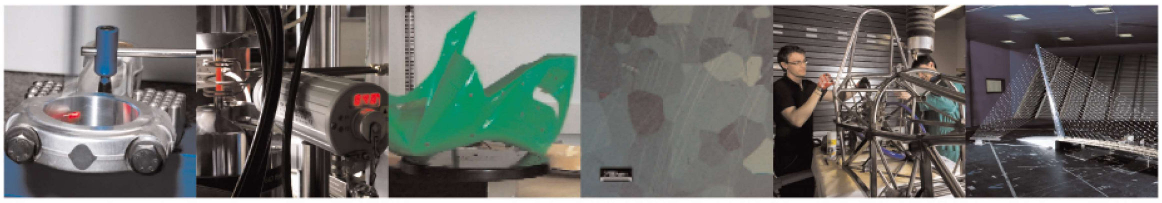
POLITECNICO
MILANO 1863

DIPARTIMENTO DI MECCANICA

mecc



An Online Policy for Energy-Efficient State Control of Manufacturing Equipment

Frigerio, N.; Marzano, L.; Matta, A.

This is a post-peer-review, pre-copyedit version of an article published in IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING. The final authenticated version is available online at: http://dx.doi.org/10.1109/TASE.2020.3044107

DIPARTIMENTO DI MECCANICA ■ POLITECNICO DI MILANO
via G. La Masa, 1 ■ 20156 Milano ■ EMAIL (PEC): pecmecc@cert.polimi.it
http://www.mecc.polimi.it
Rev. 0

1

An On-Line Policy for Energy-Efficient State Control of Manufacturing Equipment

Nicla Frigerio, Lorenzo Marzano, and Andrea Matta

***Abstract*—Machine state control is one of the most promising energy efficient measures for machining processes. A proper control reduces the energy consumed during idle periods by switching off/on the machines. A critical barrier for practical implementation is related to the knowledge of part arrival process that is affected by uncertainty. The stochastic processes involved in the system are usually assumed to be known. However, real production environments are subject to several sources of randomness that are difficult to model a priori. This work provides an on-line time-based algorithm able to control the machine state. Through a method for the estimation of the stochastic process, the algorithm provides the optimal control parameters based on a collected set of observations. A new policy is formulated to manage the control over time such that changes in the control parameters are applied only under certain conditions. Potential benefits are discussed by means of realistic numerical cases.**

***Note to Practitioners*—The paper analyzes the control problem of switching off/on a machine tool for energy saving during machine idle periods. A control policy based on time information is investigated when the machine requires a startup time to resume the service after being switched off. The proposed policy works on-line while acquiring information from the real system. An algorithm is described for identifying and applying the optimal control parameters. The results of this research will be useful for a practical implementation of a switching policy for energy saving. This implementation requires the estimation of the power adsorbed by the machine in four different states and, therefore, it reduces the implementation effort for practitioners.**

***Index Terms*—Energy efficiency, optimal control, machine learning.**

I. Introduction

ENERGY efficiency is a key issue in the manufacturing field and real-time strategies that switch off/on the resources in production systems have been recently proposed in the literature. These strategies apply when a resource, mostly a machine tool, is idle such that some machine components could be switched off. A machine partially switched off cannot start a new process and the service cannot be resumed until all machine components are switched on. Commonly, a startup procedure that uses both time and energy is required to resume the service. The control problem is not trivial and control parameters should be selected properly to assure effectiveness of the control and, simultaneously, not to jeopardize the production rate. Moreover, since manufacturing systems are affected by uncertainty, the arrivals at a certain machine follow a certain stochastic process.

A critical barrier for a practical implementation is related to the knowledge of such stochasticity. A large amount of data should be collected to obtain a proper fit of the arrival distribution at a certain machine. Further, this data collection must be performed at each machine and it should be repeated frequently, for instance, after maintenance interventions on the system or changing in the production planning. As a result, the implementation of energy control policies might have a complex and costly management. In common practice, energy efficient state control policies are rarely used and the machine tools are mostly kept ready-for-process. Nevertheless, in the industrial market there are several energy control systems (i.e., PLC embedded or external devices), but the selection of the control parameters is manual and experience-based, which is risky in terms of unexpected high energy consumption.

This work[1] deals with the problem of controlling a machine on-line while acquiring data such that the control policy extracts knowledge from environment conditions. Nevertheless, the decision maker should take into account the risk of incurring in unexpected costs. Indeed, as the acquisition is on-line, the control problem is solved based on estimated parameters and the risk of implementing a control on a biased estimate might be high. A proper amount of information should be collected before implementing the control. In the proposed policy, the control is applied only if the estimated advantage is significant considering the risk of incurring in unexpected high energy consumption.

A. Energy-Efficient Control at Machine Level

The energy efficient control (EEC) of machine states addresses the problem of energy efficiency at machine level focusing onto the reduction of the *non processing energy* (NPE). This energy is usually denoted as fixed energy or base load and it is related to the power requests of machine auxiliary systems that keep executing their functions although the machine is not producing [1]. Auxiliaries allow to keep the machine in ready-for-process conditions such that, at part arrival, the process can immediately start. NPE is separated from the *processing energy* that is required while the machine tool is working on parts.

When a machine tool starts executing the process, it passes from an *idle* state to a *busy* state; then, at process completion, the machine returns idle until the next cycle starts. These

N. Frigerio, L. Marzano and A. Matta are with the Mechanical Engineering Department, Politecnico di Milano, 20156 Milano, ITA. Contact e-mail: nicla.frigerio@polimi.it, and andrea.matta@polimi.it.

[1]This paper was presented in part in IEEE CASE 2019 in Vancouver, Canada

transitions represent the common behavior of a not controlled machine tool under the *Always On* (AO) policy.

EEC applies when the machine is idle: a switching off command triggers the machine in a low energy consumption state, i.e., a *standby* or *sleeping* state, and the service is interrupted. Machine auxiliaries and peripheral units (e.g., the hydraulic unit, the chillers, the chip conveyor) become inactive and the machine power request is reduced. With a switching on command, the service is resumed. It is noteworthy that machines need to visit a *startup* transitory state to resume the service such that the quality of parts is guaranteed. Indeed, all machine components should be active at the beginning of the process to enable machine functionality. For instance, chiller assures the necessary thermal condition to avoid misalignment and distortions of machine structure.

The EEC problem belongs to the last control level in the production planning & control hierarchy. In the literature, it is sometimes referred as *real–time* control to be distinguished from scheduling problems. A recent and complete review on energy efficient scheduling literature can be found in Gahm et al. [2]. These scheduling problems belong to a different level of hierarchy compared to EEC. Energy efficient scheduling plans off/on modes over a specific period of time given the jobs assigned to machines. Whereas, EEC literature provides policies at machines during production progress without knowing when the next part arrives.

