IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 10, 2017, accepted October 29, 2017, date of publication November 17, 2017,
date of current version February 14, 2018.

Digital Object Identifier 10.1109/ACCESS.2017.2771827

Reinforcement Learning-Based and Parametric Production-Maintenance Control Policies for a Deteriorating Manufacturing System

A. S. XANTHOPOULOS[1], ATHANASIOS KIATIPIS[2], D. E. KOULOURIOTIS[1], AND SEPP STIEGER[2]
[1]Department of Production and Management Engineering, Democritus University of Thrace, 671 00 Xanthi, Greece
[2]Fujitsu Technology Solutions GmbH, 80807 Munich, Germany

Corresponding author: Athanasios Kiatipis (athanasios.kiatipis@ts.fujitsu.com and tkiatipis@gmail.com)

The work of A. Kiatipis and S. Stieger was supported by the BigStorage: Storage-Based Convergence Between HPC and Cloud to Handle Big Data project from the European Union through the Marie Skłodowska-Curie Actions framework under Grant H2020-MSCA-ITN-2014-642963.

ABSTRACT The model of a stochastic production/inventory system that is subject to deterioration failures is developed and examined in this paper. Customer interarrival times are assumed to be random and backorders are allowed. The system experiences a number of deterioration stages before it ultimately fails and is rendered inoperable. Repair and maintenance activities restore the system to its initial and previous deterioration state, respectively. The duration of both repair and maintenance is assumed to be stochastic. We address the problem of minimizing the expected sum of two conflicting objective functions: the average inventory level and the average number of backorders. The solution to this problem consists of finding the optimal tradeoff between maintaining a high service level and carrying as low inventory as possible. The primary goal of this research is to obtain optimal or near-optimal joint production/maintenance control policies, by means of a novel reinforcement learning-based approach. Furthermore, we examine parametric production and maintenance policies that are often used in practical situations, namely, Kanban, (s, S), threshold-type condition based maintenance and periodic maintenance. The proposed approach is compared with the parametric policies in an extensive series of simulation experiments and it is found to clearly outperform them in all cases. Based on the numerical results obtained by the experiments, the behavior of the parametric policies as well as the structure of the control policies derived by the Reinforcement Learning-based approach is investigated.

INDEX TERMS Inventory control, preventive maintenance, reinforcement learning, intelligent manufacturing systems.

I. INTRODUCTION

Over the past few years, advances in Information Technology (IT) have transformed production operations in modern plants. For example, the emergence of Radio Frequency Identification (RFID) technology, and the subsequent development of Internet of Things, has enabled the accurate tracking of inventory levels within a production system at any given time. Moreover, the deterioration level of manufacturing equipment can be effectively and continuously monitored by means of computerized systems.

Nevertheless, in many practical situations, the production and maintenance planning functions are still largely ad hoc. Typically, a maintenance plan is devised and then taken as granted, without taking into consideration possible interactions with production scheduling decisions. On the other hand, production operations are often controlled using simple heuristics, based on the experience and know-how of scheduling specialists and manufacturing engineers.

The primary goal of this research is to obtain optimal or near-optimal, *integrated* maintenance and production control policies for deteriorating, stochastic production/inventory systems. To this end, Reinforcement Learning (RL) methods are used, along with discrete-event simulation. More specifically, the proposed approach consists of interfacing RL-based, decision-making agents with simulation models of the investigated production/inventory systems. A simulation model generates sample paths of the system dynamic evolution. The decision-making agent interacts with the

2169-3536 © 2017 IEEE. Translations and content mining are permitted for academic research only.
Personal use is also permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

simulation model by observing the current system state and selecting some admissible control action. Subsequently, the decision-making agent is presented with the outcome of its action, i.e. the new state in which the system has transited to and a numerical value that represents the *relative* merit of making the aforementioned selection. This cycle is repeated sufficiently many times and through this learning process, the agent determines the best control action for each system state, i.e. the optimal control policy.

The secondary goal of this research is to study prominent *parametric* production and maintenance policies that are often used in practice. We examine the most well known *pull* type and *push* type production policies, namely Kanban and (s, S). The origins of the Kanban policy can be found in the Toyota automotive industries of the early 70's and this control mechanism is currently being used in modern plants worldwide. The (s, S) policy has also received considerable attention in the relevant scientific literature over the years. In respect to parametric maintenance policies, condition-based maintenance and periodic maintenance is investigated. According to a periodic maintenance policy, the production facility is maintained at fixed time intervals, whereas a condition based maintenance policy makes maintain/no maintain decisions based on the current deterioration level of the production equipment. Both approaches are often used in many practical situations.

The proposed joint maintenance and production control approach is compared to the parametric policies in an extended series of meticulously conducted simulation experiments. The most important aspects of this research can be summarized in the following points:

- the model of a stochastic, production/inventory system with deterioration failures and minimal maintenance/repair actions is introduced and the relevant production/maintenance optimization problem is formulated. The proposed model extends previously published work in this field and it has not been examined in the relevant literature up to now, to the authors' knowledge.
- a novel approach for deriving optimal or near-optimal, joint maintenance and production control policies is proposed. The proposed approach is based on Reinforcement Learning and the detailed description of its development is provided in this paper.
- parametric production and maintenance policies, that are often encountered in practical situations, are investigated. The performance of the parametric control polices is compared to that of the proposed approach in an extensive series of simulation experiments.

The remainder of this paper is structured as follows. A synopsis of relevant publications is presented in section II. The system description and the related optimization problem are given in section III. The parametric production and maintenance policies that are examined in this research are described in section IV. The detailed implementation of the proposed RL-based approach is given in section V. The results from the simulation experiments are presented and commented upon in section VI. The paper is concluded with section VII, where some directions for future research are also outlined.

II. RELATED WORK

In recent years, there has been a surge in the literature regarding combined production and maintenance problems. Published papers that fall within this category vary significantly in numerous aspects, including: the description/assumptions of the investigated system, the optimization problem formulation and the solution approach. In the following paragraph, an overview of the most recent and prominent papers is presented.

Single-machine systems are examined in [1]–[6], whereas flow lines, i.e. systems which consist of several machines in series that are separated by buffers, are studied in [7] and [8]. Hajej *et al.* [9] investigate a rather singular system, in the sense that it comprises of a manufacturing facility coupled with an output buffer and an additional inventory location. Finished goods are stored in the output buffer and then transported to the inventory location so as to be delivered to customers. The production systems examined in the existing literature either manufacture multiple product types ([2] and [6]) or a single product type ([5], [10], [11]). Typically, the deterioration state of the production facility is considered to be known at all times to the decision-maker. Nevertheless, there are some publications (e.g. [1] and [5]) where the deterioration state is determined by means of periodic inspections, whereas He *et al.* [12] consider periodic as well as *imperfect* inspections. In the majority of published works, the assumption is made that a deteriorated system has the ability to produce, normally at a lower service rate, end-items of acceptable quality before it ultimately fails. However, in [3], [4], [11], and [13] imperfect production quality is considered, i.e. system deterioration leads to a fraction of the manufactured items to be non-conforming in terms of quality. Maintenance activities can be either minor or major ([1], [14]). A major maintenance restores the system to the good-as-new state whereas a minor maintenance merely decreases the deterioration level of the production system. Typically, time or monetary costs are associated with maintenance activities but in [10] and [15] the provisioning of the necessary spare parts is also taken into consideration. Existing publications on joint production and maintenance control can be categorized in respect to whether the relevant decisions are made over an infinite or a finite horizon ([3], [9], [10], [16]). In establishing the optimal production/maintenance policy or program, production, inventory, backorder, and maintenance costs, among others, are typically considered. The modeling/solution approaches are also quite disparate and span from mixed-integer ([6]) and linear-quadratic programming ([10]), to Reinforcement Learning ([8]) and Dynamic Programming ([3]).

