategies are determined. These alternative strategies are required for the system-level optimization and can be found by excluding the optimal strategy from the search space [3]:

$$(a_{k+1}^*, t_{k+1}^*) = \arg \min_{(a,t) \in A \times T \setminus \{(a_l^*, t_l^*) | l=1, \dots, k\}} C_m(a, t) + C_i(a, t) + C_r(a, t) \quad (6)$$

$$C_{k+1}^* = \min_{(a,t) \in A \times T \setminus \{(a_l^*, t_l^*) | l=1, \dots, k\}} C_m(a, t) + C_i(a, t) + C_r(a, t) \quad (7)$$

The resulting optimization problem can be solved using standard optimization techniques (see e.g. [35]). Which of the available algorithms is the most suitable depends e.g. on the specific application, the size of the system, the number of possible maintenance activities a and maintenance times t , the frequency at which diagnosis, prognosis, and maintenance planning are carried out, and the number of required near-optimal solutions. A further elaboration on the selection of an optimization approach is beyond the scope of this paper.

5.2. To plan or to postpone

In this step, it is decided whether to accept the previously found component-level maintenance strategy (i.e. the combination of maintenance time t^* and type a^*) or to wait for a potential better maintenance strategy (i.e. a strategy with lower costs C^*) at a later time. So we have a trade-off between potential cost savings and the risk and inconvenience of postponing maintenance planning. The problem can thus be considered as a sequential decision problem. At each monitoring instant, based on the outcome of the previously determined optimal maintenance strategy, it has to be decided whether to plan maintenance or to postpone the maintenance decision. If the decision is postponed, we face the decision again at the next monitoring time instant. The problem repeats itself until the maintenance is planned or the system fails, in which case corrective maintenance is needed.

Inspired by the approaches proposed in [6, 36, 37], we propose to solve the sequential decision problem by formulating it as a Markov decision process and use dynamic programming or reinforcement learning to solve the obtained Markov decision problem. The choice to take advantage of the methods proposed in [6, 36, 37] is motivated by the resemblance between the considered maintenance optimization task and the task of buying durable goods or airline tickets: First, the risk that an airline ticket is sold out before buying is comparable to the risk of failure before maintenance. Second, both buying durable goods and

maintenance planning are associated with a cost for postponing: when buying durable goods, postponing means that you cannot immediately use the product; when planning maintenance, postponing means that less freedom remains in scheduling. Third, for both buying durable goods or airline tickets and planning maintenance, there is uncertainty about the future ‘‘costs’’, while having some (qualitative) knowledge about their further evolution.

5.2.1. Background on Markov decision processes

A (finite) Markov decision process is defined by the tuple (S, A_m, P, R) , where S represents a finite set of states, A_m a finite set of actions, P a probabilistic transition function, and R a reward function [38, 39]. The transition function indicates how the state changes as a result of action a_m . Assuming a probabilistic setting, the transition function $P(a_m, s, s')$ outputs the probability that the system moves to state s' given that it is currently in state s and action a_m is taken. The reward function $R(a_m, s, s')$ evaluates the immediate effect of moving from state s to state s' under action a_m . Note that both the transition function and the reward function satisfy the Markov property. The goal is to design an optimal policy π that defines which action $a_m \in A_m$ to take when the system is in state $s \in S$ such that the expected long-term reward is maximized. When the reward function R and the state transition function P are known, the problem can be solved by dynamic programming [40] by recursively solving the following equations:

$$\pi(s) = \arg \max_{a_m} \left(\sum_{s'} P(a_m, s, s') (R(a_m, s, s') + \gamma V(s')) \right) \quad (8)$$

$$V(s) = \sum_{s'} P(\pi(s), s, s') \left(R(\pi(s), s, s') + \gamma V(s') \right) \quad (9)$$

with $\gamma \in (0, 1]$ a discount factor. When the reward function or the transition probabilities are unknown, reinforcement learning [41] can be used to learn the optimal policy, which can be done offline or online. Online learning is not recommended for maintenance decision making, as we want to avoid certain (e.g. safety-critical) errors in the maintenance decisions. Advantages of learning in sequential decision making are that it relieves the designer of the system from deciding upon everything in the design phase and that it can cope with uncertainty and changing situations [41].

The final choice for dynamic programming or learning is problem-specific and depends on the available domain knowledge and data.

5.2.2. Reformulation as a Markov decision process

To reformulate the considered decision task, i.e. deciding whether to plan maintenance or to postpone the maintenance decision, as a Markov decision problem, we first define the set of states S as the set of all possible instances

of the following state vector:

$$s = [C^* \ P(H) \ \mu(\Theta) \ \sigma(\Theta) \ \tau - \tau_0 \ D \ F]^\top \quad (10)$$

with C^* the costs associated with the maintenance strategy under consideration (see Section 5.1), $P(\cdot)$ the probability mass function over the health state, $\mu(\Theta)$ and $\sigma(\Theta)$ the mean and standard deviation of the parameter vector $\Theta = [\Theta_1, \dots, \Theta_\ell]$ of the degradation model, τ_0 the initial time of the decision, D a binary variable indicating whether maintenance is already planned or not, and F a binary variable indicating whether the system fails. The state vector is chosen this way as the decision whether or not to plan depends on the current costs C^* and the expected costs of future strategies, which can be predicted based on C^* , $P(H)$, and the estimated time-to-failure distribution (characterized by $[\mu(\Theta_1), \dots, \mu(\Theta_\ell)]$ and $[\sigma(\Theta_1), \dots, \sigma(\Theta_\ell)]$). Since timely maintenance planning is preferred over last-minute planning, the benefits of planning decrease over time. This motivates to include a decreasing term in the immediate reward function for planning. To ensure that the Markov property is satisfied, $\tau - \tau_0$ is included in the state. The last two elements, D and F , are included to define the terminal states. When one of the two is true, the decision process ends, i.e. all states for which holds that $D = 1$ or $F = 1$ are terminal states.