A first group of studies analyses machine EEC problems where the service is interrupted and resumed based on time information ([3],[4],[5],[6],[7],[8]). Under a time-based control, machines are controlled during starvation periods, i.e., waiting for parts. A second group analyses machines where the service is interrupted and resumed based on the number of parts accumulated in buffers such that machines can be controlled during starvation periods ([9]), blocking periods, or both ([10],[11],[12],[13]). A combination of time and buffer information is used in a third group of studies ([14],[15],[16],[17]). The more information used, the more complex the control. At system level, the information coming from one or more buffers are used to control machines.

The optimal control problem incorporates an energy efficiency criterion in the objective function, a minimum production rate target as second objective ([8]) or as a constraint ([3], [4], [8], [9], [11], [15], [17]). Also, several works consider the energy that might be required to hold parts waiting for resource availability ([3], [4], [9], [17]).

The state model used to represent controlled resources might have different assumptions. All works mentioned in this section consider the startup state, although only a subset considers also a closedown transitory between the idle and the standby state ([8], [13], [15]). The closedown and startup times are considered either stochastic or deterministic. The deterministic assumption of transitory duration is effective for a wide number of practical applications. The processing times at machines can be considered deterministic or stochastic, and failures can be modeled as randomness onto processing times. Machine starvation time depends onto the arrival process and it is therefore affected by uncertainty, e.g., random processing times and failures of upstream machines.

B. Paper Contribution

All EEC problems proposed in the literature assume that the stochastic processes involved in the control problem are known. Therefore, it is assumed implicitly that a large amount of data has been collected previously to fit probability distributions. Although off-line policies only require the fitting of arrival distribution, their application to many resources can require high effort. Also, the data collection should be frequently repeated in case of production changes. As an exception, a study has been recently proposed where the on-line EEC problem is firstly addressed [19] by fitting the parameters of a known distribution family.

This paper provides an innovative algorithm for the energy control of machine while learning from an on-line data collection about part arrivals. The work extends a state-of-the-art off-line EEC policy proposed and analyzed by Frigerio and Matta [3] enabling its on-line application. In more details, the algorithm includes:

- A new estimation model to predict the energy consumption per part produced;
- A new policy to identify when a change in control parameters becomes advantageous.

A learning method is used for parameter estimation, and the optimization problem is solved in real time to minimize the predicted energy consumption per part. As acquired data increases, the algorithm improves the solution. The challenge is to decide when the learning phase is sufficiently enough to implement or not the optimal parameters. Indeed, the method should take into account the risk of implementing a policy with wrong learned parameters.

In this work, we specifically refer to machine tool executing machining operations, but the approach can be applied to other machine types. The proposed algorithm learns autonomously from collected data reducing the implementation effort for practitioners. Also, as shown in the numerical results, the machine control is profitably applied after few collected data, i.e., before the off-line approach can be applied.

C. Paper Outline

The paper is divided into eight sections. After the introduction (Section I), section II is dedicated to the description of off-line EEC policies from the literature. Section III introduces the on-line EEC problem with the new estimation model. Section IV compares off-line and on-line policies to show the potential benefit of the on-line approach. In section V, a new policy is proposed to manage the implementation of control parameters over time. Numerical results follow in Section VI. Section VII includes an investigation about how the policy performs at system level, and section VIII concludes the work. Numerical results have been obtained in Matlab environment.

II. Off-Line Time-Based Control Policies for Energy Efficiency

OFF-LINE approaches are based on the assumption of having complete knowledge of the problem. In other words, it is assumed that problem parameters are known such that the off-line approach is required to directly output the problem solution. In the specific case of energy state control, problem parameters include machine parameters as well as the stochastic processes involved. In this section, we describe a general time-based off-line EEC policy available in the literature and the related control problem.

A. System Description and Assumptions

A single machine working a single part type is considered. Machine can fail and processing times are assumed to be random variables with mean t_p. Further, we assume that the machine might be starving of raw parts and is never blocked. Let us consider a *cycle* as the time interval starting from the departure of a part and the departure of the next one. The cycle starts at $t = 0$ with the machine waiting for the part arrival. It is assumed that an upstream mechanism manages the arrival process at machine and it is affected by stochasticity such that machine idle times X are distributed accordingly to a probability density function (pdf) $f(x)$ with mean t_a. In off-line policies, the pdf $f(x)$ is assumed to be known. Denote x the realization of X; hence, x is also the arrival time realization in a cycle.

B. Control Policy and Energy Consumption Model

We model machine states as represented in Figure 1. The following states are considered: *standby* ($s = 1$), *startup* ($s = 2$), *idle* ($s = 3$), and *busy* ($s = 4$). The machine is *busy* while working on parts. The *idle* state represents machine in ready-for-work conditions. The *standby* is a state where some machine components are not active such that the machine requires a low amount of power although part process cannot start. The *startup* state represents the transitory state to pass from the standby to the idle state. While in startup, the control activates machine components to achieve the proper working condition. We assume the startup procedure requires a deterministic duration t_{su}.

The *Switching* (SP) control policy [3] uses time-based thresholds τ_{off} and τ_{on} to control machine state:

> *Switch-off when τ_{off} has elapsed from the last departure. Switch-on when $\tau_{on} > \tau_{off}$ has elapsed from the last departure or when next part arrives.*

Controlled transitions are: (i) the switch-off transition from the idle state to the off state when $t = \tau_{\text{off}}$, and (ii) the switch-on transition from the off state to the startup state when $t = \tau_{\text{on}}$ or upon part arrival $t = x$. As in the literature [3], the SP policy is effective, despite its simplicity. It allows to delay the switch-off command when the probability of part arrival is high, and to switch-on the machine in advance, when the probability of part arrival rises. Also, the switch-on command might be by the actual part arrival, whenever it happens before τ_{on}.

We do not include the busy state into the energy model because the energy consumed by the process in the busy state is not affected by the control. We assume that machine in state s requires a constant and deterministic amount of power w_s such that $w_2 > w_3 > w_1 \geq 0$. This assumption is realistically representing manufacturing equipment that commonly requires high power while executing the startup.