This paper primarily extends the work of [17]–[21]. Xanthopoulos *et al.* [17] develop the continuous time Markov chain model of a single-machine Kanban system that experiences a number of deterioration levels until it

ultimately fails. The deterioration of the system is determined by periodic inspections and maintenance decisions are based on a threshold type policy. Mathematical expressions of several performance metrics are derived and two related mixed integer optimization problems are solved, by means of an augmented Langrangian genetic algorithm. Yao *et al.* [18] study a discrete time production system in the presence of time-dependent failures, corrective/preventive maintenance and constant demand. Joint production and maintenance policies are derived by means of Markov decision process modeling and the structural properties of the optimal policy are investigated. Chen and Trivedi [19] study a continuous time Markovian model of a deteriorating machine with stochastic processing, failure, inspection, maintenance and repair times. Minimal and major maintenance decisions are made on the basis of a double threshold policy. Closed form analytical expressions of availability and mean-time-to-fail metrics are derived. Das and Sarkar [20] examine an unreliable, stochastic production/inventory system operating under the (s, S) policy. They develop the related Markov chain model and derive expressions for the metrics of service level, average inventory level and system productivity. Their goal is to find the optimal maintenance policy in respect to an objective function that consists of the following components: additional revenue due to increased service level, savings related to repair and maintenance costs per unit time. Iravani and Duenyas [21] investigate a stochastic single machine system that experiences a number of deterioration stages. Preemption, as well as lost sales, is allowed. The problem of obtaining joint production maintenance control policies is modeled as a semi-Markov decision process. The optimal policy, in respect to minimizing inventory, repair and lost sales costs, is derived by means of Dynamic Programming.

III. SYSTEM DESCRIPTION

The definition of the symbols used in Section III and III-A are given in Table 1. The system consists of a manufacturing facility coupled with a finished goods buffer. The facility can produce a single type of end-items. Raw materials are assumed to be continuously available. The manufacturing facility can process one item at a time, while no preemption is allowed, i.e. once the processing of an item has started, it cannot be interrupted prior to its completion. The processing times are exponentially distributed with a mean value of $1/\lambda_p$. A completed item is stored in the finished goods buffer, provided that there are no backorders at that time. The capacity of the finished goods buffer is I_{max}. Consequently, the manufacturing facility is idling if there are I_{max} items in the buffer.

Customer demand is random and the time interval between two successive demand arrivals is considered to be exponentially distributed with a mean value of $1/\lambda_a$. All demand quantities are assumed to be constant and equal to one unit of the produced end item. If there is available inventory at the time of a demand arrival, then the demand is satisfied instantaneously. On the contrary, if there is no inventory available, then the customer demand is backordered. The discipline of the backorders queue is FCFS (first-come-first-served). As soon as an end item is produced, the first customer in the backorders queue receives the item and exits the system. The maximum allowed length of the backorders queue is B_{max}. If there are B_{max} pending demands and a new demand arrives at the system, then this demand is discarded (lost sales).

TABLE 1. Definition of symbols pertaining to system description and problem formulation (section iii).

Symbol	Description
I_{max}	maximum allowed inventory level
B_{max}	maximum allowed number of backorders
d	number of deterioration stages
$1/\lambda_a$	mean time between arrivals
$1/\lambda_p$	mean service time (processing time)
$\lambda_{f,i}$	deterioration failure rate in stage i
$1/\mu_r$	mean time to repair
$1/\mu_{m,i}$	mean time to maintain in stage i
T	duration of simulation model replication
i	i-th deterioration stage
$I(t)/B(t)/i(t)$	inventory level/number of backorders/deterioration stage at time t
$\bar{I}\,/\,\overline{B}$	mean inventory level/mean number of backorders
ω	realization of random variables used in simulation
E	expected value operator
$\mathbf{x}$	vector of input parameters used in simulation

The manufacturing facility is subjected to *deterioration failures* (soft failures). A deterioration failure can take place only when the facility is processing an item, i.e. not when it is idling or under maintenance/repair. The state of the facility, regarding its deterioration level, is described by d stages. The facility is considered to be in deterioration stage 0 if it is in an ''as-good-as-new'' state. The occurrence of a deterioration failure in stage i (where $i < d$) causes the system to transit to the next deterioration stage $(i + 1)$. The manufacturing facility has the ability to operate as long as it is in deterioration stage $0, 1, \ldots, d$. If the manufacturing system experiences a deterioration failure in stage d, it breaks down (hard failure) and repair actions are initiated. Repair renders the facility as-good-as-new (deterioration stage 0) and its duration is considered to be an exponential random variable with a mean value of $1/\mu_r$. The times between successive deterioration failures are also exponentially distributed. The failure rate in stage i is $\lambda_{f,i}$, where $i = 0, 1, \ldots, d$, and it is reasonable to assume that $\lambda_{f,0} < \lambda_{f,1} < \ldots < \lambda_{f,d}$, i.e. the more deteriorated the facility is, the more frequently failures occur.

Hard failures can be prevented by carrying out maintenance actions. In this paper, we consider *minimal maintenance* actions, i.e. maintaining the facility in deterioration state i (where $0 < i \leq d$) causes it to transit to stage $i - 1$ (as-bad-as-before). It is noted that, in the case of a hard failure, repair is the only available option and no mainte-

nance can be carried out. Maintenance times are exponentially distributed and the mean time to maintain in stage i is $1/\mu_{m,i}$, where $i = 1, 2, \ldots, d$. The reasonable assumption that $\mu_{m,1} > \mu_{m,2} > \ldots. > \mu_{m,d}$ is made, that is, the more deteriorated the system is, the more time is needed to maintain it.

A. PROBLEM FORMULATION

In the context of the system described in section III, the following optimization problem is addressed in this paper, which is both computationally challenging and also important from a managerial point of view:

$$\min E\left\{\overline{I}\left(\mathbf{x}, \omega\right) + \overline{B}\left(\mathbf{x}, \omega\right)\right\} \tag{1}$$

In equation (1), $E\{\cdot\}$ is the expected value operator, $\overline{I}$ is the mean inventory level, $\overline{B}$ is the mean length of the backorders queue, $\mathbf{x}$ is a vector of input parameters, e.g. arrival rate, processing rate, failure rate etc., and ω is a realization of the relevant random variables (i.e. inter-arrival times, processing times etc.). The expected values of $\overline{I}$ and $\overline{B}$, for a given set of input parameters, are obtained by means of simulation experiments. The mean inventory and mean number of backorders computed *in a single replication* of the simulation model is:

$$\overline{I} = \frac{1}{T}\int_0^T I(t)dt \tag{2}$$

$$\overline{B} = \frac{1}{T}\int_0^T B(t)dt \tag{3}$$

where T is the length of the simulation replication, and $I(t)/B(t)$ is the inventory level/number of backorders at simulated time t.

The mean number of backorders reflects the customer service level and excessive finished goods inventories are typically regarded as a waste of resources, so these two metrics should be minimized. Nevertheless, inventory offers protection against unexpected fluctuations of the demand and production process. Therefore, the objectives of minimizing $\overline{I}$ and $\overline{B}$ are conflicting and, consequently, the aim is to find a good trade-off between them. Ideally, the manufacturing system should be controlled in a way that facilitates a high service level, while carrying as low inventory as possible.

The purpose of this research is to study parametric joint production – maintenance control policies as well as to derive optimal or near – optimal policies, in respect to the problem defined in equation (1).

IV. PARAMETRIC PRODUCTION AND MAINTENANCE POLICIES

In sections IV.A – IV.D, a description of some important parametric production control and maintenance policies, which were investigated in the context of this research, is given. These policies do not come with optimality guarantees; nevertheless, they are often used in practical situations. This is because they are easy to implement and they are characterized by a few parameters which can be fine-tuned, in order to yield satisfactory results in complex environments.

TABLE 2. Definition of symbols pertaining to parametric production and maintenance policies (section v).

Symbol	Description
b	maintenance threshold for condition based maintenance
TBM	time between consecutive periodic maintenance activities
t_i^m	time of i – th periodic maintenance activity
K	parameter of Kanban production control policy
s, S	parameters of (s, S) production control policy
P	raw materials buffer (in description of Kanban control policy)
D	queue that contains kanban cards (in description of Kanban control policy)

Maintenance policies can be largely categorized as *condition based* or *periodic*. According to a periodic maintenance policy, the production equipment is maintained at fixed time intervals, regardless of its actual deterioration level. On the other hand, a condition based maintenance policy makes the relevant decisions solely on the basis of the current deterioration stage of the manufacturing facility.

Production control policies mostly fall within two broad categories: *push type* and *pull type*. In a manufacturing system that operates under a pull type control policy, production decisions are driven by actual occurrences of demand. On the contrary, according to a push type policy, the inventory is replenished up to a target level, without having specific customer requests.

