On the Use of Prognostics and Health Management to Jointly Schedule Production and Maintenance on a Single Multi-Purpose Machine

Omar Bougacha
AS2M Department, FEMTO-ST
Université de Bourgogne Franche-Comté
Besançon, France
omar.bougacha@femto-st.fr

Christophe Varnier
AS2M Department, FEMTO-ST
Université de Bourgogne Franche-Comté
ENSMM, CNRS
Besançon, France
christophe.varnier@femto-st.fr

Noureddine Zerhouni
AS2M Department, FEMTO-ST
Université de Bourgogne Franche-Comté
ENSMM, CNRS
Besançon, France
noureddine.zerhouni@femto-st.fr

***Abstract*—This paper address the problem of using prognostic information in the decision-making process of a single multi-purpose machine. The prognostics and health management method is compared to condition-based maintenance combined with a genetic algorithm to determine the joint schedule of maintenance and production. The paper presents a methodology to select the adequate strategy while considering several factors that influence the functioning of the machine. The results show that operational and conditions variability influence the choice of the suitable methods. In the presented case, we show configurations where prognostic information is useless or useful.**

***Keywords*—Production and Maintenance Scheduling, Predictive Maintenance, Condition-Based Maintenance, Prognostic Information, Ant Colony Optimization, Genetic Algorithm**

I. Introduction

Maintenance optimization has been extensively studied in the literature. One can find several reviews for the maintenance policies [1], [2] and for the maintenance optimization techniques [3], [4]. Some works investigated the joint optimization of spare part inventory and maintenance planning. One can refer to the review paper done by Van Horenbeek *et al.* on this subject [5].

Production planning has been in its turn extensively studied. Consider several methods for production scheduling [6] use exact methods [7], heuristics [8], and meta-heuristics [9].

However, the two services: production and maintenance in the industrial context, are highly inter-dependent. Some works have studied the optimization of this joint problem [10]–[13]. With the emergence of new methodologies like condition-based maintenance and prognostics and health management, it has become difficult to choose what method should be applied in each scenario. Almost any work in condition-based maintenance or prognostics and health management context proposes to compare its approach to other classic maintenance approaches most commonly systematic periodic maintenance. The authors in the works of Camci [14] or Langeron *et al.* [15], prove that the use of prognostics and health management is more beneficial than other policies. Therefore, one can wonder if this statement is true no matter what type of component or machine.

Very few works have compared their results to the different maintenance policies to select the suitable one. For example, Van Horenbeek and Pintelon in [16] proposed a prognostic-based predictive maintenance policy, the results of their method are compared to several maintenance policies such as condition-based maintenance, and classic age-based policies. They also studied the influence of some parameters like the dependencies between the components and the prognostic horizon on the policies' performances. Hence, providing readers with a methodology to select a suitable maintenance policy. However, the authors have only done this study for a particular machine configuration. In other words, one can conclude on the choice of the optimal policy only if a similar machine configuration is present. Therefore, the question of which policy is more suitable for other configurations remains unanswered.

Moreover, the work of Van Horenbeek and Pintelon, like many other works, has not included the production scheduling and how it can influence the performance of the maintenance policy. The works on CBM or PHM maintenance scheduling implicitly assume that the operating conditions of the machines and their future loads are constant such as the works of Camci [14], Shi and Zeng [17], and Langeron *et al.* [15].

In this paper, we propose to solve the joint problem of production and maintenance scheduling on a single multi-purpose machine using two meta-heuristics based on PHM and CBM methodologies. The proposed methods' performances are analyzed on a multiple components system. We also propose a methodology to select the appropriate method according to the system properties and the problem characteristics. Therefore, several machine configurations are considered (i.e., the degradation speed of components, the operational profiles, and some production problem characteristics). These test cases allow us to conclude when it is better to use one method instead of the other.

The rest of this paper is organized as follows. In section 2, the joint problem of maintenance and production scheduling is defined. The used methodologies to solve the problem are described in section 3 along with the comparison methodology. A numerical example and the results are presented in section 4. Finally, this work is concluded and some future works are presented in section 5.