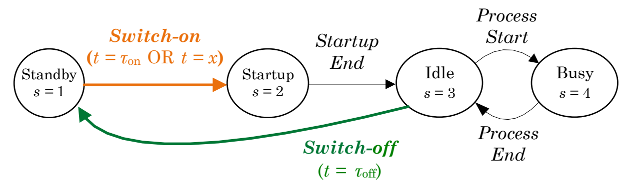


Fig. 1: State model of a machine controlled with the SP policy

According to the arrival occurrence x and control parameters $\boldsymbol{\tau}$, four events A_k may happen; hence, machine energy demand e_k and part holding time h_k are conditioned to the occurrence of the event A_k. A description follows:

- Parts arrive before the switch-off: $A_1 = \{0 \leq x \leq \tau_{\text{off}}\}$ and machine consumes $e_1 = w_3 \cdot x$. The process starts immediately ($h_1 = 0$).
- Parts trigger the switch-on: $A_2 = \{\tau_{\text{off}} < x \leq \tau_{\text{on}}\}$ and machine consumes $e_2 = w_3 \cdot \tau_{\text{off}} + w_2 \cdot t_{\text{su}} + w_1(x - \tau_{\text{off}})$. The part waits for the whole startup and the part is held for a time $h_2 = t_{\text{su}}$.
- Parts arrive while the machine is executing the startup: $A_3 = \{\tau_{\text{on}} < x \leq \tau_{\text{on}} + t_{\text{su}}\}$ and machine consumes $e_3 = w_3 \cdot \tau_{\text{off}} + w_1(\tau_{\text{on}} - \tau_{\text{off}}) + w_2 \cdot t_{\text{su}}$. Part holding time is $h_3 = \tau_{\text{on}} + t_{\text{su}} - x$.
- Parts arrive after the machine has completed the startup: $A_4 = \{x > \tau_{\text{on}} + t_{\text{su}}\}$. The energy consumed is $e_4 = w_3 \cdot \tau_{\text{off}} + w_1(\tau_{\text{on}} - \tau_{\text{off}}) + w_2 \cdot t_{\text{su}} + w_3(x - \tau_{\text{on}} - t_{\text{su}})$ and the process starts immediately ($h_4 = 0$).

Events $A_k | k = 1, 2, 3, 4$ are mutually exclusive and collective exhaustive with probability of occurrence $P(A_k)$. According to the vector of control parameters $\boldsymbol{\tau} = \{\tau_{\text{off}}, \tau_{\text{on}}\}$, the expected NPE consumed in a cycle $\Phi(\boldsymbol{\tau})$ and the expected holding time $H(\boldsymbol{\tau})$ are respectively:

$$\Phi(\boldsymbol{\tau}) = \sum_{k=1}^{4} \mathbb{E}[e_k | A_k] \cdot P(A_k) \tag{1}$$

$$H(\boldsymbol{\tau}) = \sum_{k=1}^{4} \mathbb{E}[h_k | A_k] \cdot P(A_k). \tag{2}$$

$\Phi(\boldsymbol{\tau})$ and $H(\boldsymbol{\tau})$ are function of machine parameters (i.e., power and startup duration), of the control parameters applied and of the pdf $f(x)$.

C. Optimization Problem

The following control problem needs to be solved to search for proper values of control parameters:

$$\min_{\boldsymbol{\tau}} \quad \varphi(\boldsymbol{\tau}) = \Phi(\boldsymbol{\tau}) + w_q \cdot H(\boldsymbol{\tau}) \tag{3}$$

$$\text{Subject to:} \quad \tau_{\text{on}} > \tau_{\text{off}} \tag{4}$$

$$\tau_{\text{off}}, \tau_{\text{on}} \in \mathbb{R}_0^+ \tag{5}$$

Constraint (4) represents the control feasibility between control parameters (i.e., the switching on must happen after the switching off). Constraints (5) define the domain of decision variables. Problem objective in equation (3) considers both the NPE $\Phi(\boldsymbol{\tau})$ and an holding penalty $w_q \cdot H(\boldsymbol{\tau})$ consumed per part produced. The weight w_q should be properly tuned in numerical experiments to represent a production rate target such that the controlled system assures a certain service level. Hence, as w_q increases, the optimal control policy tends to the AO policy because the holding time becomes more important.

III. On-Line Time-Based Control Policy for Energy Efficiency

The off-line control problem formalized in section II is extended by keeping the same assumptions except the knowledge of the arrival distribution. Herewith, we assume that machine starvation times X are randomly distributed according to an unknown pdf $f(x)$. In off-line policies, the pdf $f(x)$ is assumed to be known.

This section describes an innovative algorithm for solving the on-line EEC problem. The algorithm performs iteratively two phases: a learning phase where the distribution is estimated (section III-A) and an optimization phase that searches for the optimal control parameters of the control policy (section III-B).

A. Parameter Estimation for Machine Idle Times

Complete reviews on the estimation of random variable distributions can be found in the literature (e.g., [20],[21]) where several methods are discussed and classified, e.g. maximum likelihood estimation, method of moments, polynomial estimation, Kernel estimation method.

According to the amount of available information, the proposed algorithm uses two different methods selected from the literature. The Maximum Likelihood Estimation (MLE, [22]) is used to estimate the parameters of a known distribution type. When the distribution type is assumed to be unknown, the Kernel Density Estimation (KDE, [23][24]) with a Gaussian Kernel function is used. Both methods are based on a frequentist inference, which means the estimation is drawn only by the sample data. Alternative methods could be used without making large extensions of the developed analysis.

1) Estimation with a Known Distribution Type: In order to use the MLE method, the pdf $f(x;\boldsymbol{\theta})$ is assumed to be known except for a vector of k parameters $\boldsymbol{\theta} = \{\theta_j | j = 1, \ldots, k\}$, which are unknown. Therefore, the MLE method finds the estimated parameters $\hat{\boldsymbol{\theta}}_n$ given a set of n observed arrivals $\mathbf{x} = \{x_i | i = 1, \ldots, n\}$. Define the Likelihood function as:

$$\mathcal{L}(\boldsymbol{\theta};\mathbf{x}) = \prod_{i=1}^{n} f(x_i;\boldsymbol{\theta}). \tag{6}$$

Function $\mathcal{L}(\boldsymbol{\theta};\mathbf{x})$ represents the pdf of $\boldsymbol{\theta}$ given observations $\mathbf{x}$. The MLE method solves the following optimization problem:

$$\hat{\boldsymbol{\theta}}_n = \arg\max_{\boldsymbol{\theta}\in\mathbb{R}^k} \mathcal{L}(\boldsymbol{\theta};\mathbf{x}) \tag{7}$$

where k is the problem dimension. The estimated distribution of arrivals is $\hat{f}(x|\mathbf{x}) = f(x;\hat{\boldsymbol{\theta}}_n)$ and the estimates are asymptotically unbiased as n increases.

2) Estimation with an Unknown Distribution Type: Given a set of n observed arrivals $\mathbf{x}$, the kernel density estimator is:

$$g(x;h,\mathbf{x}) = \frac{1}{nh}\sum_{i=1}^{n} \mathcal{K}\left(\frac{x - x_i}{h}\right) \tag{8}$$

where $\mathcal{K}(\cdot)$ is the kernel function and $h > 0$ is a smoothing parameter, i.e., the bandwidth. We consider a Gaussian Kernel function $\mathcal{K}(\cdot)$. The estimates defined by equation (8) are asymptotically unbiased as n increases ([24]), then:

$$\lim_{n\to\infty} g(x;h,\mathbf{x}) = f(x). \tag{9}$$

Further, since the random variable X is defined onto positive domain, i.e., $\mathbb{R}_0^+$, the estimator is normalized:

$$g^{\text{norm}}(x;h,\mathbf{x}) = \frac{g(x;h,\mathbf{x})}{\int_0^\infty g(x;h,\mathbf{x})}. \tag{10}$$