A. CONDITION BASED MAINTENANCE

In condition based maintenance, only the deterioration state of the manufacturing facility determines the action that will be performed. Typically, a threshold deterioration level is defined, and the manufacturing facility is maintained once this threshold is reached. In this paper, a parametric maintenance policy is examined, that is completely characterized by parameter b, i.e. the deterioration threshold:

IF current deterioration stage $= i$ AND $d \geq i \geq b$
  carry out minimal maintenance
ELSE
  do nothing
END IF

B. PERIODIC MAINTENANCE

In the periodic maintenance framework, the time intervals between successive maintenance epochs are of equal, fixed length. The parameter, which characterizes a time-driven maintenance policy, is the size TBM of these intervals (refer to Table 2). If $\mathbf{t}^m = \left\{t_1^m, t_2^m, t_3^m \ldots.\right\}$ is the sequence of time-points where minimal maintenance occurs, then this sequence can be calculated according to:

$$t_1^m = TBM$$

$$t_{i+1}^m = t_i^m + TBM \tag{4}$$

In every maintenance cycle, the manufacturing facility is restored to the as-bad-as-before state, regardless of its current deterioration level.

C. KANBAN PRODUCTION CONTROL POLICY

Fig. 1 shows a Kanban system with a single manufacturing facility; B is the backorders queue, I is the finished goods inventory, MF symbolizes the manufacturing facility, D is a queue that contains *kanban* cards (production authorizations) and P is the raw materials buffer (assumed to be non-empty at all times).

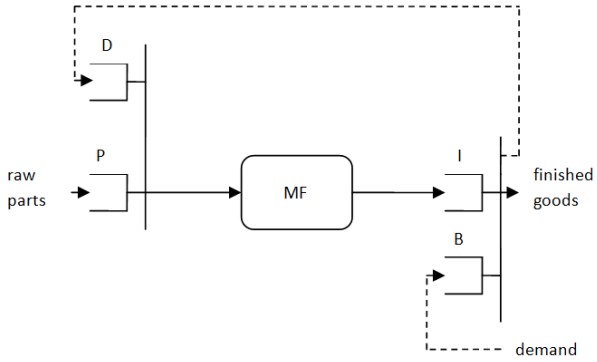

FIGURE 1. Kanban system with a single manufacturing facility and backorders.

Initially, there are $K > 0$ end-items in the output buffer and the manufacturing facility is idle. Each end-item has a kanban card attached to it. Parameter K (the fixed number of kanban cards) fully characterizes this control policy and corresponds to the maximum number of end items allowed in the output buffer. It follows that $K = I_{max}$ (refer to section III).

When an end-item exits the output buffer, the associated kanban card is detached from it and it is forwarded to the manufacturing facility, in order to authorize the production of a new item. Once the new item is manufactured, it is stored in the output buffer, with its kanban card attached to it.

The information of a customer demand arrival is transmitted to the manufacturing facility via the flow of the kanban cards. This transmission is interrupted if the output buffer is empty at the time when a customer demand arrives to the system. Kanban cards are released to the manufacturing facility based solely on actual demand realizations, consequently constituting the Kanban mechanism a pull type control policy ([22]).

D. (s, S) PRODUCTION CONTROL POLICY

The (s, S) policy belongs to the family of push type methods, as it controls production on the basis of a target inventory level and not on actual customer demand arrivals ([23]). This control policy is characterized by two parameters, namely s and S ($s < S$), where S is the maximum allowed inventory of finished goods, and thus $I_{max} = S$.

The manufacturing facility is idling as long as the inventory level is above s. At the time when the finished goods stock drops to s, the manufacturing facility is turned on and it is authorized to produce until the inventory is replenished up to S. At that time point, the state of the facility transits to idle again and this production cycle is repeated perpetually (refer to Fig. 2).

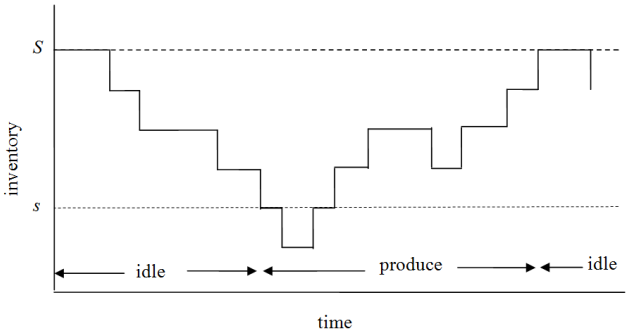

FIGURE 2. Production cycle in a system operating under the (s, S) policy.

TABLE 3. Definition of symbols pertaining to reinforcement learning-based approach (section v).

Symbol	Description
$\mathbf{s}_t$	state at time t
S	set of states
$A(s_1, s_2)$	admissible actions at state $\mathbf{s} = (s_1, s_2)$
$t_{d,i}$	i – th decision epoch
$r_{t_{d,i+1}}$	reward for action selection in i – th decision epoch
ρ	average reward
π	control policy of decision-making agent
$Q(\mathbf{s}, c)$	estimated "value" of selecting action c in state $\mathbf{s}$
$\mathbf{s}_{t_{d,i}}$	state at decision epoch i
$c_{t_{d,i}}$	selected action at decision epoch i
a, β	parameters of R-learning algorithm
e	parameter of e – greedy exploration strategy

V. REINFORCEMENT LEARNING BASED JOINT MAINTENANCE AND PRODUCTION CONTROL

In order to derive optimal or near-optimal integrated maintenance and production control policies, Reinforcement Learning (RL) is employed. According to the RL paradigm, a decision-making agent is placed within an environment whose dynamics are initially unknown ([24], [25]).

The agent interacts at certain time points (decision epochs) with its environment. The timing of the decision epochs is not known beforehand and depends on the dynamic evolution of the controlled system, i.e. the agent environment.

At a decision epoch, the agent receives a representation of the environment's current state and selects some action from a set of admissible controls. At the next decision epoch, the agent observes the result of its previous action selection, i.e. the new state of the environment and a numerical reward that quantifies the *relative* merit of making that decision. This cycle is repeated and after a sufficient number of

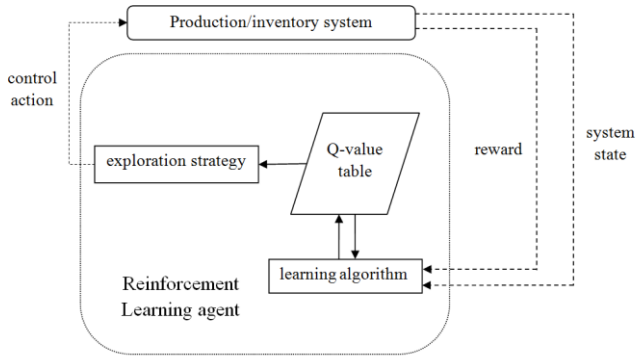

FIGURE 3. Interface between production system and decision-making agent.

decision epochs, the agent identifies through the process of trial-and-error the optimal control action in respect to some performance metric for each state.

In this research, the agent environment is the production/inventory system described in section III. The system dynamic behavior is obtained by means of discrete-event simulation ([26]). The goal of the agent is to minimize the expected sum of the mean finished goods inventory and the mean length of the backorders queue. In order to obtain the optimal or near-optimal joint maintenance and production control policies, the decision-making/learning agent is interfaced with the production system simulation model. The following elements need to be defined, in order to fully describe the agent-environment interface:

1. the timing of the decision epochs
2. the state representation/state space of the controlled system, as perceived by the agent
3. the agent admissible controls, i.e. a mapping from the system state space to the agent's action set
4. the rewards received by the agent and the associated goal/objective function
5. the agent learning algorithm and Q– value table, where the control policy computed by the agent is stored
6. the agent *exploration* strategy

The overall agent-environment interface is presented graphically in fig. 3. In sections V.A-V.F, we elaborate on the details of the implementation (the definition of the symbols therein is given in Table 3).