The set of actions A_m is:

$$A_m = \{\text{plan, postpone}\} \quad (11)$$

Next, we define the transition probabilities as follows:

$$P(C^*(\tau + \Delta)) = f_C(C^*(\tau), P(H(\tau)), \mu(\Theta, \tau), \sigma(\Theta, \tau)) \quad (12)$$

$$P(H(\tau + \Delta)) = P(H(\tau)) \quad (13)$$

$$\mu(\Theta, \tau + \Delta) = \mu(\Theta, \tau) \quad (14)$$

$$\sigma(\Theta, \tau + \Delta) = f_\sigma(P(H(\tau)), \mu(\Theta, \tau), \sigma(\Theta, \tau)) \quad (15)$$

$$D(\tau + \Delta) = \begin{cases} 0 & \text{if } a_m = \text{postpone} \\ 1 & \text{otherwise} \end{cases} \quad (16)$$

$$P(F(\tau + \Delta) = 1) = P((\tau_{\text{fail}} - \tau) < \Delta) \quad (17)$$

with Δ the time interval between two monitoring instants. Equation (12) predicts the costs at the next time instant given the current costs, the diagnosis outcome, and the prognosis result. The function $f_C(\cdot)$, which specifies the relation between $C^*(\tau + \Delta)$ and $C^*(\tau)$, $P(H(\tau))$, $\mu(\Theta, \tau)$, and $\sigma(\Theta, \tau)$, is application-specific and needs to be learned from historical data, possibly in combination with expert knowledge. Note that an important condition is that the diagnosis, prognosis, and maintenance optimization method are in use when collecting the data. Equation (13) indicates that the prediction of the diagnosis result at time $\tau + \Delta$ equals the diagnosis result at time τ . Similarly, (14) specifies that the prediction of the mean of parameter vector Θ at time $\tau + \Delta$ equal the mean at time τ . The variance

of the parameter vector at time $\tau + \Delta$ is expressed as a (decreasing) function of $P(H(\tau))$, $\mu(\Theta, \tau)$, and $\sigma(\Theta, \tau)$. Like $f_C(\cdot)$, the function $f_\sigma(\cdot)$ is application-specific and needs to be defined by domain experts or learned from data. Equation (16) specifies that maintenance is not planned as long as $a_m = \text{“postpone”}$. Finally, (17) specifies the probability that the system fails at the next monitoring instant.

With respect to the reward function, we model the loss of utility due to waiting by means of discounting [6]. If we decide to plan the maintenance at time τ , this gives us a utility of $u_{\text{max}}\delta^{\tau-\tau_0}$, with τ_0 the initial time maintenance planning is considered, δ the discounting rate, and u_{max} the maximum utility for planning. The net reward for planning maintenance equals the obtained utility minus the costs:

$$R(a_m = \text{plan}, s_D = 0, s'_D = 1) = u_{\text{max}}\delta^{\tau-\tau_0} - C^*$$

with s_D the D -component of the state (10). The immediate reward associated with postponing the decision is zero as long as the system does not fail in the next time step. When the system fails, this is penalized with a negative reward $-\alpha$. All together, we define our reward function as follows:

$$R(a_m, s, s') = \begin{cases} u_{\text{max}}\delta^{\tau-\tau_0} - C^* & \text{for } s'_D = 1 \\ -\alpha & \text{for } s'_D = 0 \text{ and } s'_F = 1 \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

with s_F the F -component of the state (10). Note that we propose the use of a finite decision horizon, where the horizon corresponds to the predicted failure time $\mu(\tau_{\text{fail}})$ minus $(w\sigma(\tau_{\text{fail}}) + z)$, with w an application-specific and user-defined parameter defining the latest time one wants to schedule maintenance. The larger w , the more failure risk avoidant one is. Planning is forced when the end of the horizon is reached. Because we consider a finite horizon, we do not discount future rewards, i.e. $\gamma = 1$.

6. System-level maintenance optimization

In the system-level optimization we search for the optimal system-level maintenance strategy accounting for economic and structural dependencies among system components.

6.1. Problem formulation

As we consider maintenance planning based on real-time condition monitoring, we face both newly entered maintenance needs and already scheduled⁵ (but not yet

⁵The order of scheduling at the system-level is determined by the order in which the maintenance needs are set at the component level.

carried out) maintenance activities in the system-level optimization. Consider that η components need to be maintained and that for η_{ns} of them maintenance has not yet been planned. Without loss of generality they are renumbered such that $l = 1, \dots, \eta_{\text{ns}}$ represent the components for which maintenance is not yet planned. For these components, we have to select the optimal system-level strategy x_l from the set A_l^* of optimal and near-optimal component-level maintenance strategies, with $A_l^* = \{x_{1,l}^*, x_{2,l}^*, \dots\} = \{(a_{1,l}^*, t_{1,l}^*), (a_{2,l}^*, t_{2,l}^*), \dots\}$ (see Section 5.1). For each other component $l = \eta_{\text{ns}} + 1, \dots, \eta$, the maintenance time and type are already defined, i.e. $x_l = x_{o,l}^*$. Now the aim is to find the optimal system-level maintenance strategy X_{free}^* , i.e. a strategy $x_l \in A_l^*$ for each component $l = 1, \dots, \eta_{\text{ns}}$ such that the system-level criterion $C_{\text{SL}}(X_{\text{free}}, X_{\text{fixed}})$ is minimized:

$$X_{\text{free}}^* = \arg \min_{X_{\text{free}}} C_{\text{SL}}(\underbrace{X_{\text{free}}, X_{\text{fixed}}}_X) \quad (19)$$

with

$$\begin{aligned} X_{\text{free}} &= (x_1, \dots, x_{\eta_{\text{ns}}}) \text{ with } x_l \in A_l^* \\ X_{\text{fixed}} &= (x_{\eta_{\text{ns}}+1}, \dots, x_\eta) \text{ with } x_l = x_{o,l}^* \end{aligned}$$

6.2. Optimization criterion

The optimization criterion $C_{\text{SL}}(X)$ expresses the total costs of system-level strategy X . The total costs equal the sum of the individual maintenance costs corrected for the cost benefits/drawbacks obtained from combining or spreading maintenance.

6.2.1. Cost of individual maintenance activities

The first part of the optimization criterion expresses the total costs of system-level strategy X in the absence of system dependence, i.e.:

$$C_0(X) = \sum_{l=1}^{\eta} C_{x_l}, \quad x_l \in X \quad (20)$$

with C_{x_l} the component-level costs of maintenance strategy x_l .

6.2.2. System dependence

We make a distinction between:

1. economics of scale;
2. loss of utility.