II. Problem Statement

This work deals with the joint problem of production scheduling and predictive maintenance planning for a single multi-purpose machine. The production scheduling problem focuses on building a timetable for production jobs with a suitable profile. However, few constraints need to be considered while creating this schedule (e.g., order deadline and due date and maintenance inspection). Before scheduling order, a prognostic algorithm assesses the machine's ability to achieve the task. The decision-making algorithm considers this information while solving the scheduling problem. Predictive maintenance is scheduled to find a compromise between early maintaining the equipment, risking its failure, and missing opportunities due to machine unavailability.

A. The Machine Model

Let us considered a multi-purpose single machine (e.g., computer numerical control (CNC) machines). CNC machines are known for their ability to operate on different types of products with different configurations (cutting speed, feed rate, and various tools). In this context, we assume that the machine can produce N types of products with J possible profiles. The machine is assumed to be a serial multi-component system i.e. if one of its components fails the whole system fails. Let us denote L the number of components. Each component l is subject to degradation that is influenced by the product type n and the production profile j. The degradation evolution of component l is a function of the product type n, the production profile j used, and the quantity to produce x. For simulation purposes, we assume that the degradation $H_l(x, j, n)$ of a component is described by an exponential function (1).

$$H_l(x, j, n) = a_l(e^{(b_l * S_j * s_n * x)} - 1) \quad (1)$$

Where a_l and b_l are two parameters that define the shape of the exponential function for a component l. S_j is the coefficient that reflects the influence of production profile j. s_n reflects the severity of the producing product n. The RUL of component l is defined as the time left during which component l can still be used before its degradation level reached a failure threshold, denoted Th_l.

B. Production Scheduling Problem

During each period i, $K(i)$ production orders $O_{i,k}$ have to be processed. Each order k is characterized by a product type n_k, a quantity of products Q_k, a release data r_k, a due date d_k, and a deadline D_k. The production problem consists of finding the best schedule σ_i^* of the $O_{i,k}$ for the i^{th} period and the profile used for this job. Each order k has a duration of processing noted $p_{k,j}$ under each profile j, and a completion date $c_{k,j}$. For any scheduled job, if its completion date $c_{k,j}$ exceeds its due date then a penalty will be paid, and the penalty value is fixed whatever the duration of the delay. Any order that exceeds its deadline is considered as a lost opportunity. The scheduled jobs have a production cost $Cp(k, j)$ (2) and generate a gain from selling the products $Gp(k)$ (3).

$$Cp(k, j) = Q_k * C_{n_k,j} + U_k * Q_k * LP_{n_k} \quad (2)$$

$$Gp(k) = Q_k * P_{n_k} \quad (3)$$

With $C_{n_k,j}$ is the cost of producing one unit of product n_k with profile j, $U_k = 1$ if the job k is late, LP_{n_k} is the tardiness penalty for product type n_k, and P_{n_k} is the sale price of product type n_k. Any produced schedule should satisfy some constraints:

- No preemption between jobs is allowed,
- the machine can only produce one job at the time,
- the sum of processing time of all scheduled jobs should not exceed the period duration (i.e., the decision horizon)

C. Maintenance Problem

During the operating time of the machine, its components are subject to wear and tear. When a component degradation level reaches a failure threshold, the machine is no longer able to fulfill the required service. To avoid failure and to maximize the machine's availability and reliability, maintenance interventions are scheduled. We assume that the maintenance quality is perfect, meaning that if one component is scheduled for maintenance it regains an "as good as new" condition after maintenance. Each component l of the machine has a replacement cost noted M_l. When a component is scheduled for maintenance, a maintenance cost is generated for this component. This maintenance cost is defined in (4) as the sum of the replacement cost and a penalty if it is maintained early. This penalty depends on the difference between the current degradation level and the failure threshold. However if the component degradation level exceeds the threshold ($H_l \geq Th_l$), the component l fails $CM_l = 1$ involving a corrective maintenance activity with a cost P_{CM}.

$$C_m(l) = M_l + P_{rul} * max(Th_l - H_l, 0) + P_{CM} * CM_l \quad (4)$$

As the machine has several components, the cost of a maintenance action of replacing a set of components noted $\mathcal{PM}$ is defined in (5). The maintenance activity does not have a direct effect on the estimated gain. Therefore, the gain of a maintenance decision G_m is assumed to be equal to zero.