A. DECISION EPOCHS

The agent selects the control actions only when the manufacturing facility is idling, that is, the agent does not interfere with ongoing operations, e.g. it cannot interrupt the processing of a part or some maintenance/repair activity. Consequently, there are decision epochs at the following time points:

- at a new customer demand arrival, provided that the facility is idling at that time
- at the completion of an end-item. Note that, at that time, the manufacturing facility's state switches from working to idle, and a produce/idle/maintain authorization is expected by the controller
- at the completion of a minimal maintenance activity
- at the completion of a repair activity

There is only one exception to the above; the agent does not make a decision at the time when production/maintenance/repair is completed if the current inventory level and the deterioration stage is I_{max} and 0, respectively. In that case, the agent cannot authorize the production of a new part or initiate minimal maintenance, thus a decision epoch is redundant.

B. STATE REPRESENTATION AND STATE SPACE

In every decision epoch, the agent receives a representation of the production system's current state and selects an action on that basis. The state of the system at time t, as it is perceived by the agent, is defined to be the vector:

$$\mathbf{s}_t = \left(s_{1,t}, s_{2,t}\right) = (I(t) - B(t), i(t)) \tag{5}$$

where $I(t)$ is the finished goods inventory, $B(t)$ is the number of backorders, and $i(t)$ is the deterioration state of the facility at time t, respectively. The expression $I(t)$ - $B(t)$ is often referred to as the *inventory position* of the system and assumes integer values whereas the second component of $\mathbf{s}_t$ takes on non-negative integer values. The complete state space of the system can be written as:

$$S = \{(s_1, s_2) \mid s_1 = -B_{\max}, -B_{\max}+1, ..., I_{\max}, \quad s_2 = 0, 1, ..., d\} \tag{6}$$

where I_{max}, B_{max}, d is the maximum allowed inventory, the maximum allowed number of backorders and the number of deterioration stages, respectively (refer to section III for details). Consequently, the number of alternative states that the decision-making agent can find itself in is:

$$card(S) = (I_{\max} + B_{\max} + 1) \times (d+1) - 1 \tag{7}$$

since $(s_1, s_2) = (I_{max}, 0)$ does not constitute a decision-making state as explained in section V.A.

C. ADMISSIBLE ACTIONS

The agent can i) authorize the production of a new end-item, ii) authorize a minimal maintenance of the manufacturing facility or, iii) authorize the facility to remain idle. The *admissible controls function* $A(s_1, s_2)$ defines a mapping from system states to agent actions and determines which actions are available to the agent for every state in S:

$$A(s_1, s_2) = \{\text{idle, produce}\}, \quad s_1 = -B_{\max}, ..., I_{\max} - 1, \quad s_2 = 0 \tag{8}$$

$$A(s_1, s_2) = \{\text{idle, maintain}\}, \quad s_1 = I_{\max}, \quad s_2 = 1, 2, 3, ..., d \tag{9}$$

$$A(s_1, s_2) = \{\text{idle, produce,maintain}\}, \quad s_1 = -B_{\max}, ..., I_{\max} - 1, \quad s_2 = 1, 2, \ldots, d \tag{10}$$

Clearly, there is no point in maintaining the manufacturing facility if it is as-good-as-new. Moreover, equation (9) shows that the agent cannot authorize production if the maximum

allowable inventory I_{max} has been reached. In all other cases, the agent can select either one of the three available actions.

D. REWARDS AND GOAL OF THE DECISION-MAKING AGENT

Let $\mathbf{t}_d = \{t_{d,1}, t_{d,2}, t_{d,3}, ...\}$ denote the sequence of time-points, where the learning agent makes a decision. In decision epoch $t_{d,i}$, the agent selects an action and in the next step $t_{d,i+1}$, in part as a result of its selection, it receives a numerical reward $r_{t_{d,i+1}}$:

$$r_{t_{d,i+1}} = \frac{1}{t_{d,i+1} - t_{d,i}} \int_{t_{d,i}}^{t_{d,i+1}} (-I(t) - B(t))\, dt \qquad (11)$$

where $I(t)$ and $B(t)$ is the inventory level and the number of backorders at time t. Expression (9) is the negative of (mean inventory level + mean number of backorders) in the time interval $[t_{d,i}, t_{d,i+1}]$ between two successive decision epochs. The reward signal provides the agent with information regarding the *relative* cost of selecting some action in a certain state. Formally stated, the agent goal is to maximize the expected average reward per decision epoch ρ:

$$\rho = \lim_{n\to\infty} \frac{1}{n} E\left\{\sum_{i=1}^{n} r_{t_{d,i}}\right\} \qquad (12)$$

Through this reward scheme, the objective of minimizing the expected sum of the mean inventory level, plus the mean length of backorders (refer to section III), is conveyed to the agent. The agent achieves this objective by learning the *gain-optimal policy* π^*, i.e. a mapping from system states to control actions that maximizes the average reward: $\rho^{\pi^*}(\mathbf{s}) > \rho^{\pi}(\mathbf{s})$, $\forall$ policy π and $\forall$ state $\mathbf{s}$.

E. LEARNING ALGORITHM AND Q-VALUE TABLE

The joint maintenance and production control policy computed by the decision-making agent is stored in a *Q-value table*. An entry $Q(\mathbf{s}, c)$ of this table is an estimate of $Q^{\pi}(\mathbf{s}, c)$ which in turn is the "usefulness" of taking the control action c while being in state $\mathbf{s}$ under policy π:

$$Q^{\pi}(\mathbf{s}, c) = \sum_{j=1}^{\infty} E_{\pi}\left\{r_{t_{d,i+j}} - \rho^{\pi} | \mathbf{s}_{t_{d,i}} = \mathbf{s}, c_{t_{d,i}} = c\right\} \qquad (13)$$

The informal term "usefulness" refers to the expected sum of future rewards, adjusted by the average reward, when following the policy π. In equation (11), $r_{t_{d,i+j}}$ is the reward received by the agent for its $(i+j-1)$ – th action selection and ρ^{π} is the average reward under policy π. Furthermore, $\mathbf{s}_{t_{d,i}}$ and $c_{t_{d,i}}$ denotes the state and the control action that is selected by the agent at the i – th decision epoch.

The elements of the Q – value table are often referred to as *action values*. Selecting the control action with the highest action value (greedy action) in all states maximizes the expected average reward, provided that the true action values have been computed accurately.

Nonetheless, the true action values are not known initially, since the environment in which the agent is situated is initially unknown. The agent interacts with its environment and updates the action value estimates so as to eventually obtain the actual values of the Q-value table elements. In order to select the most appropriate learning algorithm for the problem addressed in this paper, the authors conducted pilot experiments using standard *model-free* methods namely, Schwartz's R-learning ([27]), variants of R-learning ([28]) and R-smart ([29]). The R-learning algorithm exhibited the best performance and was ultimately selected. According to the R-learning algorithm, the action value and average reward estimates are updated as follows (the subscripts that indicate time have been dropped for simplicity):

$$Q(\mathbf{s}, c) \leftarrow Q(\mathbf{s}, c) + a\left(r - \rho + Q\left(\mathbf{s}', c'\right) - Q(\mathbf{s}, c)\right) \qquad (14)$$

$$\rho \leftarrow \rho + \beta\left(r - \rho + Q\left(\mathbf{s}', c'\right) - Q(\mathbf{s}, c)\right) \qquad (15)$$

where $\leftarrow$ is the assignment operator, $\mathbf{s}$ and c is the state and the selected action in the current decision epoch, $\mathbf{s}'$ and r is the state and the received reward in the next decision epoch, ρ is the average reward, c' is the greedy action for state $\mathbf{s}'$ and finally, a and β are real-valued parameters. The average reward update in R-learning takes place only when the agent selects the action with the highest action value, whereas the Q-value update occurs in all decision epochs.

F. EXPLORATION STRATEGY

In order to obtain accurate Q-value estimates, the agent must try all admissible actions, in the system states that it "visits", sufficiently many times. The e-greedy technique is used in order to *explore* the state-action space effectively. According to the e-greedy exploration strategy, in a decision epoch $t_{d,i}$:

- with probability $1 - e$, the agent selects the greedy action

or

- with probability e, some action is selected randomly: $\Pr\left(c_{t_{d,i}} = c\right) = 1/card\left(A\left(\mathbf{s}_{t_{d,i}}\right)\right)$

where e is a real-valued parameter in the range (0, 1). Clearly, the higher the value of e is, the more often the agent will make exploratory moves.