To incorporate economics of scale, we split the direct costs of maintenance $c_m(a)$ into three components:

$$c_m(a) = c_{m,1}(a) + c_{m,2}(a) + c_{m,3} \quad (21)$$

with $c_{m,1}(a)$ the fixed costs that do not depend on economics of scale, $c_{m,2}(a)$ the costs that can be shared between components that simultaneously undergo maintenance action a , and $c_{m,3}$ the costs that can be shared between all simultaneously maintained components, regardless of the type of maintenance. This way, the costs savings resulting from economics of scale can be computed as:

$$C_{\text{EOS}}(X) = \sum_{t \in T} \sum_{a \in A} \beta_{1,(a,t)}(X) c_{m,2}(a) + \sum_{t \in T} \beta_{2,t}(X) c_{m,3} \quad (22)$$

with:

$$\beta_{1,(a,t)}(X) = \max(0, n_{1,(a,t)}(X) - 1)$$

$$\beta_{2,t}(X) = \max(0, n_{2,t}(X) - 1)$$

$n_{1,(a,t)}(X)$: number of components that undergo action a at time t under strategy X

$n_{2,t}(X)$: number of maintained components at time t under strategy X

To incorporate loss of utility, we consider both the reduction in downtime and the reduction in system functionality as a consequence of combining. To assess the costs saving resulting from downtime reduction, we split the indirect maintenance costs $c_i(a, t)$ into two parts:

$$c_i(a, t) = c_{i,1}(t) + c_{i,2}(a, t) \quad (23)$$

with $c_{i,1}(t)$ the costs that are directly related to system downtime and $c_{i,2}(a, t)$ all other indirect costs. The reduction in downtime when $n_{2,t} \geq 1$ components are maintained at time t is $(n_{2,t} - 1)c_{i,1}(t)$. Accordingly we define the reduction in downtime costs of strategy X as:

$$C_{\text{DT}}(X) = \sum_{t \in T} \beta_{2,t}(X) c_{i,1}(t) \quad (24)$$

To incorporate the additional loss of functionality when multiple components are maintained simultaneously, we divide the system's components into ν groups g_1, \dots, g_ν , such that if only components of one group are maintained at time t , no additional loss of functionality is induced. When components of different groups are maintained simultaneously, the functionality of the system reduces, and an additional costs term $C_{\text{LF}}(X)$ is added to the optimization criterion, with $C_{\text{LF}}(X)$ defined as:

$$C_{\text{LF}}(X) = \sum_{t \in T} f_{\text{LF}}(\mathcal{X}_t(X))$$

with $\mathcal{X}_t(X)$ the set containing all groups g_ζ of which a component is maintained at time t under strategy X , and $f_{\text{LF}}(\cdot)$ a function assigning a penalty cost to each possible set \mathcal{X}_t .

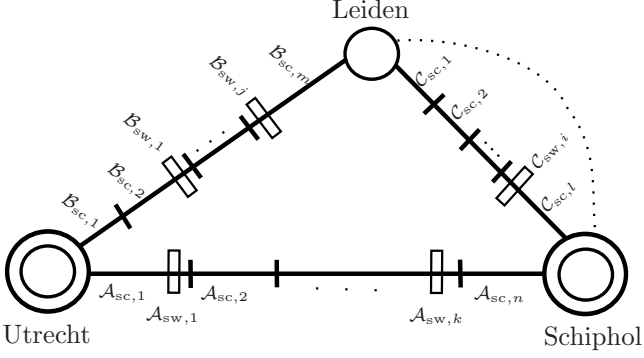


Figure 5: Railway network considered in the case study.

6.3. System-level optimization formulation

Taking all together, the system-level optimization criterion C_{SL} is defined as follows:

$$C_{SL}(X) = C_0(X) - C_{EOS}(X) - C_{DT}(X) + C_{LF}(X) \quad (25)$$

Together with (19) this defines the system-level optimization problem.

System-level optimization problems of small size can be solved by brute-force approaches. For computationally demanding problems, approximate algorithms, like pattern search heuristics or evolutionary algorithms can be used [3].

7. Case study

We illustrate the proposed approach on a case study concerning maintenance planning for a railway network. A railway network consists of different types of components (e.g., tracks, switches, bridges) that are located in areas with different environmental conditions, meaning that the system should be considered as heterogeneous with respect to deterioration processes and costs. Furthermore, railway networks are subject to economic and structural dependencies, meaning that costs and downtime can be reduced when maintenance activities are combined or spread in time. Note that this case study has an illustrative purpose. A full evaluation is only possible in combination with a diagnosis and prognosis method, which is beyond the scope of this paper. As the considered configuration is not so computationally demanding, an exhaustive search algorithm is used to solve the associated optimization problem.

7.1. Problem specification

7.1.1. System description

Consider the network depicted in Figure 5, representing a part of the Dutch railway network. Utrecht and Schiphol are two important and busy railway stations in the Netherlands. Besides the direct line between the two cities, there is an indirect connection between the two cities via Leiden. Furthermore, there is a bus connection between Leiden and Schiphol.

Assume that the railway network consists of two types of components:

1. sections (parts of the track);
2. switches.

A section can suffer from two types of faults: rail defects f_{rd} and rail contamination f_{rc} . A switch can suffer from one fault f_{sw} . Accordingly we have three maintenance actions: a_{rd} to repair a rail defect, a_{rc} to remove rail contamination, and a_{sw} to repair a switch. The expected temporal behavior of the degradation measure $d_i(\cdot)$ as a consequence of each fault is described by the following parametric models:

$$m_{i,rd}(\tau|\Theta_{i,rd}) = \theta_{i,1} + \theta_{i,2}e^{\theta_{i,3}(\tau-\tau_c)} \quad (26)$$

$$m_{i,rc}(\tau|\Theta_{i,rc}) = \theta_{i,4} + \theta_{i,5}(\tau - \tau_c) \quad (27)$$

$$m_{i,sw}(\tau|\Theta_{i,sw}) = \theta_{i,6} + \theta_{i,7}(\tau - \tau_c) \quad (28)$$

with $\theta_{i,3}$, $\theta_{i,5}$, and $\theta_{i,7}$ normally distributed random variables and the other parameters deterministic. For both sections and switches the failure threshold is set to 100.

7.1.2. Component-level cost functions

Type and time of maintenance. We define the cost of failure $C_{fail,j}$ as follows: For section faults, f_{rd} and f_{rc} , we make a distinction depending on the location of the section. Sections close to switches or level crossings influence the proper functioning of these switches and crossings. Therefore, a failure of such a section is more disastrous than a failure of another section. Sections that influence the proper functioning of other assets are said to be of type *II*, all other sections are of type *I*. Accordingly, the costs of a failure $C_{fail,j}$ are defined as:

$$C_{fail,f_{rd}} = \begin{cases} 500 & \text{if section is of type I} \\ 2000 & \text{if section is of type II} \end{cases} \quad (29)$$

$$C_{fail,f_{rc}} = \begin{cases} 750 & \text{if section is of type I} \\ 2500 & \text{if section is of type II} \end{cases} \quad (30)$$

$$C_{fail,f_{sw}} = 1000 \quad (31)$$

Let the costs of a wrong maintenance decision $C_{wrong,j}(a)$ be given by:

$$C_{wrong,f_{rc}}(a_{rd}) = 350 \quad (32)$$

$$C_{wrong,f_{rd}}(a_{rc}) = 500 \quad (33)$$

Note that we know the type of monitored component (section or switch). Hence, we will not schedule a section maintenance action (a_{rd} or a_{rc}) for a switch. Vice versa, we will not schedule a switch maintenance action (a_{sw}) for a section.