$$C_m = \sum_{l \in \mathcal{PM}} M_l + P_{rul} * max(Th_l - H_l, 0) + P_{CM} * CM_l \quad (5)$$

D. Joint Problem

The objective behind this application is to find a suitable compromise between the production and the maintenance activities. This joint problem aims to establish a settlement between two services maintenance and production. The simulation horizon SH will be divided into smaller decision horizon called periods. At each period i, the joint schedule σ_i of maintenance and production activity is built out of the local decisions of production and maintenance. The objective function can be described

$$max \sum_{i \in SH} B_i = max \sum_{i \in SH} G_i - C_i \tag{6}$$

With B_i the benefits of the i^{th} step and C_i and G_i are respectively the cost and the gain of the i^{th} step as described in (7) and (8).

$$C_i = \sum_{(k,j) \in \sigma_i} C_p(k,j) + \sum_{l \in \mathcal{PM}_\rangle} C_m(l) + L_{op} * Idl_i \tag{7}$$

$$G_i = \sum_{(k,j) \in \sigma_i} G_p(k) \tag{8}$$

With: L_{op} is a penalty on the time interval Idl_i during which the machine is capable of producing but it is idle for the lack of a production order. Idl_i is defined as in (9), with DH_i, $Prod_i$ and $Maint_i$ are respectively the decision horizon (i.e., the available time for production), the time where the machine is used to produce and the time spent in maintenance in the i^{th} step.

$$Idl_i = DH_i - Prod_i - Maint_i \tag{9}$$

III. Resolution

A. Prognostic-based method

Unlike in the classic prognostic-based decision-making process in which the decision is built independently from the machine's ability to achieve the task and only based on a single estimated RUL, we propose to build the production schedule iteratively and based on short-term prognostic. Therefore, at each instant t, the decision builder algorithm considers the list of available orders, combines them with the possible production profiles, and sends this list to the short-term prognostic algorithm. The prognostic algorithm estimates the degradation evolution of the different components under each of these decisions and creates the list of feasible decisions (i.e., the final degradation level does not exceed the failure threshold). This list is feed once again to the decision algorithm, which evaluates and selects the appropriate one. This cycle is called a decision construction step. It is repeated until the scheduled time reaches the decision horizon.

Therefore, we obtain an iterative process of building the schedule by considering a dynamic prognostic process. We qualify this decision process as a closed-loop decision building. Figure 1 presents this dynamic of building. Figure 2 presents the sequential diagram of this process.

The construction loop of prognostic-based decisions is implemented in a modified ant colony optimization (ACO) algorithm [18]. The modification incorporates the short-term prognostics and some technicalities about the decision building

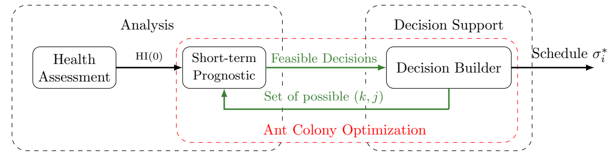

Fig. 1. Decision building loop

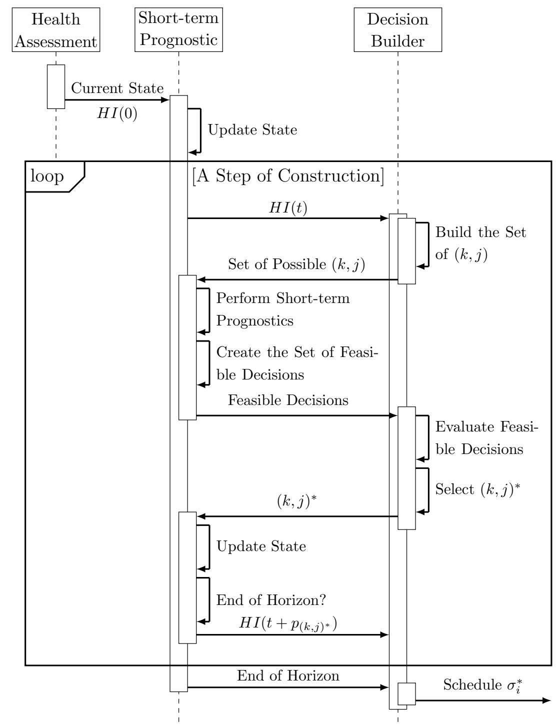

Fig. 2. Decision building sequence diagram

process. The obtained algorithm is described in previous work [19]. We will refer to this method as **PHM+**.