VI. COMPUTATIONAL EXPERIMENTS

The behavior of the alternative parametric maintenance and production control policies, as well as that of the Reinforcement Learning based approach, was studied in seven simulation cases. All simulation cases share the following subset of input parameters: maximum allowed inventory $I_{max} = 10$, maximum allowed number of backorders $B_{max} = 10$ and number of deterioration stages $d = 6$. The remaining parameters that characterize the alternative simulation cases are summarized in table 4.

In simulation cases 1 – 7, the effect of varying the levels of the most important simulation parameters are examined, that is: i) the arrival rate, ii) the deterioration failure rate, iii) the mean duration of minimal maintenance, and

TABLE 4. Parameters of simulation cases. The definitions of the symbols are given in section iii.

	$1/\lambda_a$	$1/\lambda_p$	$1/\mu_r$	$1/\lambda_{f,i}$ $i = 0,\ldots, 6$	$1/\mu_{m,i}$ $i = 1,\ldots, 6$
case 1	1.5	1	30	(10, 9, 8, 7, 6, 5, 4)	(3, 4, 5, 6, 7, 8)
case 2	2	1	30	(10, 9, 8, 7, 6, 5, 4)	(3, 4, 5, 6, 7, 8)
case 3	2.5	1	30	(10, 9, 8, 7, 6, 5, 4)	(3, 4, 5, 6, 7, 8)
case 4	1.5	1	30	(20, 19, 18, 17, 16, 15, 14)	(3, 4, 5, 6, 7, 8)
case 5	1.5	1	30	(30, 29, 28, 27, 26, 25, 24)	(3, 4, 5, 6, 7, 8)
case 6	1.5	1	50	(10, 9, 8, 7, 6, 5, 4)	(3, 4, 5, 6, 7, 8)
case 7	1.5	1	30	(10, 9, 8, 7, 6, 5, 4)	(5, 6, 7, 8, 9, 10)

iv) the mean duration of repair. Case 1 is the *base* simulation case. In cases 2 and 3, the arrival rate is varied, while keeping all other parameters fixed, in order to investigate the effect of alternative average workload levels imposed on the production system. It should be noted that workload levels are mostly defined by the relative difference between the arrival rate and the service rate, so there is no need to vary explicitly λ_p. Cases 4 and 5 differ from the base case in respect to the frequency of the deterioration failures occurrences. Simulation cases 6 and 7 correspond to lengthier repair and maintenance activities, respectively, as compared to the base case.

A. CONFIGURATION OF PARAMETRIC CONTROL POLICIES AND RL-BASED APPROACH

The proposed approach for integrated production and maintenance control was compared to:

- the Kanban system with condition based maintenance (Kanban - CBM)
- the Kanban system with periodic maintenance (Kanban - PM)
- the (s, S) system with condition based maintenance ((s, S) - CBM)
- the (s, S) system with periodic maintenance ((s, S) - PM)

In the four aforementioned parametric control policies, the maintenance component overrides the production control component, similarly to what happens in many practical situations. For example, in a Kanban – PM system, if there is a pending production authorization for an end – item at the time when a minimal maintenance has been scheduled, then the maintenance activity is given higher priority and precedes the production operation.

In order to compare the alternative approaches on the same basis, the best parameters for each simulation case and maintenance/production control method need to be obtained. The parameter levels that were probed for the purposes of this simulation study are summarized in table 5.

Note that in all simulation cases, the maximum allowed inventory is $I_{max} = 10$ and consequently, by definition of the Kanban and (s, S) control policies, $K = 10$ and $S = 10$, respectively. All feasible values of parameter b for the control policies with condition based maintenance are considered, since the number of deterioration stages $d = 6$ for all simulation cases.

TABLE 5. Parameter space of alternative maintenance/production control methods.

Kanban - CBM	$K = 10$ $b = 1, 2, 3, \ldots, 6$
Kanban - PM	$K = 10$ $TBM = 5, 10, 15, \ldots, 50$ ($\times 1/\lambda_a$)
(s, S) - CBM	$S = 10$ $s = 0, 1, 2, \ldots, 5$ $b = 1, 2, 3, \ldots, 6$
(s, S) - PM	$S = 10$ $s = 0, 1, 2, \ldots, 5$ $TBM = 5, 10, 15, \ldots, 50$ ($\times 1/\lambda_a$)
RL	$e = 0.5, 0.3, 0.1$ $a = 0.005, 0.02, 0.1$ $\beta = 0.005, 0.02, 0.1$

The search space for the remaining parameters reported in table 5 was set by conducting pilot experiments and by adhering to guidelines suggested in the relevant literature. In table 5, $1/\lambda_a$ refers to the mean time between arrivals of the respective simulation case. For example, in simulation case 2 the search space for parameter *TBM* is 10, 20, 30, ..., 100. Finally, it is reiterated that the RL parameters are the exploration probability e, and the R-learning parameters a and β. From table 5 it can be seen that the total number of experiments that were conducted is: [6 (Kanban - CBM) + 10 (Kanban - PM) + 36 ((s, S) - CBM) + 60 ((s, S) - PM) + 27 (RL)] × 7 (simulation cases) = 973 simulation experiments/models.

B. PARAMETERS OF SIMULATION EXPERIMENTS

10 independent replications of each simulation model configuration were executed, where each replication ran up to the point where 4.5 million end-items were completed in the manufacturing facility so as to assure that the system had reached *steady state*. The output of each simulation model configuration, i.e. average inventory and backorders level, was averaged over all its replications.

The RL-based approach was evaluated off-line, i.e. for each simulation case there was an additional *training replication*, whose duration also corresponded to the completion of 4.5 million end-items, where the agent simply derived a control policy and no statistics were monitored. In the training replications, all elements of the Q- table were set to 0 initially and the exploration probability e (refer to section V for details) was held constant throughout the simulation. The RL-based control policy that was computed in the training replication was then evaluated in 10 replications, similarly to the alternative parametric maintenance and production control policies.

In the evaluation phase, the decision-making agent does not make explorative decisions ($e = 0$) nor updates its action selection policy, in order to ensure a fair comparison among the alternative approaches. Furthermore and on the same grounds, the same random number streams, in replications with the same index, were used for all alternative maintenance/production control approaches. The simulation models, as well as the RL agent, were coded in standard C++

and the experiments were carried out on a PC with 64-bit Windows 7 OS, 3.4 GHz CPU and 4 GB RAM.

C. OVERVIEW OF EXPERIMENTAL RESULTS

In fig. 4, the performance of the alternative maintenance/production control schemes is summarized.

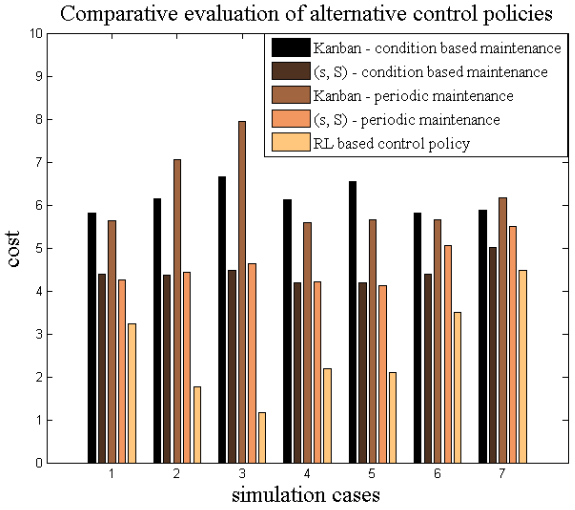

FIGURE 4. Lowest total cost of alternative maintenance/production control approaches.

The height of the bars corresponds to the lowest objective function value (refer to section III) attained by each approach. The control policies computed by the RL-based agent clearly outperform all parametric control policies in all simulation cases. This indicates the potential of applying Machine Learning methods to complex optimization problems from the field of industrial engineering. Moreover, it is evident that ad hoc control policies are rather far from being optimal in this setting. The RL-based agent selects control actions by explicitly taking into account the synergy between production and maintenance decisions and this highlights the benefits of following an integrated maintenance/production policy that has been computed, based on the feedback of the controlled system.

Regarding the parametric policies, it is observed that the (s, S) – CBM and (s, S) – PM systems outperform their *pull type* counterparts, namely Kanban – CBM and Kanban – PM in this series of experiments. No definitive conclusions can be reached regarding the maintenance policy type, since condition based maintenance seems to yield better results in some cases and worse in others, compared to periodic maintenance. The best parameters of the alternative maintenance/production control approaches are given in table 6. The effect of the parameter values on the performance of the various ad hoc control policies is detailed in the following section VI.D.