Let the different components of the direct maintenance

costs $c_m(a)$ be given by:

$$c_{m,1}(a) = \begin{cases} 87.5 & \text{if } a = a_{rd} \\ 72.5 & \text{if } a = a_{rc} \\ 130 & \text{if } a = a_{sw} \end{cases} \quad (34)$$

$$c_{m,2}(a) = \begin{cases} 7.5 & \text{if } a = a_{rd} \\ 10 & \text{if } a = a_{rc} \\ 15 & \text{if } a = a_{sw} \end{cases} \quad (35)$$

$$c_{m,3} = 5 \quad (36)$$

The different components of the indirect maintenance costs $c_i(a, t)$ are defined as follows:

$$c_{i,1}(t) = \begin{cases} 70 & \text{if } t \text{ is during the day} \\ 20 & \text{if } t \text{ is during the night} \end{cases} \quad (37)$$

$$c_{i,2}(a, t) = 55 \quad (38)$$

To plan or to postpone. We define the parameters defining the reward function for timely scheduling (see Section 5.2) as:

$$u_{\max} = 100 \quad (39)$$

$$\delta = 0.99 \quad (40)$$

$$\alpha = 5000 \quad (41)$$

Moreover, the transition probabilities are defined by (12)-(17), with the functions $f_C(\cdot)$ and $f_\sigma(\cdot)$ assumed to be known.

7.1.3. System-level cost functions

To define the cost function $f_{LF}(\cdot)$ expressing the additional loss of functionality when multiple components are maintained simultaneously, we divide all components into three groups:

g_A : Components of line A ‘‘Utrecht-Schiphol’’, i.e. $\mathcal{A}_{sc,1}, \dots, \mathcal{A}_{sc,n}, \mathcal{A}_{sw,1}, \dots, \mathcal{A}_{sw,k}$

g_B : Components of line B ‘‘Utrecht-Leiden’’, i.e. $\mathcal{B}_{sc,1}, \dots, \mathcal{B}_{sc,m}, \mathcal{B}_{sw,1}, \dots, \mathcal{B}_{sw,j}$

g_C : Components of line C ‘‘Leiden-Schiphol’’, i.e. $\mathcal{C}_{sc,1}, \dots, \mathcal{C}_{sc,l}, \mathcal{C}_{sw,1}, \dots, \mathcal{C}_{sw,i}$

and accordingly define the cost function $f_{LF}(\cdot)$ as:

$$f_{LF}(\mathcal{X}_t) = \begin{cases} 35 & \text{if } g_A \in \mathcal{X}_t \ \& \ g_B \in \mathcal{X}_t \\ 20 & \text{if } g_A \in \mathcal{X}_t \ \& \ g_C \in \mathcal{X}_t \ \& \ g_B \notin \mathcal{X}_t \\ 0 & \text{otherwise} \end{cases} \quad (42)$$

Simultaneously maintaining components on lines A and B or components on lines A and C means that there is no train connection between Utrecht and Schiphol. Hence combining maintenance activities on line A and B and on line A and C is penalized. Because there are other public transport options between Leiden and Schiphol, which are absent between Utrecht and Leiden, simultaneously maintaining lines A and B is penalized more severely.

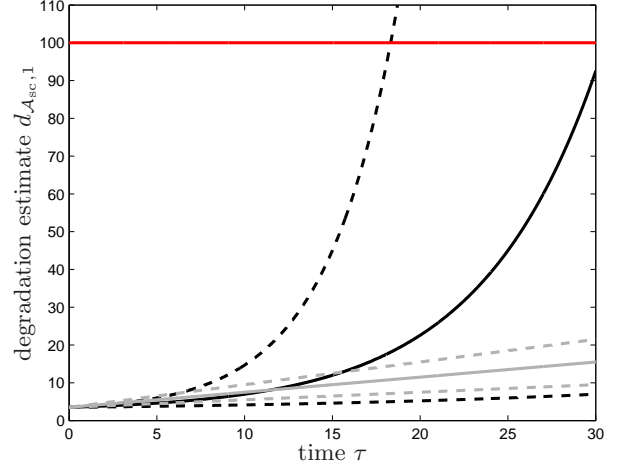


Figure 6: Prognosis result at τ_0 . The solid black line correspond to the expected time behavior of the degradation measure in case of a rail defect and the solid gray line represents the time behavior in case of rail contamination. The dashed lines represent the 2.5% and 97.5% bounds of the distribution.

7.2. Component-level optimization

Consider that maintenance is planned once a day, i.e.:

$$\Delta = 1 \text{ day}$$

At time $\tau_0 = 0$, section $\mathcal{A}_{sc,1}$, which is of type II indicates a need for maintenance. Component $\mathcal{A}_{sc,1}$ was last maintained 150 days ago, i.e. $t_{mnt} = \tau_{-150}$. At time τ_0 , the diagnosis result of component $\mathcal{A}_{sc,1}$ is specified as:

$$P(H(\tau_0) = f_{rd}) = 0.1$$

$$P(H(\tau_0) = f_{rc}) = 0.9$$

and the prognosis result as:

$$\begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} 2.5 \\ 1 \\ 0.15 \pm 0.1 \end{bmatrix}$$

$$\begin{bmatrix} \theta_4 \\ \theta_5 \end{bmatrix} = \begin{bmatrix} 3.5 \\ 0.4 \pm 0.2 \end{bmatrix}$$

The prognosis result is shown in Figure 6.

From the diagnostic and prognostic result, we conclude that if rail contamination is present, the system degrades slowly and the 95% confidence interval of the expected time to failure is [161, 483]. However, there is also a small possibility of a rail defect, in which case the system degrades much faster and the 95% interval of the expected time to failure is [18, 92].

Assume that at time τ_0 the following times are available for maintenance:

$$t_1 = \tau_0 + 0.2$$

$$t_p = \tau_{p-1}, \ p = 2, \dots, 500$$

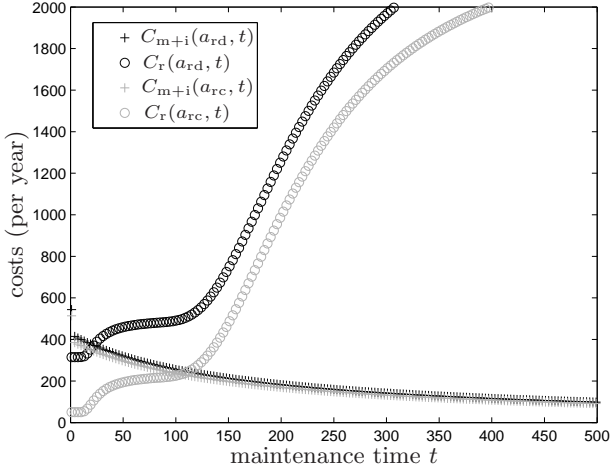


Figure 7: Cost components for the different maintenance strategies ($\tau = \tau_0$).