The KDE method finds the optimal bandwidth h_n^* and provides the estimator $\hat{f}(x;\mathbf{x}) = g^{\text{norm}}(x;h_n^*,\mathbf{x})$. In more details, the bandwidth parameter value h_n^* is found with the leave-one-out cross validation method ([25]). Therefore, optimal bandwidth h_n^* makes most likely the n data observed by solving the following optimization problem:

$$h_n^* = \arg\max_{h\in\mathbb{R}^+} \{\mathcal{L}(h;\mathbf{x})\} \tag{11}$$

where function $\mathcal{L}(h;\boldsymbol{x})$ is the Likelihood function as:

$$\mathcal{L}(h;\mathbf{x}) = \prod_{i=1}^{n} \frac{1}{h\cdot(n-1)} \left[\sum_{\substack{j=1\\ j\neq i}}^{n} \mathcal{K}\left(\frac{x_i - x_j}{h}\right)\right]. \tag{12}$$

B. Optimization

Similarly for the SP policy described in section II, time-based thresholds are used to control the machine. Given n observed arrivals $\mathbf{x}$, the vector of control parameters $\boldsymbol{\tau}_n = \{\tau_{\text{off},n};\tau_{\text{on},n}\}$ includes two thresholds, i.e., $\tau_{\text{off},n}$ and $\tau_{\text{on},n}$, used to control machine state after the departure of part n. Denote respectively $\Phi_n(\boldsymbol{\tau};\mathbf{x})$ and $H_n(\boldsymbol{\tau};\mathbf{x})$ the expected energy consumed per cycle and the expected holding time obtained with the estimated pdf. The objective function $\varphi(\boldsymbol{\tau};\mathbf{x})$ becomes:

$$\min_{\boldsymbol{\tau}} \ \varphi(\boldsymbol{\tau};\mathbf{x}) = \Phi_n(\boldsymbol{\tau};\mathbf{x}) + w_q \cdot H_n(\boldsymbol{\tau};\mathbf{x}). \tag{13}$$

The control problem in equation (13) is solved subject to constraints (4) and (5) at each algorithm iteration. As n increases, the control problem approaches that in equations (3)-(5) as the estimates $\hat{f}(x;\mathbf{x})$ are asymptotically unbiased. This phase is independent from the estimation method used in the learning phase.

IV. The Benefit of an On-Line Approach

A Numerical analysis is provided to investigate the benefit of the proposed approach. The On-line approach is compared to the Off-line approach in terms of expected saving and expected throughput of the controlled machine. Off-line and on-line approaches are applied and common random numbers ([26]) are used for the comparison among the two approaches to decrease the noise.

TABLE I: Machine parameters

Machine	w_1 [kW]	w_2[kW]	w_3[kW]	t_{su} [s]	t_p [s]
M1	0.52	6.08	5.35	24	168
M2	3.12	12.5	11	24	168

A. *Off-line vs On-line*

We consider a machining center (M1) for powertrain applications characterized as in Table I, with processing time $t_p = 168$ s, and holding penalty $w_q = 12$ kW. Also, we assume an Erlang-3 distribution defined by rate parameter $\lambda_{\text{real}} = 0.037$ and scale parameter $k = 3$ such that the starvation mean time is $t_a = 81$ s. The AO policy, i.e., $\boldsymbol{\tau} = (\infty, \infty)$, obtains an objective value of $\varphi(\infty, \infty) = w_3 \cdot t_a = 433$ kJ/part and an expected production rate (PR) of $PR(\infty, \infty) = 14.46$ part/h (Table II). The AO policy is used as reference policy. The SP policy achieves 8.78% of saving on the objective function: the optimal control is $\boldsymbol{\tau}^* = (0, 37.2)$ s, the expected energy $\varphi(0, 37.2) = 395$ kJ/part and the expected production rate $PR(0, 37.2) = 14.08$ part/h (Table II). The SP policy assumes the parameter λ_{real} is known.

The off-line approach requires a learning phase where a certain number of observations N is collected and the estimate of λ_{real} is provided afterwards. The number N of collected observations is usually high and varies from case to case. In this case, $N = 500$ observations are acquired (i.e., around 35 hours of acquisition) and, then, the optimal control parameters of the off-line approach are computed (Table II). The MLE method is applied to provide estimate $\hat{\lambda}_N$ given $N = 500$ observations. It can be noticed that the average control parameter (i.e., $\tau_{\text{on},N} = 36.46$ s) is slightly smaller than the off-line theoretical solution (i.e., $\tau^*_{\text{on}} = 37.2$ s) resulting in a higher expected energy and in a lower production rate. This is related with the limited number of observations N and the error in the estimated $\hat{\lambda}_N$.

The on-line approach is applied while observations are collected such that the advantage compared to an off-line approach can be identified in the transient period, i.e., for $n < N$ when the off-line approach is accumulating observations and not controlling the system yet. Results obtained by increasing n are in Table III. In particular, the on-line algorithm estimates $\hat{\lambda}_n$. The optimal control parameters are computed at each algorithm iteration (i.e., each n = 10 new observations) meaning that an on-line policy immediately starts controlling the machine. Figure 2a shows the range of change for the control parameter $\tau_{\text{on},n}$ over 10 replications and Figure 2b provides the details of one particular replication. Although the estimated $\hat{\lambda}_n$ varies significantly from replication to replication and it might be far from λ_{real}, the objective value of the control problem is close to the optimum. In the learning period, while off-line policies cannot be applied, on-line policies can achieve good advantages in terms of expected energy saving. The control parameters change at each iteration although they tend to stabilize as n increases (Figure 2). Since $\tau^*_{\text{off}} = 0$ in this case, the on-line algorithm chooses values very close to *zero* which are not reported herewith.

Further, the actual performance based on the stream of observations is analyzed for each replication. Figure 3 shows the saving obtained for each replication as n increases when the on-line approach is used. Actual savings are sample-dependent and include the sequence of controls applied over time. Among the 10 replications analyzed, some replications show an increase of energy consumption with respect to AO (negative savings in Figure 3). Since the estimate $\hat{\lambda}_n$ obtained might be very biased in presence of few observations available, the control parameters obtained with the on-line approach might cause an unexpected increase of energy consumption.

B. *The Value of Information: MLE vs KDE Methods*

The assumption to apply the MLE method is that distribution $f(x)$ is known except for its parameters. Further relaxing this assumption might be necessary in the practice, where the shape of the distribution is often unknown a priori. We use KDE method and we compare the results with MLE.

The scenario described in Section IV-A is solved with KDE method and the same observations are used to obtain results in Figure 4. Results are similar, despite more variability appears and more replications result in unexpected high energy consumption.