The proposed PHM+ method is compared to a classic condition-based maintenance combined with a genetic algorithm to schedule production activities.

B. Condition-based maintenance method

To solve the joint problem, we combined condition-based maintenance with a genetic algorithm. At the start of each period, if the period corresponds to the inspection date, then an inspection of the components degradation level takes place. Components in which the degradation level exceeds the CBM threshold are scheduled for maintenance at the start of the period. Then, the remaining time is scheduled for production using the genetic algorithm (GA). In this paper, this method is referred to as **CBM-GA**.

To implement GA, several components should be considered:

- the genetic representation of the solution,
- the fitness function,
- the method to generate initial population,
- the genetic operators (mutation and crossover), and

- the survival rules.

Even though GA does not guarantee the global optimum solution, it is a commonly used method in cases of combinatorial, high instances or non-linear optimization problems [20]. The required components of the GA implementation are presented in this subsection.

1) Solution representation: The algorithm search for a good combination of orders and the profile to produce them. Then, we propose a coding consisting of a 2D matrix composed of two lines of integers. The first line contains the sequence of orders, while the second line contains the production profile to be used. Then each allele is composed of the order index k and a production profile j.

2) Fitness function: Individuals in the GA are evaluated to measure their fitness toward an objective function. Here, the fitness evaluation uses the benefit function defined by (6). The evaluation is obtained through simulating the outcome of the chromosome (i.e., we simulate the application of the individual's schedule to the machine and we evaluate the outcome cost).

3) Initial population: GA requires an initial generation (or set) of valid solutions. Here, a solution needs to verify some constraints to be considered as valid. These constraints consist of having a schedule with a duration that does not exceed the duration of a period and in which each order has a completion time lower than its deadline (see 10 and 11). The individuals of the initial population are built randomly by combining orders and production profiles then scheduling them in random order. Invalid individuals are modified to guarantee that the constraints are respected.

$$\sum_{(k,j)\in\sigma} p_{k,j} \leq \ DH_i \ \forall \ i \ \in \ SH \tag{10}$$

$$c_{k,j} \leq D_k \ \forall \ (k,j) \ \in \ \sigma \tag{11}$$

4) Genetic operators: The proposed GA uses two genetic operators: mutations and crossover. For the mutation, we proposed two operators: (i) a single point mutation operator in which a randomly selected allele is mutated by exchanging the current order with another order from the list of unscheduled available orders, and (ii) random two-point mutation operator in which two alleles are selected and are exchanged. In this algorithm, we used a classic order crossover (also noted OX crossover).

5) Survival rules: New generations are created from the previous generation survivors and genetic operators' offspring. Here, a novel generation is built by; (i) $X_{Survival}\%$ of the best chromosomes from previous generation, (ii) $X_{Mutation}\%$ of the mutation's offspring , and (iii) $X_{Crossover}\%$ of the crossover's offspring. The offspring selection is based on a roulette wheel process using the fitness value for defining the selection probability. Since the objective is to maximize the fitness function, the probability of each individual can be defined as in (12).

$$Prob(Indiv_i) = \frac{fitness(Indiv_i)}{\sum_{Indiv\in Offspring} fitness(Indiv)} \tag{12}$$

C. Comparison protocol

To study the selection of the adequate method, we need to compare the two methodologies in several test cases that take into consideration different factors. In this application, the considered factors are:

- The degradation's speed of the component, three categories are considered: (i) rapidly deteriorating, (ii) normally deteriorating, and (iii) slowly deteriorating. The speed of the deterioration is determined by the component's coefficients a_l and b_l. These components are used to create four machines as presented in Table I.
- The initial condition of the components. Four sets of initial conditions are considered: (i) all components are new, (ii) components are at the first half of their life with random degradation level, (iii) a mix of components degradation levels one new, one used, and on at the end of its life, and (iv) all components have the same degradation level and they are at the second half of their life. These initial conditions are presented in Table II.
- Three categories of production orders were considered. The categories are defined according to the quantity of the demanded product; small tasks where $Q \in [10, 50]$, medium tasks where $Q \in [30, 100]$ and large tasks where $Q \in [100, 200]$.
- Product type: to capture the influence of the product type, we considered two cases: (i) 5 sets of production orders that contain three different types of products, and (ii) 5 sets of orders that have the same unique type of product.
- Production profile: to capture the influence of the production speed, we defined two cases: (i) the machine can only produce with one speed, and (ii) the machine can be used with three different profiles (i.e. three speeds; low, normal, and high).