There is one immediate possibility (t_1) during the day in the case of an urgent fault. When the problem is not urgent, the maintenance will be scheduled at the most convenient time slot during night.

The optimal and suboptimal maintenance strategies are found based on (1)-(7), with the cost functions as defined in (29)-(38). We found the optimal maintenance strategy with associated costs:

$$(a^*, t^*) = (a_{rc}, t_{12})$$

$$C^* = 418.2$$

and the first ten alternative strategies are given in Table 1.

Table 1: Alternative maintenance strategies and associated costs at τ_0 .

| k | a_k^* | t_k^* | C_k^* | k | a_k^* | t_k^* | C_k^* |
|-----|----------|----------|---------|-----|----------|----------|---------|
| 2 | a_{rc} | t_{13} | 418.2 | 7 | a_{rc} | t_9 | 423.1 |
| 3 | a_{rc} | t_{11} | 419.2 | 8 | a_{rc} | t_{16} | 424.6 |
| 4 | a_{rc} | t_{14} | 419.3 | 9 | a_{rc} | t_8 | 425.4 |
| 5 | a_{rc} | t_{10} | 420.9 | 10 | a_{rc} | t_7 | 427.8 |
| 6 | a_{rc} | t_{15} | 421.5 | 11 | a_{rc} | t_{17} | 428.4 |

Figure 7 shows the cost components $C_{m+i}(a, t)$ ⁶ and $C_r(a, t)$ for the different maintenance strategies. For both types of maintenance a , the larger we chose t the lower the (direct plus indirect) maintenance costs $C_{m+i}(a, t)$, but the higher the risk costs $C_r(a, t)$. The total costs C for different strategies are shown in Figure 8.

The next step is to decide whether to accept this maintenance strategy or to postpone the maintenance decision

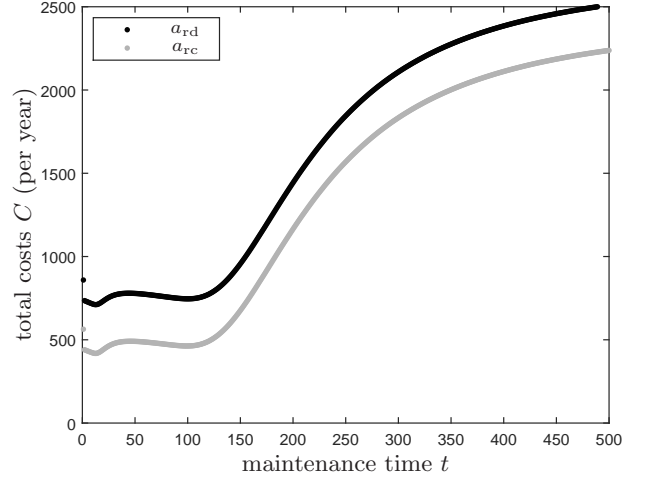


Figure 8: Expected costs for the different maintenance strategies ($\tau = \tau_0$).

to a later time (see Figure 3). The decision is postponed if it is expected that scheduling at a later monitoring instant results in a higher reward. So, we postpone if 1. it is expected that at least at one later monitoring instant the costs of the optimal maintenance strategy have reduced more than the associated penalty costs for postponing have increased; and 2. according to our sequential decision making strategy, we will actually decide to plan maintenance at one of these monitoring instants. Assuming that we have a certain prediction of the future costs and considering reward function (18), with the parameters as defined in (39)-(41), the decision is postponed if there exists $\tau_h > \tau_0$ for which:

$$\underbrace{(100 \cdot 0.99)^{\tau_h - \tau_0}}_{u_{\max}} - \underbrace{\hat{C}^*(\tau_h | \mathcal{I}_0)}_{\delta} (1 - P(\mathcal{F}(\tau_h) = 1)) - \underbrace{5000}_{\alpha} P(\mathcal{F}(\tau_h) = 1) \geq \underbrace{100}_{u_{\max}} - C^*(\tau_0)$$

with $\hat{C}^*(\tau_h | \mathcal{I}_0)$ the expected costs of the optimal maintenance strategy determined at τ_h given the available diagnostic and prognostic information up to τ_0 . At τ_0 , the probability of failure is negligible in the first few days. Because of the low prediction accuracy, we expect a cost reduction that is larger than the penalty cost of postponing. Therefore, we postpone the decision to τ_1 . Similarly we postpone at the next 149 decision instants τ_1 till τ_{149} , meaning that for all $\tau_i \in \{\tau_1, \dots, \tau_{149}\}$ there exists a $\tau_h > \tau_i$ for which:

$$(100 \cdot 0.99)^{\tau_h - \tau_i} - \hat{C}^*(\tau_h | \mathcal{I}_i) (1 - P(\mathcal{F}(\tau_h) = 1)) - 5000 P(\mathcal{F}(\tau_h) = 1) \geq 100 - C^*(\tau_i)$$

At time τ_{150} , the diagnosis result of component $\mathcal{A}_{sc,1}$ is specified as:

$$P(f_{rd}) = 0.0025$$

$$P(f_{rc}) = 0.9975$$

⁶ $C_{m+i}(a, t) = C_m(a, t) + C_i(a, t)$

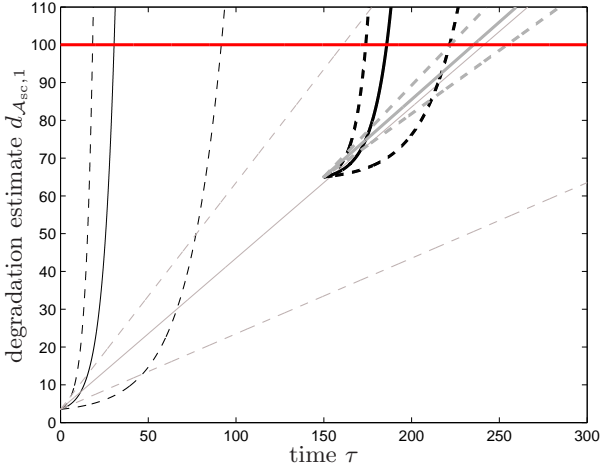


Figure 9: Prognosis results at τ_0 and τ_{150} . The black lines correspond to the expected time behavior of the degradation measure in case of a rail defect and the gray lines represent the time behavior in case of rail contamination. The dashed lines represent the 2.5% and 97.5% bounds of the distribution.

and the prognosis result as:

$$\begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} 64 \\ 1 \\ 0.10 \pm 0.05 \end{bmatrix}$$

$$\begin{bmatrix} \theta_4 \\ \theta_5 \end{bmatrix} = \begin{bmatrix} 65 \\ 0.41 \pm 0.075 \end{bmatrix}$$

The prognosis result is shown in Figure 9.