As second example, we consider M1 as in Table I and $w_q = 1$ kW. Also, we assume a Weibull distribution with $k_{\text{real}} = 0.45$ and $\lambda_{\text{real}} = 15.73$ such that the starvation mean time is $t_a = 39$ s. As reference, the AO policy obtains an objective value of $\varphi(\infty, \infty) = 208.65$ kJ/part and the expected production rate is $PR(\infty, \infty) = 17.39$ part/h. The SP policy, assuming that the parameters λ_{real} and k_{real} are known, achieves around 40% of saving on the objective function: the optimal control is $\boldsymbol{\tau}^* = (15, \infty)$ s and $\varphi(15, \infty) = 125$ kJ/part. With SP policy, the machine is switched on upon arrival and the expected production rate is $PR(15, \infty) = 13.68$ part/h. Results obtained with MLE and KDE methods are reported in Figure 5. The two methods perform similarly on the average, but the KDE method is more variable.

C. *A Comparison with a Simple Policy*

A comparison with a simple policy is also provided. We consider an on-line approach that only estimates the starvation mean time $\hat{t}_a$ from observations.

If $\hat{t}_a < t_{\text{su}}$, the control is not advantageous and the machine is kept AO. Otherwise, this control policy switches off the machine with $\tau_{\text{off}} = 0$ and switches it on with $\tau_{\text{on}} = \hat{t}_a - t_{\text{su}}$ when the following condition holds:

$$\hat{t}_a \cdot w_3 > (\hat{t}_a - t_{\text{su}}) \cdot w_1 + t_{\text{su}} \cdot w_2. \tag{14}$$

The left-hand side of equation (14) is the estimated energy for the AO policy. The right-hand side represents that obtained with control parameters $\boldsymbol{\tau} = \{0, \hat{t}_a - t_{\text{su}}\}$.

Results are in Figure 6. For the Weibull distribution, this simple policy actually consumes more than the AO policy (+40%). This happens because 78% of arrivals occur before the estimated mean and pay for the holding penalty. The controls are triggered too early (i.e., $\tau_{\text{off}} = 0$ and $\tau_{\text{on}} \approx 14$ s) when the probability of arrival is high. For the Erlang, the simple policy switches off the machine at $\tau_{\text{off}} = 0$ s, as for the

TABLE II: Policy comparison among Always On and SP which assume λ_{real} is known, and Off-line approach which estimates $\hat{\lambda}_N$ after $N = 500$ observations (95%IC, 10 replications, M1, MLE method).

Policy	λ_{real}	τ^*_{off} [s]	τ^*_{on} [s]	$\varphi(\boldsymbol{\tau}^*)$ [kJ/part]	$PR(\boldsymbol{\tau}^*)$ [part/h]
Always On	0.037	∞	∞	433	14.46
SP	0.037	0	37.2	395	14.08
Policy	$\hat{\lambda}_N$	$\tau_{\text{off},N}$ [s]	$\tau_{\text{on},N}$ [s]	$\varphi(\boldsymbol{\tau}_N)$ [kJ/part]	$PR(\boldsymbol{\tau}_N)$ [part/h]
Off-line SP	0.0373 $\pm$ 0.0006	0	36.46 $\pm$ 1.72	389.05 $\pm$ 0.56	13.602 $\pm$ 0.021

TABLE III: Performance of the on-line algorithm (95%CI, 10 replications, M1, Erlang distribution, MLE method).

n	50	100	250	500
$\hat{\lambda}_n$	0.0372 $\pm$ 0.0028	0.0373 $\pm$ 0.0022	0.0370 $\pm$ 0.0009	0.0373 $\pm$ 0.0006
$\tau_{\text{off},n}$ [s]	0	0	0	0
$\tau_{\text{on},n}$ [s]	38.47 $\pm$ 8.60	37.49 $\pm$ 6.85	37.34 $\pm$ 2.54	36.46 $\pm$ 1.72
$\varphi(\boldsymbol{\tau}_n)$ [kJ/part]	398.35 $\pm$ 3.31	397.27 $\pm$ 2.17	395.65 $\pm$ 0.31	395.50 $\pm$ 0.21
$PR(\boldsymbol{\tau}_n)$ [part/h]	13.621 $\pm$ 0.088	13.611 $\pm$ 0.075	13.613 $\pm$ 0.031	13.602 $\pm$ 0.021

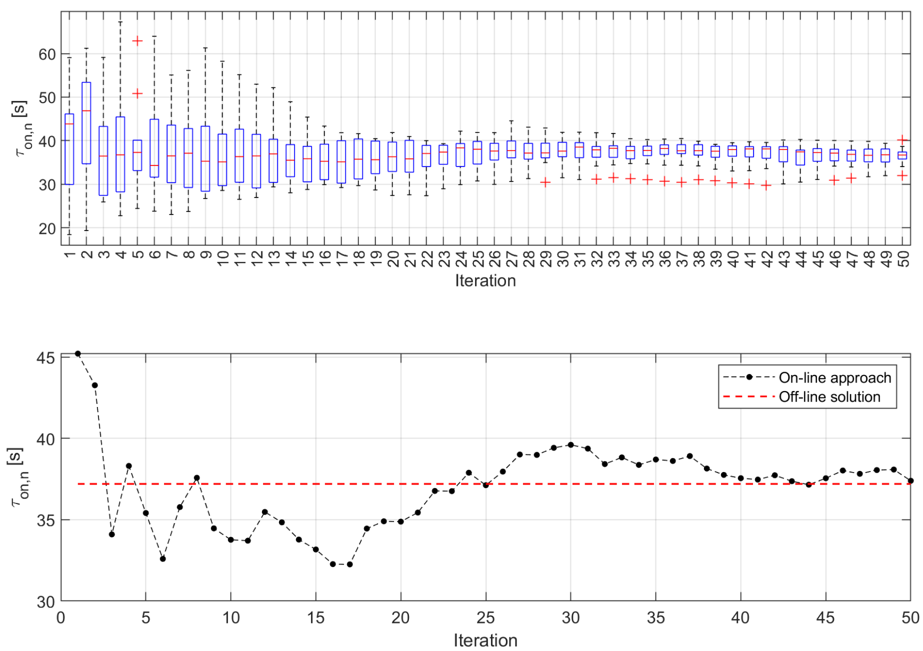


Fig. 2: Control parameter $\tau_{\text{on},n}$ according to the on-line approach (M1, Erlang distribution, MLE method): a) boxplot of 10 replications, b) single-replication example. The algorithm iterates every n = 10 observations.

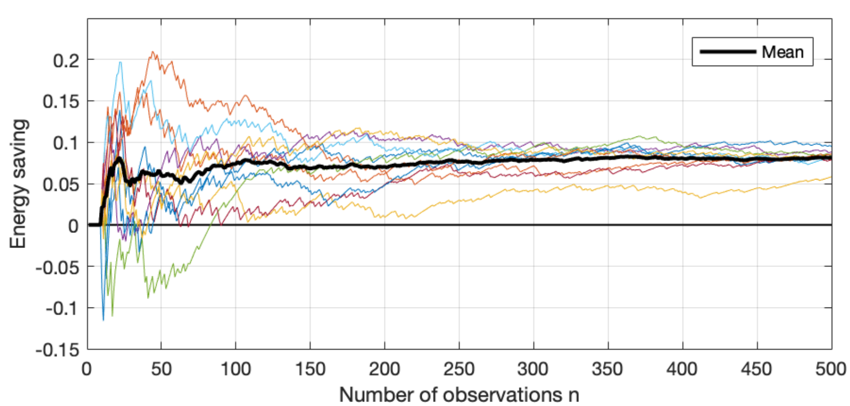


Fig. 3: Sample-based saving with on-line approach compared to AO policy (10 replications, M1, Erlang distribution, MLE method).