D. ANALYSIS OF PARAMETRIC PRODUCTION AND MAINTENANCE POLICIES

In this section, and based on the results obtained by the simulation experiments, we discuss the properties of the cost function for the alternative parametric production/maintenance control policies. Furthermore, the effect of the respective control parameters (b, s, TBM) on the cost function is examined. It should be noted that, the total number of conducted experiments pertaining to parametric policies is 784. Therefore, in order to save space and to prevent figures 5-8 from becoming cluttered, the analysis is limited to an indicative subset of the experimental results. It is underlined that this section is not meant to be an exhaustive

TABLE 6. Best parameters of alternative maintenance/production control methods.

	Kanban – CBM	(s, S) - CBM	Kanban - PM	(s, S) - PM	RL
case 1	$b = 2$	$s = 0$ $b = 1$	$TBM = 7.5$	$s = 5$ $TBM = 7.5$	$e = 0.5$ $a = 0.02$ $\beta = 0.005$
case 2	$b = 4$	$s = 0$ $b = 1$	$TBM = 10$	$s = 0$ $TBM = 10$	$e = 0.5$ $a = 0.02$ $\beta = 0.005$
case 3	$b = 5$	$s = 0$ $b = 3$	$TBM = 75$	$s = 0$ $TBM = 12.5$	$e = 0.3$ $a = 0.005$ $\beta = 0.02$
case 4	$b = 5$	$s = 0$ $b = 1$	$TBM = 7.5$	$s = 0$ $TBM = 15$	$e = 0.3$ $a = 0.02$ $\beta = 0.005$
case 5	$b = 5$	$s = 0$ $b = 1$	$TBM = 7.5$	$s = 0$ $TBM = 15$	$e = 0.1$ $a = 0.005$ $\beta = 0.005$
case 6	$b = 2$	$s = 0$ $b = 1$	$TBM = 7.5$	$s = 0$ $TBM = 7.5$	$e = 0.5$ $a = 0.1$ $\beta = 0.005$
case 7	$b = 1$	$s = 0$ $b = 1$	$TBM = 15$	$s = 0$ $TBM = 75$	$e = 0.1$ $a = 0.005$ $\beta = 0.005$

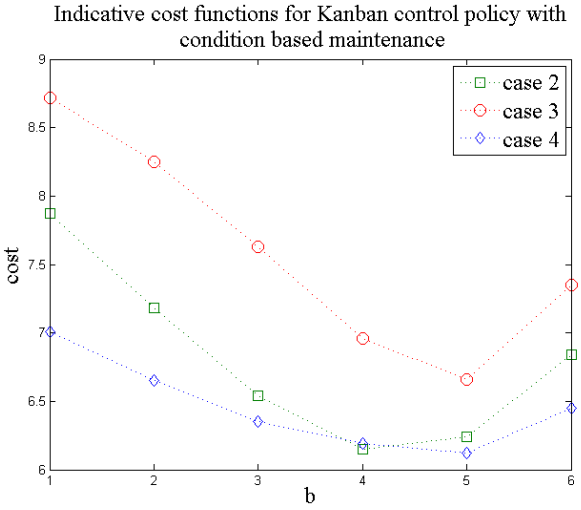

FIGURE 5. Expected cost for varying levels of b for the Kanban – CBM policy.

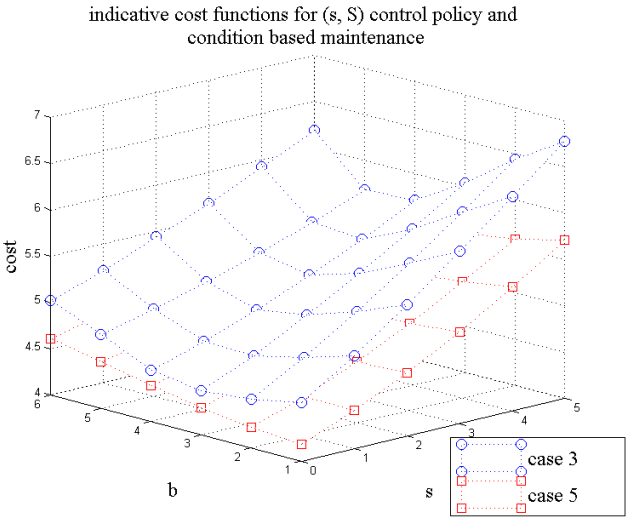

FIGURE 6. Expected cost for varying levels of b and s, for the (s, S) – CBM policy.

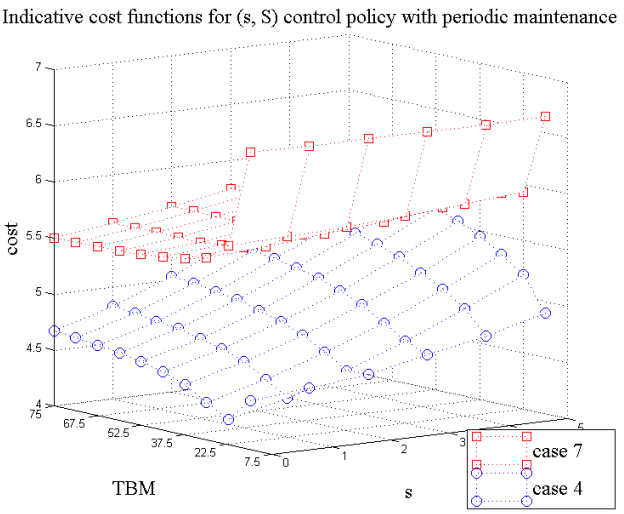

FIGURE 8. Expected cost for varying levels of *TBM* and s for the (s, S) – PM policy.

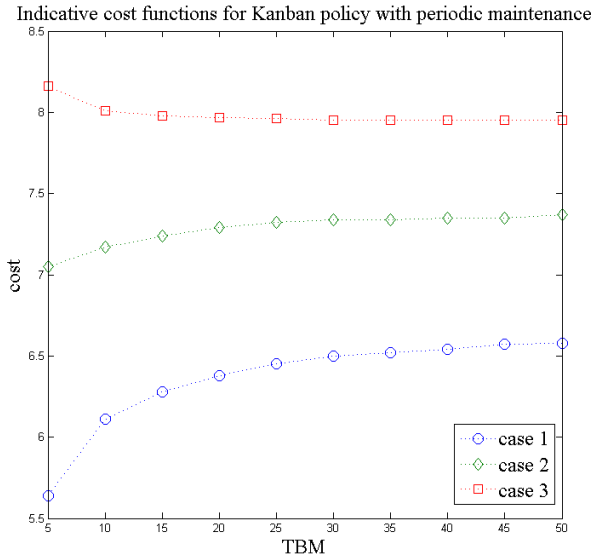

FIGURE 7. Expected cost for varying levels of *TBM* for the Kanban – PM policy.

investigation of the various properties of the policies, because this is beyond the scope of this paper. Nevertheless, useful insights regarding the behavior of the examined parametric production/maintenance policies can be gained from the analysis of this section.

Fig. 5 shows the cost curves in respect to parameter b of the Kanban – CBM policy for simulation cases 2, 3, and 4. It can be seen that the objective function is convex in respect to b and that the positioning of the function curve in the 2D plane, as well as the location of the unique minimum, depends on the simulation case.

It is reiterated that b is the deterioration threshold for conducting minimal maintenance of the manufacturing facility and that the mean duration of maintenance activities rises with the deterioration level. Therefore, a relatively small b means that the production system is maintained frequently for rather short periods. On the other hand, a relatively high b leads to less frequent but lengthier maintenance activities. Maintenance prevents hard failures and so, by increasing the availability and throughput of the system, the mean length of backorders is decreased. However, during maintenance production is ceased and therefore a manufacturing facility down-time is also incurred. The optimal value for b is the one that resolves the aforementioned trade-off effectively.