We conclude that the actual value of the degradation measure $d_{sc,1}(\tau_{150})$ is close to the value predicted by the parametrized model of rail contamination defined at τ_0 . The newly obtained parametrized models are however more accurate. Given that rail contamination is present, the 95% confidence interval of the time-to-failure distribution has reduced to $[222 - 150, 254 - 150]$. In case of a rail defect, the probability of which has become really small, the 95% confidence interval is $[174 - 150, 222 - 150]$.

Assume that the following maintenance time slots are available:

$$\begin{aligned} t_{150} &= \tau_{150} + 0.2 \\ t_p &= \tau_{p-1}, \quad p = 151, \dots, 500 \end{aligned}$$

The optimal and suboptimal maintenance strategies are found according to (1)-(7), with the cost functions as defined in (29)-(38):

$$\begin{aligned} (a^*, t^*) &= (a_{rc}, t_{201}) \\ C^* &= 174.0 \end{aligned}$$

The first ten alternative strategies and their associated costs are given in Table 2, and Figures 10 and 11 show

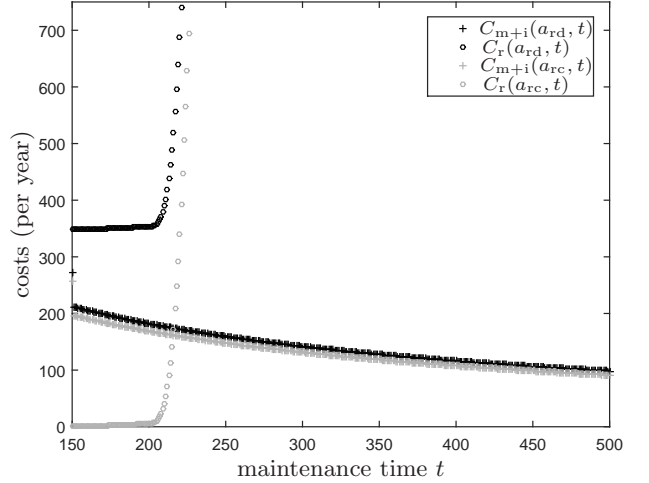


Figure 10: Cost components for the different maintenance strategies ($\tau = \tau_{150}$).

the costs for the different maintenance strategies. We conclude that the costs of the optimal strategy obtained at τ_{150} are significantly lower compared to the costs of the optimal strategy obtained at τ_0 . By postponing the decision we have limited the scheduling possibilities. However, the penalty cost for the delay in planning (maximum 100) is lower than the reduction in maintenance costs ($418.2 - 174.0$). We assume that the expected cost reduction with later policies is small. Therefore, we accept this maintenance strategy.

Table 2: Alternative maintenance strategies and associated costs at τ_{150} .

| k | a_k^* | t_k^* | C_k^* | k | a_k^* | t_k^* | C_k^* |
|-----|----------|-----------|---------|-----|----------|-----------|---------|
| 2 | a_{rc} | t_{202} | 174.0 | 7 | a_{rc} | t_{197} | 175.2 |
| 3 | a_{rc} | t_{200} | 174.1 | 8 | a_{rc} | t_{204} | 175.2 |
| 4 | a_{rc} | t_{203} | 174.4 | 9 | a_{rc} | t_{196} | 175.6 |
| 5 | a_{rc} | t_{199} | 174.4 | 10 | a_{rc} | t_{195} | 176.0 |
| 6 | a_{rc} | t_{198} | 174.8 | 11 | a_{rc} | t_{194} | 176.5 |

7.3. System-level optimization

Consider that 7 system components, namely $\mathcal{A}_{sc,1}$, $\mathcal{A}_{sc,2}$, $\mathcal{A}_{sw,1}$, $\mathcal{B}_{sc,1}$, $\mathcal{B}_{sw,1}$, $\mathcal{C}_{sc,1}$, and $\mathcal{C}_{sw,1}$, are in need of maintenance. For section $\mathcal{A}_{sc,1}$, maintenance is already scheduled (see Table 3). For each other component, the optimal and near-optimal component-level strategies are given in Table 3.

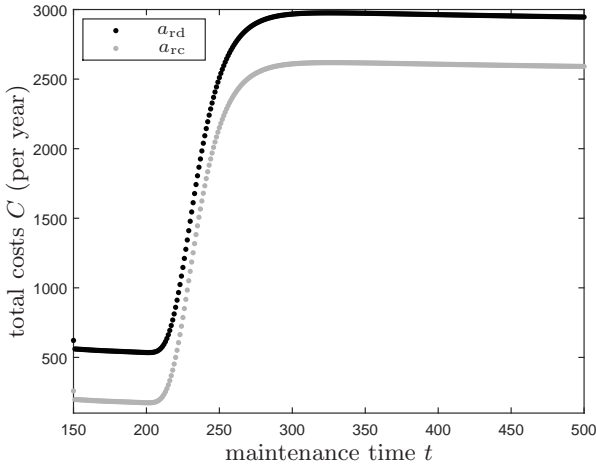


Figure 11: Expected costs for the different maintenance strategies ($\tau = \tau_{150}$).