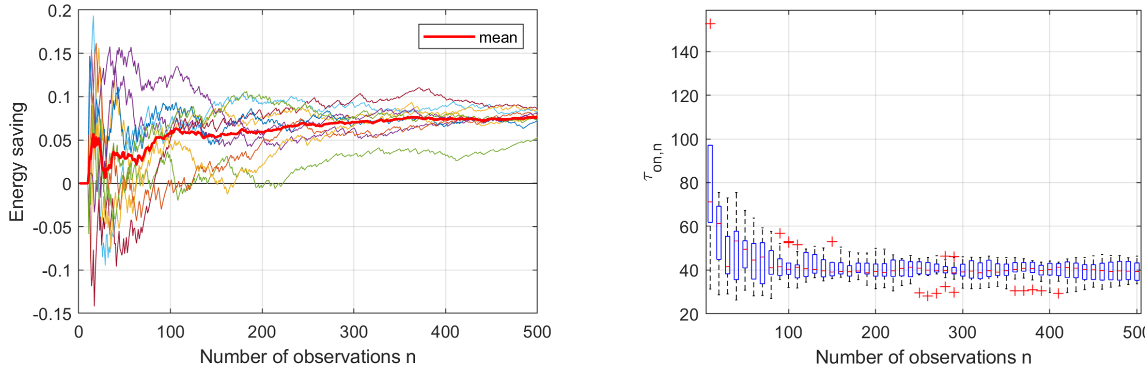


Fig. 4: Results obtained with KDE method (10 replications, M1, Erlang distribution): left panel sample-based saving with on-line approach compared to AO policy, right panel boxplot of control parameter $\tau_{\text{on},n}$.

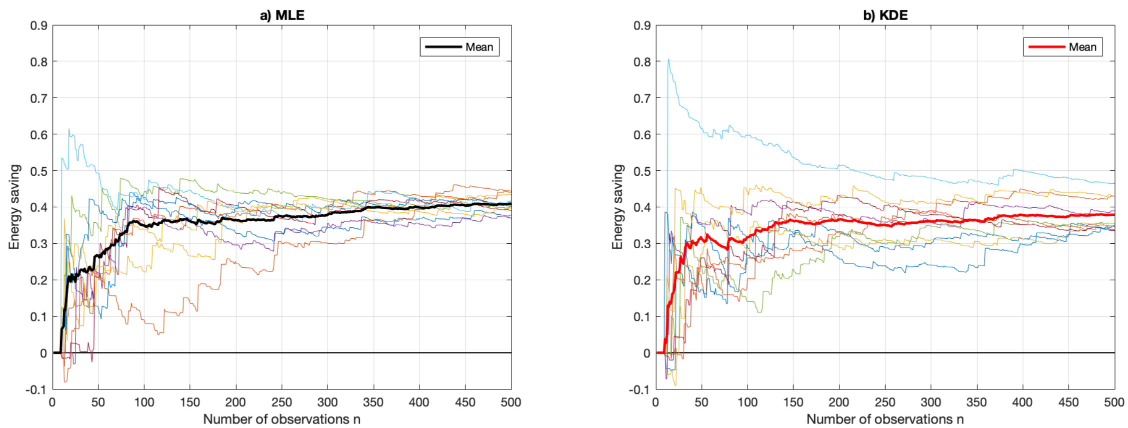


Fig. 5: Sample-based saving with on-line approach compared to AO policy (10 replications, M1, Weibull distribution).

off-line theoretical solution, and switches on the machine at $\tau_{\text{on}} \approx 56$ s. Despite the control parameter is far from the off-line theoretical solution (37.2 s), it has good results (saving of around 6%) compared to that of the on-line policy (8%).

D. Remarks

We achieved the following insights:

- An *Off-line* approach requires a long learning phase during which the machine is not controlled and stays idle while waiting for parts. If the machine operates in frequently changing production environments, the learning phase must be frequently repeated and the policy management effort increases significantly;
- An *On-line* approach starts machine control while learning from data and the performance are promising;
- Control parameters change frequently over time in the on-line approach; this could be an issue because the variability in the downstream production process might increase.

Since the estimates obtained during the learning phase of the algorithm might be very biased in presence of few observations available, the control parameters obtained in the optimization phase might cause an unexpected increase of energy consumption. Also, the obtained control varies significantly among replications. This situation appears more frequently with KDE method. The risk of incurring in unexpected high energy consumption should be reduced. Furthermore, at each algorithm iteration, control parameters might change incurring in an implementation cost. Therefore, the new control parameters should not be adopted if the implementation cost is higher than the expected advantage the change causes.

V. Implementation Policy

In addition to the two phases described in Section III, the resolution of the on-line problem includes a third phase as in Figure 7: the learning phase (section III-A), the optimization phase (Section III-B), and an implementation phase (section V) of the control where a new *On-line implementation policy* (IP) is proposed. The idea is to compare two controls in terms of objective function and to select the best.

Assume that control parameters $\tilde{\boldsymbol{\tau}}$ are currently implemented and control $\boldsymbol{\tau}_n$ is obtained from equation (13). The expected difference δ among the implemented control and the last computed one is:

$$\delta = \varphi(\tilde{\boldsymbol{\tau}}) - \varphi(\boldsymbol{\tau}_n) \tag{15}$$

such that the new computation is improving the solution when $\delta > 0$. Further, an implementation cost c_n of changing control

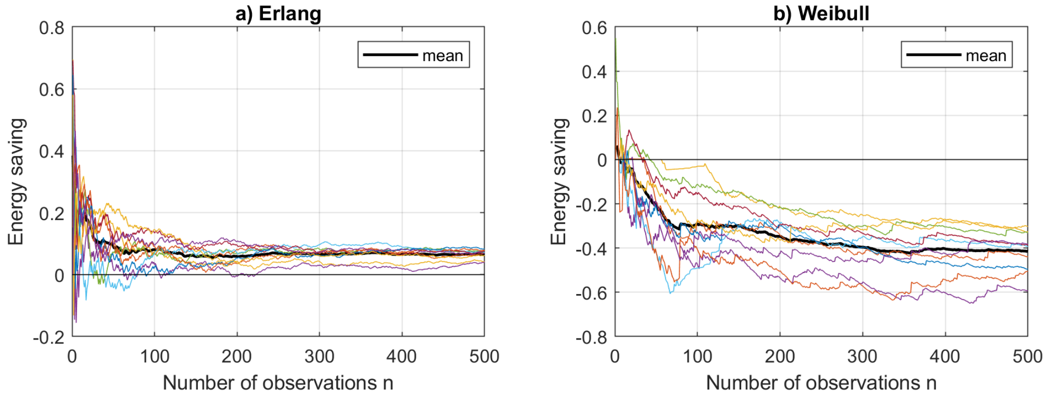


Fig. 6: Sample-based saving with the simple on-line policy based on the computation of the starvation mean time (10 replications) - Erlang (a) and Weibull (b).