TABLE I
MACHINES CONFIGURATIONS

Machines	Components		
	l=1	l=2	l=3
M_1	rapid	slow	rapid
M_2	slow	normal	rapid
M_3	rapid	rapid	rapid
M_4	slow	slow	slow

TABLE II
INITIAL CONDITIONS CONFIGURATIONS

Initial Condition	Degradation Level		
	D1	D2	D3
IC1	0	0	0
IC2	0.3	0.4	0.24
IC3	0.8	0.4	0
IC4	0.6	0.6	0.6

For each machine, we consider 4 initial conditions for 3 categories of production orders each presents 10 sets of production orders (5 with a single product, and 5 with multiple products). This leads to a total of 120 test cases for each machine when using a single speed and 120 test cases when using three speeds.

IV. Results and discussion

Tables III and IV present the obtained results respectively for the single-speed case and the multiple profiles. Each row of these tables presents for a specific machine the mean of benefits over 20 test cases (i.e., 5 production orders sets and 4 initial conditions). All the values in these tables are expressed in k u.m. (thousand units of currency). One can easily note that the benefits are more important for the slow deteriorating machine (M4) compared to the rapidly deteriorating machine (M3). The rapid the degradation of the component gets, the more frequently it undergoes maintenance and the higher the maintenance cost gets which causes the benefits to getting lower.

TABLE III
Results of the single-speed tests

		Machine	CBM-GA	PHM+	Difference
Small Tasks	Single Product	M1	27.1	35.9	8.8
		M2	27.7	37.2	9.5
		M3	12.1	28.3	16.2
		M4	31.8	34.8	3
	Multiple Products	M1	18.6	31.1	12.5
		M2	24.5	32.6	8.1
		M3	10	25.2	15.2
		M4	28	30.8	2.8
Medium Tasks	Single Product	M1	27.8	32.4	4.6
		M2	28.2	30.6	2.4
		M3	14.6	25.5	10.9
		M4	31.5	28.9	-2.6
	Multiple Products	M1	20.6	27.4	6.8
		M2	26	28.4	2.4
		M3	9.8	22.8	13
		M4	28.7	25	-3.7
Large Tasks	Single Product	M1	27.3	26	-1.3
		M2	27.1	26.1	-1
		M3	14.8	20.3	5.5
		M4	30.9	23.7	-7.2
	Multiple Products	M1	23.2	21.5	-1.7
		M2	26.8	22.5	-4.3
		M3	9.7	17.5	7.8
		M4	30.3	19.3	-11

In table III, one can note that the difference between PHM+ and CBM is always positive for the small tasks. In other words, it is more beneficial to use PHM+ in this case. By comparing the values of the difference between different machines, one can notice that the benefits of using PHM+ reach its maximum for the machine M3 which is composed of only rapidly deteriorating component. Furthermore, the benefit of using PHM+ has a minimum value for the machine with the slowest deteriorating components. Machines with slow deteriorating components are easier to control from a health management perspective. Although the variation of future decisions influences the evolution of the degradation, the machine's states do not present a big variation between two inspection dates. Therefore, the use of PHM+ does not produce a big difference.

TABLE IV
Results of tests on multiple profiles

		Machine	CBM-GA	PHM+	Difference
Small Tasks	Single Product	M1	27.1	40.5	13.4
		M2	27.7	40.9	13.2
		M3	12.1	26.1	14
		M4	31.8	45.4	13.6
	Multiple Products	M1	18.6	33.5	14.9
		M2	24.5	36.2	11.7
		M3	10	26.9	16.9
		M4	28	40.5	12.5
Medium Tasks	Single Product	M1	27.8	38.6	10.8
		M2	28.2	41	12.8
		M3	14.6	30.5	15.9
		M4	31.5	44.9	13.4
	Multiple Products	M1	20.6	33.1	12.5
		M2	26	35.7	9.7
		M3	9.8	27.2	17.4
		M4	28.7	40	11.3
Large Tasks	Single Product	M1	27.3	35.5	8.2
		M2	27.1	38.2	11.1
		M3	14.8	28.2	13.4
		M4	30.9	43.1	12.2
	Multiple Products	M1	23.2	31.2	8
		M2	26.8	33.7	6.9
		M3	9.7	25.3	15.6
		M4	30.3	38.7	8.4