Table 3: Inputs system-level optimization and optimal system-level strategy (marked in gray).

| component | action a | time t | costs C |
|----------------------|------------|-----------|-----------|
| $\mathcal{A}_{sc,1}$ | a_{rc} | t_{202} | 181.6 |
| $\mathcal{A}_{sc,2}$ | a_{rc} | t_{155} | 186.2 |
| | a_{rc} | t_{180} | 190.0 |
| $\mathcal{A}_{sw,1}$ | a_{rc} | t_{202} | 242.6 |
| | a_{sw} | t_{155} | 195.0 |
| | a_{sw} | t_{180} | 191.7 |
| | a_{sw} | t_{202} | 198.3 |
| $\mathcal{B}_{sc,1}$ | a_{rc} | t_{155} | 201.0 |
| | a_{rc} | t_{180} | 191.2 |
| $\mathcal{B}_{sw,1}$ | a_{rc} | t_{202} | 179.8 |
| | a_{sw} | t_{155} | 181.0 |
| | a_{sw} | t_{180} | 211.2 |
| $\mathcal{C}_{sc,1}$ | a_{sw} | t_{202} | 233.0 |
| | a_{rc} | t_{155} | 203.6 |
| | a_{rc} | t_{180} | 191.2 |
| | a_{rc} | t_{202} | 169.8 |
| $\mathcal{C}_{sw,1}$ | a_{sw} | t_{155} | 201.5 |
| | a_{sw} | t_{180} | 189.3 |
| | a_{sw} | t_{202} | 165.4 |

Minimizing the system-level optimization criterion (25) results in the following system-level maintenance schedule:

- at t_{155} : $\mathcal{B}_{sw,1}$
- at t_{180} : $\mathcal{A}_{sc,2}$
- at t_{220} : $\mathcal{A}_{sc,1}, \mathcal{A}_{sw,1}, \mathcal{B}_{sc,1}, \mathcal{C}_{sc,1}, \mathcal{C}_{sw,1}$

So, except for switch $\mathcal{B}_{sw,1}$ and section $\mathcal{A}_{sc,2}$, maintenance is scheduled simultaneously with the maintenance on section $\mathcal{A}_{sc,1}$ at t_{220} . For components $\mathcal{B}_{sc,1}$, $\mathcal{C}_{sc,1}$, and $\mathcal{C}_{sw,1}$,

t_{202} coincides with the component-level optimal time. For switch $\mathcal{A}_{sw,1}$, t_{202} is not optimal. However, the cost benefit of combining maintenance on two switches is larger than the increase of the component-level maintenance costs when scheduling the maintenance at a near-optimal time. At t_{155} maintenance on line B is scheduled, meaning that scheduling maintenance on $\mathcal{A}_{sw,1}$ at t_{155} results in an additional penalty of 35. Therefore, the maintenance of switch $\mathcal{A}_{sw,1}$ is scheduled at t_{202} .

Maintenance for section $\mathcal{A}_{sc,2}$ and switch $\mathcal{B}_{sw,1}$ is scheduled at another time. For these components, the costs of performing maintenance at t_{202} are too high compared to the costs at their optimal component-level time t_{155} , i.e. the reduction as a consequence of economics of scale does not outweigh the increase in component-level strategy costs. To avoid penalty costs due to performing maintenance on line A and B simultaneously, maintenance on $\mathcal{A}_{sc,2}$ is scheduled at the first alternative maintenance time t_{180} .

Note that the system-level strategy found is a direct result of the adopted cost functions. In the case the cost reductions from economics of scale are extremely high, all components will be maintained at t_{220} . When loss of functionality is severely penalized, all components on line A will be maintained at maintenance time t_{155} and t_{202} , while all components on line B and C will be maintained at t_{180} .

8. Conclusions

We have proposed a two-stage optimization approach to timely maintenance planning in heterogeneous systems. In the first stage, the maintenance needs of the individual system components are determined. In the second stage we optimize the maintenance schedule at the system level. More specifically:

1. We optimize both the required type of maintenance and the time to perform the maintenance;
2. We optimize the time to settle on the aforementioned maintenance decisions, hereby trading accuracy with timeliness;
3. We decouple the maintenance optimization from the diagnosis and prognosis process. In this way, we are able to exploit both diagnostic and prognostic information for maintenance optimization without restricting ourselves to a particular degradation model;
4. We provide a systematic framework for incorporating economic and structural dependencies among system components.

Advantages of the proposed approach are: 1. at the component level, maintenance is planned timely when this does not lead to violations of cost and safety constraints. Timely planning allows to perform maintenance at a convenient time, to inform users regarding system downtime, and to optimize the management of spare parts, material, and personnel; 2. at the system level, maintenance

costs are significantly reduced by adequately combining or spreading maintenance activities. The applicability of the method is demonstrated through a case study concerning maintenance planning in a railway network.

The proposed method can be extended and improved in various ways. Some possible directions for future research are:

1. Include the next monitoring time instant as a decision variable. This would be beneficial for systems for which continuously high-frequency monitoring is expensive or not possible, e.g. measurement trains.
2. Include the decision to perform additional monitoring. In the context of the railway case, this could e.g. refer to additional measurements by a measurement train when the track-side monitoring data do not provide enough information to make an informed decision.
3. Include additional maintenance options, e.g. minimal maintenance, major maintenance, and repair.
4. Make costs savings due to economics of scale dependent on the component's location.
5. Include the possibility of rescheduling maintenance.
6. Optimize the order and frequency at which maintenance activities are planned in the system-level optimization.

Acknowledgment

This research is part of STW/ProRail project “Advanced monitoring of intelligent rail infrastructure (ADMIRE)”, project 12236, which is supported by ProRail and the Dutch Technology Foundation STW, which is part of the Netherlands Organization for Scientific Research (NWO), and which is partly funded by the Ministry of Economic Affairs.

The research leading to these results has received funding from the People Programme (Marie Curie Actions) of the European Union’s Seventh Framework Programme (FP7/2007-2013) under REA grant agreement nr. 324432 (AMBI project).

References

- [1] N. Sathaye, S. Madanat, A bottom-up solution for the multi-facility optimal pavement resurfacing problem, *Transportation Research Part B: Methodological* 45 (7) (2011) 1004–1017.
- [2] A. Medury, S. Madanat, System-level optimization of maintenance and replacement decisions for road networks, in: 11th International Conference on Applications of Statistics and Probability in Civil Engineering, 2011, pp. 235–242.
- [3] H. Yeo, Y. Yoon, S. Madanat, Algorithms for bottom-up maintenance optimisation for heterogeneous infrastructure systems, *Structure and Infrastructure Engineering* 9 (4) (2013) 317–328.
- [4] A. Furuya, S. Madanat, Accounting for network effects in railway asset management, *Journal of Transportation Engineering* 139 (1) (2012) 92–100.
- [5] G. Loewenstein, D. Read, R. F. Baumeister, *Time and Decision: Economic and Psychological Perspectives of Intertemporal Choice*, Russell Sage Foundation, 2003.