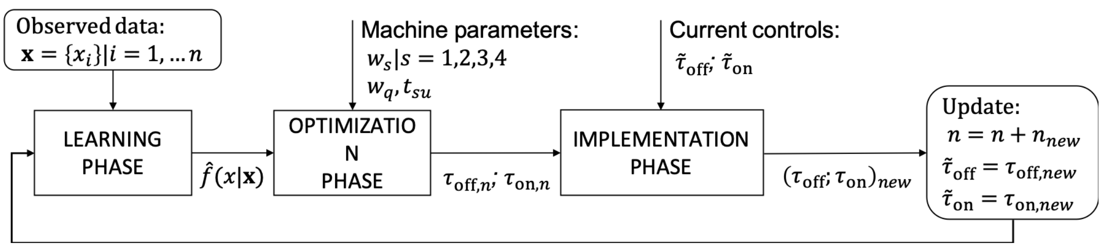


Fig. 7: Algorithm main phases. The observed data are machine starvation times and the control parameters are selected and updated at each algorithm iteration.

parameters after n observations are collected is included such that the number of changes in the applied control is reduced. The idea is to implement controls $\boldsymbol{\tau}_n$ when the improvement with respect to the current objective function is higher than the implementation costs, i.e., $\delta > c_n$. Parameter c_n is defined as:

$$c_n = c_0 \cdot (1 - \gamma)^n \tag{16}$$

where $\gamma \in [0, 1]$ is a discount factor and c_0 represents the implementation cost. The discount factor is used to consider the history such that as the number of observations increases, the cost of implementation c_0 has less impact. When $\gamma = 0$, the implementation cost is not discounted and keeps constant over time; when $\gamma = 1$, the implementation cost is not considered. Other discount functions can be chosen without a large extension to the developed analysis.

With the on-line approach, the theoretical objective function $\varphi(\cdot)$ is not known and the uncertainty embedded in the observations should be considered. For the MLE method, we can derive analytically δ in expression (15) because the distribution of $\boldsymbol{\theta}$ is provided in equation (6). Therefore, the expectation δ is a random variable (denoted with $D_{\boldsymbol{\theta}}$) being a function of $\boldsymbol{\theta}$:

$$D_{\boldsymbol{\theta}} = \varphi(\tilde{\boldsymbol{\tau}}; \boldsymbol{\theta}) - \varphi(\boldsymbol{\tau}_n; \boldsymbol{\theta}). \tag{17}$$

Therefore, given a certain control $\tilde{\boldsymbol{\tau}}$, a new control $\boldsymbol{\tau}_n$ is implemented only if the following condition holds:

$$P(D_{\boldsymbol{\theta}} \leq c_n) \quad \leq \quad \alpha \tag{18}$$

where $\alpha \in [0, 1]$ and probability $P(D_{\boldsymbol{\theta}} \leq c_n)$ is calculated on the probability space of random variable $\boldsymbol{\theta}$. Condition (18) limits the probability that the advantage of implementing new control parameters $\boldsymbol{\tau}_n$ is smaller than the implementation cost c_n. Algorithm parameters (i.e., α, c_0 and γ) are case dependent and can be related, for instance, to risk-adversity of the user. Further, the policy becomes more conservative as threshold α and discount γ decrease.

A similar argument applies when KDE method is used. In this case, the observed data ($\mathbf{x}$) are used to estimate δ. Define the following random variable Z:

$$Z = Z_1 - Z_2 - c_n \tag{19}$$

where Z_1 (Z_2) is the non processing energy consumed by one part plus the holding penalty with control $\tilde{\boldsymbol{\tau}}$ ($\boldsymbol{\tau}_n$). A paired t-test is used to evaluate if the new control parameters $\boldsymbol{\tau}_n$ enable an energy consumption statistically lower than that guaranteed with $\tilde{\boldsymbol{\tau}}$. We assume that the difference $Z(\cdot)$ is iid normal with mean μ_Z, and μ_Z is estimated using the sample mean obtained from observations $\mathbf{x}$. The following hypothesis are used in the test: $H_0 : \mu_Z > 0$ and $H_1 : \mu_Z \leq 0$. If the null hypothesis is not rejected (confidence $1-\alpha$), control parameters $\boldsymbol{\tau}_n$ provide a significant improvement compared to $\tilde{\boldsymbol{\tau}}$. Therefore, a new control $\boldsymbol{\tau}_n$ is implemented. Similarly to the MLE method, the policy becomes more conservative as confidence $1 - \alpha$ used for the paired-t test increases and discount γ decreases.

VI. Numerical Results

The effect of the proposed policy is analyzed in this section. The AO policy is the reference policy and common random numbers are used in the comparison with other policies. Further, a sensitivity analysis on algorithm parameters is provided. Numerical results have been obtained in Matlab environment.

A. Results of the Implementation Policy

Instances described in section IV are solved including the implementation phase with the following setting: $\alpha = 0.05$, $\gamma = 0$ (i.e., the implementation cost $c_n = c_0$) and $c_0 = 3$ kJ. Results in terms of energy savings are in Figure 8 compared to Figure 3 and Figure 4. Also, results are collected in Table IV. The effect of the implementation phase is double. The *implementation policy* (IP) helps to prevent negative savings (no advantages with respect to AO) in the most unfavorable cases and it also reduces the negative peak, i.e., maximum increase of energy. As side-effect, the average saving obtained with the implementation condition can be lower since the new algorithm is more conservative.

Results show that the algorithm prevents an early implementation of control parameters. Without the implementation phase, the on-line approach starts implementing the control at the first iteration. Also, control parameters change at each new iteration such that the total implementation cost is high. One replication is reported in Figure 9 and it can be noticed that the on-line approach (black-circles) starts implementing the control at $n_{\text{initial}} = 10$ and that control parameter $\tau_{\text{on},n}$ changes at each iteration varying between 32 s and 44 s. When a cost c_n is included (blue-triangles), the algorithm delays the implementation of the control which is applied after $n_{\text{initial}} = 70$ observations.

Moreover, the control parameters implemented are more stable. Therefore, it might happen that the control parameters do not tend toward the off-line theoretical solution because the expected improvement is lower than the implementation cost. However, it results in a low cost of implementation which compensates the aforementioned saving gap. When the implementation cost increases, the policy becomes more conservative: the control is applied later although the implementation costs is further reduced (i.e., less changes in control parameters). This is further discussed in section VI-B.