When the tasks get larger, the machine is spending more time producing the same product with the same operational conditions. Thus, the variation of future decisions is reduced. One can notice that the reduction of future decisions' variation reduces significantly the benefits of the PHM+ on one hand. On the other hand, the benefits of CBM are almost stable. Therefore, the difference between the two methods decreases.

The decrease in the PHM+ benefits is more important for slow degrading machines. In the case of medium tasks, one can note that the difference is negative meaning that it is more beneficial to use CBM. The values of the PHM+ benefits continue to decrease when orders get larger. For large tasks, one can notice that only machine M3 has a positive difference. In cases in which the machines have different components' degradation dynamic or only slow deteriorating components, the CBM method has the biggest benefits.

One can conclude that when the machine's overall degradation dynamic becomes slow, the PHM+ benefits decrease. Thus, for slow degradation machines, it is more beneficial to use CBM. Furthermore, When the variation of the future loads decreases the PHM+ benefits decrease and the CBM becomes more practical than PHM+. These observations allow us to conclude on the use of PHM+ or CBM with the genetic algorithm in the case of a single-speed machine. The obtained classification is presented in Figure 3.

Table IV presents the difference in benefits between using PHM+ and CBM for a single machine composed of three subsystems capable of functioning with several production profiles. The table presents the results for a single product type and multiple products.

One can notice that for these test cases the benefits of using

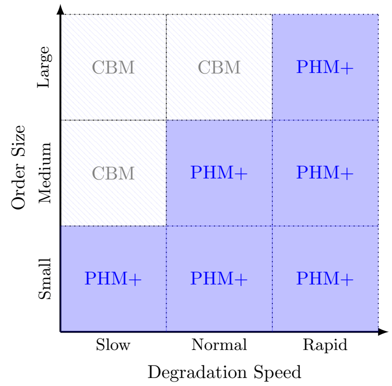

Fig. 3. Use mape of PHM+ vs CBM for a single speed test case

PHM+ are more important than using CBM even for large tasks. When using several production profiles, the variability in the degradation evolution is higher. This variability cannot be detected with CBM, in this case.

However, one note that the degradation dynamic has the same effect as in the case of a single-speed machine. The difference between the two methods is maximized for machines with rapidly deteriorating components. One can also note that the smaller the tasks get the higher the difference between the two methods. Therefore, the variability of the production orders has the same effect as the production profiles.

To conclude, the higher the variability in the machine's operational conditions the more beneficial it is to use PHM+. The operational conditions are varied using:

- different product types,
- different production orders, and
- different production profiles.

One can also conclude that the machine parameters especially the degradation dynamic of its components influences the method selected for the joint optimization problem.

V. Conclusion

In this paper, we studied the importance of using prognostic information and techniques to solve the joint production and maintenance optimization for a single multi-components machine.

We also studied the influence of the machine characteristics and the variability of its operational conditions on the choice of the appropriate maintenance strategy.

The obtained results suggest that for slow deteriorating machines with low operational variability one can settle for condition-based maintenance strategy. While for high operational variability and/or rapid deteriorating machines, it is more beneficial to use prognostics and health management strategies.

In this work, we supposed that the inspection cost is negligible and we did not consider them into the objective function. It would be interesting for future work to include the inspection cost. One can also include the costs of the deployment of different maintenance strategies.

Another possible future work is to study the utility of using PHM for a more complex system (e.g. for multi-component systems with different inter-component dependencies).

References

[1] B. Xu, X. Han, M. Li, and D. Xiao, "System maintenance scheduling: review and prospect," in *IEEE PES Innovative Smart Grid Technologies*. IEEE, 2012, pp. 1–6.