- [6] R. Agrawal, S. Jeong, R. Velu, Timing when to buy, in: *Proceedings of the 20th ACM international conference on Information and knowledge management*, 2011, pp. 709–718.
- [7] I. Fisher, et al., *The Theory of Interest*, Kelley, 1965.
- [8] C.-A. Robelin, S. M. Madanat, Reliability-based system-level optimization of bridge maintenance and replacement decisions, *Transportation Science* 42 (4) (2008) 508–513.
- [9] M. Knowles, D. Baglee, S. Wermter, Reinforcement learning for scheduling of maintenance, in: *Research and Development in Intelligent Systems XXVII*, Springer, 2011, pp. 409–422.
- [10] B. Wu, Z. Tian, M. Chen, Condition-based maintenance optimization using neural network-based health condition prediction, *Quality and Reliability Engineering International* 29 (8) (2013) 1151–1163.
- [11] S. V. Amari, L. McLaughlin, H. Pham, Cost-effective condition-based maintenance using Markov decision processes, in: *Symposium on Reliability and Maintainability*, 2006, pp. 464–469.
- [12] J. Y. J. Lam, D. Banjevic, A myopic policy for optimal inspection scheduling for condition based maintenance, *Reliability Engineering & System Safety* 144 (2015) 1 – 11.
- [13] L. Wang, J. Chu, W. Mao, A condition-based order-replacement policy for a single-unit system, *Applied Mathematical Modelling* 32 (11) (2008) 2274–2289.
- [14] W. Wang, H. Wang, Preventive replacement for systems with condition monitoring and additional manual inspections, *European Journal of Operational Research* 247 (2) (2015) 459–471.
- [15] K. T. Huynh, A. Barros, C. Brenguer, Multi-level decision-making for the predictive maintenance of k-out-of-n deteriorating systems, *IEEE Transactions on Reliability* 64 (1) (2015) 94–117.
- [16] N. Chen, Z.-S. Ye, Y. Xiang, L. Zhang, Condition-based maintenance using the inverse Gaussian degradation model, *European Journal of Operational Research* 243 (1) (2015) 190–199.
- [17] D. Tang, J. Yu, X. Chen, V. Makis, An optimal condition-based maintenance policy for a degrading system subject to the competing risks of soft and hard failure, *Computers & Industrial Engineering* 83 (2015) 100 – 110.
- [18] F. Camci, System maintenance scheduling with prognostics information using genetic algorithm, *IEEE Transactions on Reliability* 58 (3) (2009) 539–552.
- [19] A. Van Horenbeek, L. Pintelon, A dynamic predictive maintenance policy for complex multi-component systems, *Reliability Engineering & System Safety* 120 (2013) 39–50.
- [20] Z. Tian, T. Jin, B. Wu, F. Ding, Condition based maintenance optimization for wind power generation systems under continuous monitoring, *Renewable Energy* 36 (5) (2011) 1502–1509.
- [21] K. Bouvard, S. Artus, C. Brenguer, V. Cocquempot, Condition-based dynamic maintenance operations planning & grouping. Application to commercial heavy vehicles, *Reliability Engineering & System Safety* 96 (6) (2011) 601–610.
- [22] B. Castanier, A. Grall, C. Bérenguer, A condition-based maintenance policy with non-periodic inspections for a two-unit series system, *Reliability Engineering & System Safety* 87 (1) (2005) 109–120.
- [23] K.-A. Nguyen, P. Do, A. Grall, Multi-level predictive maintenance for multi-component systems, *Reliability Engineering & System Safety* 144 (2015) 83–94.
- [24] M. C. A. Olde Keizer, R. H. Teunter, J. Veldman, Clustering condition-based maintenance for systems with redundancy and economic dependencies, *European Journal of Operational Research* 251 (2) (2016) 531–540.
- [25] R. Dekker, R. E. Wildeman, F. A. van der Duyn Schouten, A review of multi-component maintenance models with economic dependence, *Mathematical Methods of Operations Research* 45 (3) (1997) 411–435.
- [26] A. Ompusunggu, J.-M. Papy, S. Vandenplas, Kalman filtering based prognostics for automatic transmission clutches, *IEEE/ASME Transactions on Mechatronics* 21 (1) (2016) 419–430.
- [27] A. Coraddu, L. Oneto, A. Ghio, S. Savio, D. Anguita, M. Figari, Machine learning approaches for improving condition-

- based maintenance of naval propulsion plants, *Journal of Engineering for the Maritime Environment*, 2014.
- [28] X.-S. Si, W. Wang, C.-H. Hu, D.-H. Zhou, Estimating remaining useful life with three-source variability in degradation modeling, *IEEE Transactions on reliability* 63 (1) (2014) 167–190.
- [29] L. Bian, N. Gebraeel, Computing and updating the first-passage time distribution for randomly evolving degradation signals, *IIE Transactions* 44 (11) (2012) 974–987.
- [30] J. Chen, R. J. Patton, *Robust Model-Based Fault Diagnosis for Dynamic Systems*, Springer Publishing Company, 2012.
- [31] Z. Cherfi, L. Oukhellou, E. Côme, T. Denoeux, P. Akinin, Partially supervised independent factor analysis using soft labels elicited from multiple experts: application to railway track circuit diagnosis, *Soft Computing* 16 (5) (2012) 741–754.
- [32] E. Fumeo, L. Oneto, D. Anguita, Condition based maintenance in railway transportation systems based on big data streaming analysis, *Procedia Computer Science* 53 (2015) 437 – 446.
- [33] N. Z. Gebraeel, M. A. Lawley, R. Li, J. K. Ryan, Residual-life distributions from component degradation signals: A Bayesian approach, *IIE Transactions* 37 (6) (2005) 543–557.
- [34] J. Sikorska, M. Hodkiewicz, L. Ma, Prognostic modelling options for remaining useful life estimation by industry, *Mechanical Systems and Signal Processing* 25 (5) (2011) 1803–1836.
- [35] R. Fletcher, *Practical methods of optimization*, John Wiley & Sons, 2013.
- [36] O. Etzioni, R. Tuchinda, C. A. Knoblock, A. Yates, To buy or not to buy: mining airfare data to minimize ticket purchase price, in: *Proceedings of the ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003, pp. 119–128.
- [37] W. Groves, M. Gini, On optimizing airline ticket purchase timing, *ACM Transactions on Intelligent Systems Technology* 7 (1) (2015) 3:1–3:28.
- [38] W. B. Powell, *Introduction to Markov Decision Processes*, John Wiley & Sons, Inc., 2011, pp. 57–109.
- [39] E. A. Feinberg, A. Shwartz, *Handbook of Markov Decision Processes: Methods and Applications*, Vol. 40, Springer Science & Business Media, 2012.
- [40] D. P. Bertsekas, *Dynamic programming and optimal control*, Vol. 1, Athena Scientific Belmont, MA, 1995.
- [41] M. van Otterlo, M. Wiering, Reinforcement learning and Markov decision processes, in: *Reinforcement Learning*, Springer, 2012, pp. 3–42.