Fig. 6 shows the cost surfaces in respect to b and s parameters of the (s, S) – CBM policy for simulation cases 3 and 5. Here, the cost is an increasing function of parameter s, or equivalently, the minimum end-item inventory level targeted by the (s, S) production control policy. The higher the parameter s is, the more time the manufacturing facility is in a working state, resulting in increased inventory levels and holding costs. However, note that in cases 3 and 5, a relatively moderate workload is imposed on the production system and consequently, the holding cost component overshadows the backorders cost component. On the other hand, in the heavy workload simulation case 1, the emphasis is on minimizing the mean length of the backorders queue, and consequently the optimal value of parameter s is 5 (refer to table 6).

The optimal b for cases 3 and 5 is found to be 3 and 1, respectively. This can be attributed to the fact that the deterioration failure rate is lower in case 5, as compared to that of case 3, thus less frequent maintenance actions are needed. This allows for parameter b to be set to the lowest possible value, in order to keep the average duration of the maintenance activities, along with the resulting down time, at low levels.

Fig. 7 shows the cost curves in respect to parameter *TBM* of the Kanban – PM policy for simulation cases 1 – 3. It is observed that the cost is an increasing function of *TBM* in cases 1, 2 and a decreasing function of *TBM* in simulation case 3.

In all three cases, the cost function tends to level after some point. It should be noted that the demand arrival rate decreases from case 1 to case 3 (refer to table 4). Therefore, in e.g. case 1, there is a relatively high arrival rate, and since Kanban is a pull type policy, the manufacturing facility is frequently authorized to produce. Consequently, the production system spends a large fraction of time in the working state and deteriorates relatively quickly. As a result, frequent periodic maintenance is needed to prevent hard failures or lengthy minimal maintenance activities, due to increased deterioration levels. This explains the shape of the cost curve for simulation case 1 and similar arguments can be also made for case 2. On the contrary, in simulation case 3 the arrival rate is relatively low and so, the actual manufacturing facility deterioration rate is also low. It follows that the time between successive maintenance activities should be sufficiently high, in order to avoid unnecessary down time of the manufacturing facility due to maintenance.

Fig. 8 shows the cost surfaces in respect to *TBM* and s parameters of the (s, S) – PM policy for simulation cases 4 and 7. Note that in case 4, the deterioration failure rate and the minimal maintenance rate is lower and higher, respectively, in relation to simulation case 7.

This is the reason why the expected cost is generally higher in case 7, as compared to that of case 4. In case 7, the mean duration of minimal maintenance, especially in high deterioration stages, is rather comparable to the mean duration of repair. Furthermore, minimal maintenance restores the system to the as-bad-as-before state, whereas repair restores the system to the as-good-as-new state. As a result, in this case it might be preferable to occasionally let the system experience a hard failure, rather than subject it frequently to maintenance activities, and this explains the best value of *TBM* for this simulation case. On the other hand, the best value of parameter *TBM* for case 4 resolves the trade-off between keeping the deterioration of the facility low and not incurring high down times due to maintenance activities. The cost is an increasing function of parameter s in both case 4 and 7, and the rationale for that is similar to the related analysis of the (s, S) – CBM policy for cases 3 and 5 in a previous point of this section.

E. ANALYSIS OF RL-BASED JOINT PRODUCTION/MAINTENANCE CONTROL POLICIES

In this section, some remarks are made on indicative integrated production/maintenance control policies, derived by the RL-based decision-making agent. Note that in all simulation cases of this research, $I_{max} = B_{max} = 10$, $d = 6$ and consequently, the Q-value table of the agent consists of 146 states, as shown in equation (7). By taking into account the admissible controls function (section V), it can be easily inferred that the Q – value table contains 20 (states) $\times$ 2 (available actions) + 6 (states) $\times$ 2 (available actions) + 120 (states) $\times$ 3 (available actions) = 412 action values.

The control policies computed by the decision-making agent are rather complex and largely differ from one simulation case to another. Nonetheless, some general features of the policies seem to uphold across some simulation cases. In order to provide a qualitative interpretation of the RL-based control policies, an indicative depiction is shown for the ones pertaining to simulation cases 1 and 3 in fig. 9.

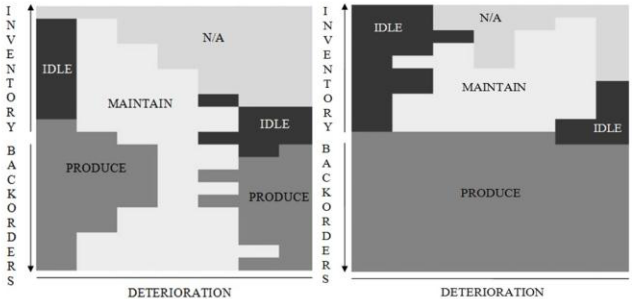

FIGURE 9. Control policies obtained by the decision-making agent in case 3 (on the left) and in case 1 (on the right).

In fig. 9, areas marked with "N/A" correspond to states not visited by the decision-making agent during the training replication and consequently, the related elements of the Q-table are fixed to their initial, arbitrarily selected, values. Note that in Reinforcement Learning, for a Q-value to be updated, the state pertaining to it must be visited during the simulation execution. This is a favorable trait of RL, because action values are updated only for situations that are actually likely to occur, thus reducing the computational overhead. It is not known beforehand which states are going to be visited, because state transitions depend on the policy followed by the agent, which in turn is dynamically updated during the learning process.

It is observed that the RL-based joint production/maintenance policies for cases 1 and 3 are complex and that they cannot be easily characterized by monotone switching curves. This can be primarily attributed to the complexity of the underlying optimization problem. Nonetheless, it is noted that these control policies are *approximations* of the respective optimal policies, because they are constructed on the basis of *sample paths* of the production system dynamic evolution generated by discrete-event simulation. Consequently, some degree of statistical error is incorporated in the computation of the respective Q-values and might also be reflected into the derived policies.

Simulation cases 1 and 3 differ only in terms of the arrival rate. In case 1 and 3 the production system is under relatively heavy and moderate workload, respectively. Therefore, in broad terms, the decision-making agent opts to authorize production in case 1 if the inventory position is under some level, regardless of the manufacturing facility deterioration stage, in order to meet the rapidly incoming demand of end-items. If the inventory position is over some level, the agent chooses to either maintain the facility or stop production to avoid the build-up of excessive inventory. In that case, the agent authorizes minimal maintenance if the facility is "amply" but not "too" deteriorated. This policy can be

interpreted as follows: when the manufacturing facility is highly deteriorated, minimal maintenance is relatively prolonged and only restores the system to the as-bad-as-before state, making it less preferable than repair, which renders the system good-as-new. On the other hand, if the facility deterioration is low, the chances of a hard failure are small and maintenance is not advised, because of the incurred down time.

The rationale of the control policy computed by the RL agent in case 3 is somewhat similar to that of case 1, with one major exception: the agent can authorize a minimal maintenance even if there is backordered demand in the production/inventory system. This is because the system is under moderate workload in case 3 and so production can easily keep up with incoming demand. Moreover, in this case, prevention of hard failures compensates for the down time incurred by minimal maintenance activities.

VII. CONCLUSIONS AND DIRECTIONS FOR FUTURE RESEARCH

The problem of integrated production/maintenance control for a deteriorating, stochastic production/inventory system was investigated. A novel approach, based on Reinforcement Learning, for deriving optimal or near-optimal policies was proposed. The Reinforcement-Learning based approach was compared to several ad hoc production and maintenance policies that are widely being used in practice. These ad hoc control policies were found to be suboptimal in all simulation cases examined in this research. Their performance depends largely on the values of the respective control parameters. The application of Reinforcement Learning for solving this complex industrial engineering problem yielded substantially encouraging results. Furthermore, the results showcased the merits of integrated production/maintenance policies that explicitly account for interactions between maintenance and production decisions.

This research can be extended by considering more complex production system configurations, e.g. manufacturing lines, and alternative objective functions. In the former case, due to the significantly increased size of the state space multi-agent system architectures might be mandated. Other directions for future research include the application of Reinforcement Learning for solving alternative industrial engineering problems, for example integrated production and quality control of manufacturing systems that produce imperfect end-items.

REFERENCES

[1] J. E. Eloy Ruiz-Castro, "Preventive maintenance of a multi-state device subject to internal failure and damage due to external shocks," *IEEE Trans. Rel.*, vol. 63, no. 2, pp. 646–660, Jun. 2014.

[2] Y. Cai, J. J. Hasenbein, E. Kutanoglu, and M. Liao, "Single-machine multiple-recipe predictive maintenance," *Probab. Eng. Inf. Sci.*, vol. 27, no. 2, pp. 209–235, 2013.