[2] S.-H. Ding and S. Kamaruddin, "Maintenance policy optimization—literature review and directions," *The International Journal of Advanced Manufacturing Technology*, vol. 76, no. 5-8, pp. 1263–1283, Feb. 2015.

[3] R. Ahmad and S. Kamaruddin, "A review of condition-based maintenance decision-making," *European J. of Industrial Engineering*, vol. 6, no. 5, p. 519, 2012.

[4] E. Ruschel, E. A. P. Santos, and E. d. F. R. Loures, "Industrial maintenance decision-making: A systematic literature review," *Journal of Manufacturing Systems*, vol. 45, pp. 180–194, 2017.

[5] A. Van Horenbeek, J. Buré, D. Cattrysse, L. Pintelon, and P. Vansteenwegen, "Joint maintenance and inventory optimization systems: A review," *International Journal of Production Economics*, vol. 143, no. 2, pp. 499–508, 2013.

[6] S. C. Graves, "A review of production scheduling," *Operations Research*, vol. 29, no. 4, pp. 646–675, Aug. 1981.

[7] J. M. Moore, "An n job, one machine sequencing algorithm for minimizing the number of late jobs," *Management Science*, vol. 15, no. 1, pp. 102–109, Sep. 1968.

[8] J. Branke, S. Nguyen, C. W. Pickardt, and M. Zhang, "Automated design of production scheduling heuristics: A review," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 1, pp. 110–124, Feb. 2016.

[9] H. Aytug, M. Khouja, and F. E. Vergara, "Use of genetic algorithms to solve production and operations management problems: A review," *International Journal of Production Research*, vol. 41, no. 17, pp. 3955–4009, Jan. 2003.

[10] J. Kaabi, C. Varnier, and N. Zerhouni, "Heuristics for scheduling maintenance and production on a single machine," in *IEEE International Conference on Systems, Man and Cybernetics*, vol. 5. Yasmine Hammamet, Tunisia: IEEE, october 2002, p. 5.

[11] F. Benbouzid, C. Varnier, and N. Zerhouni, "Resolution of joint maintenance/production scheduling by sequential and integrated strategies," in *Artificial Neural Nets Problem Solving Methods*, J. Mira and J. R. Álvarez, Eds. Springer Berlin Heidelberg, 2003, vol. 2687, pp. 782–789.

[12] M. A. selsouli, N. Najib, A. Mohafid, and E. Aghezzaf, "An Integrated Production and Maintenance Planning Model with time windows and shortage cost," *IFAC Proceedings Volumes*, vol. 42, no. 4, pp. 822–827, 2009.

[13] Q. Liu, M. Dong, F. Chen, W. Lv, and C. Ye, "Single-machine-based joint optimization of predictive maintenance planning and production scheduling," *Robotics and Computer-Integrated Manufacturing*, vol. 55, pp. 173–182, Feb. 2019.

[14] F. Camci, "System Maintenance Scheduling With Prognostics Information Using Genetic Algorithm," *IEEE Transactions on Reliability*, vol. 58, no. 3, pp. 539–552, Sep. 2009.

[15] Y. Langeron, M. Fouladirad, and A. Grall, "Controlled systems, failure prediction and maintenance," in *8th IFAC Conference on Manufacturing Modelling, Management and Control MIM*, vol. 49, Troyes, France, june 2016, pp. 805–808.

[16] A. Van Horenbeek and L. Pintelon, "A dynamic predictive maintenance policy for complex multi-component systems," *Reliability Engineering and System Safety*, vol. 120, pp. 39–50, Dec. 2013.

[17] H. Shi and J. Zeng, "Real-time prediction of remaining useful life and preventive opportunistic maintenance strategy for multi-component systems considering stochastic dependence," *Computers & Industrial Engineering*, vol. 93, pp. 192–204, Mar. 2016.

[18] M. Dorigo and G. Di Caro, "Ant colony optimization: a new meta-heuristic." IEEE, 1999, pp. 1470–1477.

[19] O. Bougacha, C. Varnier, N. Zerhouni, and S. Hajri-Gabouj, "A post-prognostic decision approach for production and maintenance planning," in *International conference on computers and industrial engineering proceeding (CIE 48)*, 2018, pp. 257–273.

[20] L. Davis, *Handbook of genetic algorithms*, 1st ed. London: Chapman & Hall, 1991.