[3] M. S. Fallahnezhad, "A finite horizon dynamic programming model for production and repair decisions," *Commun. Statist.-Theory Methods*, vol. 43, no. 15, pp. 3302–3313, 2014.

[4] N. Li, F. T. S. Chan, S. H. Chung, and A. H. Tai, "A stochastic production-inventory model in a two-state production system with inventory deterioration, rework process, and backordering," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 6, pp. 916–926, Jun. 2017.

[5] E. G. Kyriakidis, "Equilibrium probabilities for a production-inventory system maintained by a control-limit policy," *Commun. Statist.-Theory Methods*, vol. 45, no. 1, pp. 194–200, 2016.

[6] A. Wolter and S. Helber, "Simultaneous production and maintenance planning for a single capacitated resource facing both a dynamic demand and intensive wear and tear," *Central Eur. J. Oper. Res.*, vol. 24, no. 3, pp. 489–513, 2016.

[7] N. Nahas, "Buffer allocation and preventive maintenance optimization in unreliable production lines," *J. Intell. Manuf.*, vol. 28, no. 1, pp. 85–93, 2017.

[8] X. Wang, H. Wang, and C. Qi, "Multi-agent reinforcement learning based maintenance policy for a resource constrained flow line," *J. Intell. Manuf.*, vol. 27, no. 3, pp. 325–333, 2016.

[9] Z. Hajej, S. Turki, and N. Rezg, "Modelling and analysis for sequentially optimising production, maintenance and delivery activities taking into account product returns," *Int. J. Prod. Res.*, vol. 53, no. 15, pp. 4694–4719, 2015.

[10] B. Kader, D. Sofiene, R. Nidhal, and E. Walid, "Ecological and joint optimization of preventive maintenance and spare parts inventories for an optimal production plan," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 2139–2144, 2015.

[11] G.-L. Liao, "Production and maintenance policies for an EPQ model with perfect repair, rework, free-repair warranty, and preventive maintenance," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 8, pp. 1129–1139, Aug. 2016.

[12] K. He, L. M. Maillart, and O. A. Prokopyev, "Scheduling preventive maintenance as a function of an imperfect inspection interval," *IEEE Trans. Rel.*, vol. 64, no. 3, pp. 983–997, Sep. 2015.

[13] N. Li, F. T. S. Chan, S. H. Chung, and A. H. Tai, "An EPQ model for deteriorating production system and items with rework," *Math. Problems Eng.*, vol. 2015, 2015, Art. no. 957970, doi: 10.1155/2015/957970.

[14] B. Jafary, V. Nagaraju, and L. Fiondella, "Impact of correlated component failure on preventive maintenance policies," *IEEE Trans. Rel.*, vol. 66, no. 2, pp. 575–586, Jun. 2017.

[15] X. Zhang and J. Zeng, "Joint optimization of condition-based opportunistic maintenance and spare parts provisioning policy in multiunit systems," *Eur. J. Oper. Res.*, vol. 262, no. 2, pp. 479–498, 2017.

[16] S. Zhao, L. Wang, and Y. Zheng, "Integrating production planning and maintenance: An iterative method," *Ind. Manage., Data Syst.*, vol. 114, no. 2, pp. 162–182, 2014.

[17] A. S. Xanthopoulos, D. E. Koulouriotis, and P. N. Botsaris, "Single-stage Kanban system with deterioration failures and condition-based preventive maintenance," *Rel. Eng., Syst. Safety*, vol. 142, pp. 111–122, Oct. 2015.

[18] X. Yao, X. Xie, M. C. Fu, and S. I. Marcus, "Optimal joint preventive maintenance and production policies," *Naval Res. Logistics*, vol. 52, no. 7, pp. 668–681, 2005.

[19] D. Chen and K. S. Trivedi, "Closed-form analytical results for condition-based maintenance," *Rel. Eng., Syst. Safety*, vol. 76, no. 1, pp. 43–51, 2002.

[20] T. K. Das and S. Sarkar, "Optimal preventive maintenance in a production inventory system," *IIE Trans.*, vol. 31, no. 6, pp. 537–551, 1999.

[21] S. M. R. Iravani and I. Duenyas, "Integrated maintenance and production control of a deteriorating production system," *IIE Trans.*, vol. 34, no. 5, pp. 423–435, 2002.

[22] J. Geraghty and C. Heavey, "An investigation of the influence of coefficient of variation in the demand distribution on the performance of several lean production control strategies," *Int. J. Manuf. Technol. Manage.*, vol. 20, nos. 1–4, pp. 94–119, 2010.

[23] S. Axsäter, *Inventory Control*. New York, NY, USA: Springer, 2015.

[24] A. S. Xanthopoulos, D. E. Koulouriotis, V. D. Tourassis, and D. M. Emiris, "Intelligent controllers for bi-objective dynamic scheduling on a single machine with sequence-dependent setups," *Appl. Soft Comput.*, vol. 13, no. 12, pp. 4704–4717, 2013.

[25] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, vol. 1. Cambridge, MA, USA: MIT Press, 1998.

[26] A. S. Xanthopoulos, D. E. Koulouriotis, A. Gasteratos, and S. Ioannidis, "Efficient priority rules for dynamic sequencing with sequence-dependent setups," *Int. J. Ind. Eng. Comput.*, vol. 7, no. 3, pp. 367–384, 2016.

[27] A. Schwartz, "A reinforcement learning method for maximizing undiscounted rewards," in *Proc. 10th Int. Conf. Mach. Learn.*, 1993, pp. 298–305.

[28] S. Singh, "Reinforcement learning algorithms for average-payoff Markovian decision processes," in *Proc. 12th Nat. Conf. Artif. Intell.*, 1994, pp. 202–207.
[29] A. Gosavi, "A reinforcement learning algorithm based on policy iteration for average reward: Empirical results with yield management and convergence analysis," *Mach. Learn.*, vol. 55, no. 1, pp. 5–29, 2004.

A. S. XANTHOPOULOS received the M.S. and Ph.D. degrees in production and management engineering from the Democritus University of Thrace, Xanthi, Greece, in 2006 and 2010, respectively.

Since 2010, he has been a Post-Doctoral Researcher and an Adjunct Lecturer with the Department of Production and Management Engineering, Democritus University of Thrace, Xanthi, Greece. He has authored one book, four book chapters, and over ten articles. His research interests include stochastic manufacturing systems, production/inventory control, dynamic sequencing, maintenance scheduling, discrete-event simulation, Markov chains, evolutionary algorithms, and reinforcement learning.

ATHANASIOS KIATIPIS received the B.Sc. degree in mechanical engineering from the Piraeus University of Applied Sciences, Greece, in 2011, and the M.Sc. degree in computational fluid dynamics from Cranfield University, England, in 2014, where he received an AeroMSc Scholarship from the Royal Academy of Engineering.

He is currently with Fujitsu Technology Solutions GmbH, Munich, Germany. His current research interests include data management, data storage, machine learning, artificial intelligence, and manufacturing systems. He is currently a holder of a Marie-Curie EU H2020 ITN Scholarship.

D. E. KOULOURIOTIS received the Diploma (combined B.Sc. and M.Sc.) degree in electrical and computer engineering from the Democritus University of Thrace, Greece, in 1993, the M.Sc. degree in electronic and computer engineering and the Ph.D. degree in production and management engineering from the Technical University of Crete, Greece, in 1996 and 2001, respectively.

He is currently a Full Professor with the Department of Production and Management Engineering and the Director of the Industrial Production Laboratory, Democritus University of Thrace, Greece. He has authored two books and numerous articles. His research interests include intelligent systems, industrial and management engineering, machine vision and signal processing, system safety, and business intelligence.

SEPP STIEGER received the Diploma degree in electrical engineering from the Technical University of Munich in 1997. He performed research, while stationed in Silicon Valley (USA), seeking the next generation PC characteristics, contributing to consortia which gave birth to 802.11 Wireless Networks and Bluetooth.

He is currently a Principal Consultant and also a Product Manager with Fujitsu Technology Solutions GmbH, for various ETERNUS Storage products. Along with the professional experience in the technology sector where he investigates, together with partners, new products and markets, he also has research interests in the area of big data, manufacturing technologies (KANBAN), and data storage.