B. Sensitivity Analysis

Algorithm parameters are related to the implementation phase and allow the user to tune the control policy. Experiments are designed by varying factors as in Table V. We consider a machining center (M2) for powertrain applications characterized as in Table I, processing time of $t_p = 168$ s and $w_q = 0$ kW. Also, we assume a Weibull distribution (Weibull 2) defined by rate parameter $\lambda_{\text{real}} = 21$ and scale parameter $k_{\text{real}} = 0.45$ such that the starvation mean time is $t_a = 52$ s.

The AO policy, i.e., $\boldsymbol{\tau} = (\infty, \infty)$, obtains an objective value of $\varphi(\infty, \infty) = 572$ kJ/part and an expected production rate of $PR(\infty, \infty) = 16.2$ part/h. The AO policy is used as reference. The SP policy assuming that the parameters λ_{real} and k_{real} are known is considered as benchmark achieving 38% of saving on the objective function: the optimal control is $\boldsymbol{\tau}^* = (15.13, \infty)$ s, the expected energy $\varphi(15.13, \infty) = 356$ kJ/part and the expected production rate $PR(15.13, \infty) = 15.48$ part/h.

The results of some experiments are compared in Figure 10. Left panel represents four settings where no discount is applied to the experimental cost; right panel represents three cases with different discount γ. A situation with high c_0 and/or low α yields to a longer delay in implementing the control and represents a more conservative algorithm setting. As a consequence, it results in lower savings. Algorithm parameters affect the number of changes in the control parameters. Indeed, as in Table VI, the implementation of control parameters appears later in time when c_0 increases, α decreases, and γ decreases. Similarly for the occurrence of changes in the control.

C. A Note on the Production Rate

The problem addressed in this work is focused onto energy consumption reduction. In equation (3), a penalty term $H_n(\cdot)$ is included such that the algorithm can be adapted to a situation where the machine throughput must satisfy a certain target. Whenever the machine is switched off, its production rate is reduced because of the startup time. Therefore, a trade-off exists among energy efficiency and production criteria. Increasing the value of penalty w_q, the holding time becomes more important and the algorithm tends to keep the machine Always On.

Machine utilization for the analyzed cases are reported in Table III and Table IV. As the implementation cost increases, machine mean throughput in the observation period increases because the control is applied after more accumulated observations. Indeed, the machine is kept idle (AO policy) while observations are acquired with no delays in processing parts.

D. A Note on Implementation and Computational Effort

Results have been obtained with Matlab2018b on an Intel Core i7-6500U with 2.50GHz and 16GB of RAM. The estimation problem in equation (7) has been solved analytically or numerically (i.e., function *fzero*) for the MLE method, whilst function *fminsearch* has been used to find h_n^* in equation (11) (i.e., KDE method). The optimization problem in equations (8) is solved with function *fmincon*.

The computational time required by the algorithm depends on the amount of observed data and on the method used for the learning phase. The algorithm requires 13 minutes per iteration on the average. Each replication is composed by 50 iterations of the algorithm which iterates every 10 new observations (i.e., last iteration uses 500 observations). It is noteworthy that for few observations the iteration is performed in negligible time.

For practical applications, the algorithm must complete one iteration in a very short time being able to control the machine on-line. For instance, it might be required to complete the operation before the process completion of part n. If not possible, the machine will keep the current control until the iteration is completed and the number of observations collected meanwhile are used in the future iteration.

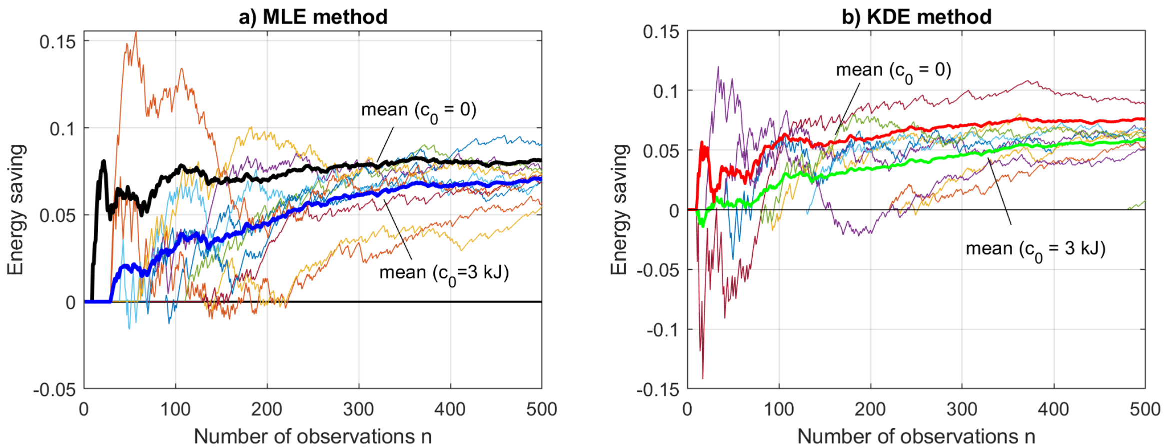


Fig. 8: Sample-based saving with on-line IP policy compared to AO policy (10 replications, M1, Erlang distribution). Solved instances are the same as in Figure 3 and Figure 4: black and red lines are obtained without the implementation phase ($c_0 = 0$).

TABLE IV: Sample-based comparison of the on-line algorithm with/without implementation phase (95% CI, 10 replications, M1, Erlang distribution). Energy is expressed in [kJ/part] and throughput PR in [part/h].

Method	Policy	Performance	$n = 50$	$n = 100$	$n = 250$	$n = 500$
None	AO	Energy	435.72 ± 33.33	432.92 ± 26.53	434.11 ± 10.60	430.66 ± 7.21
		PR	14.45 ± 0.36	14.47 ± 0.28	14.45 ± 0.12	14.49 ± 0.08
MLE	On-line ($c_0 = 0$)	Energy	407.18 ± 13.24	399.71 ± 12.16	400.78 ± 5.58	395.51 ± 5.30
		PR	14.10 ± 0.37	14.11 ± 0.31	14.08 ± 0.15	14.11 ± 0.10
	On-line ($c_0 = 3$ kJ)	Energy	426.19 ± 21.67	418.00 ± 13.58	409.66 ± 6.30	400.08 ± 5.36
		PR	14.40 ± 0.40	14.32 ± 0.38	14.15 ± 0.19	14.15 ± 0.12
KDE	On-line ($c_0 = 0$)	Energy	420.58 ± 33.33	406.88 ± 26.53	404.55 ± 10.60	397.75 ± 7.21
		PR	13.96 ± 0.38	14.00 ± 0.31	14.00 ± 0.14	14.06 ± 0.11
	On-line ($c_0 = 3$ kJ)	Energy	433.48 ± 25.52	423.17 ± 21.81	416.48 ± 12.39	405.18 ± 6.50
		PR	14.35 ± 0.36	14.31 ± 0.39	14.19 ± 0.23	14.16 ± 0.17

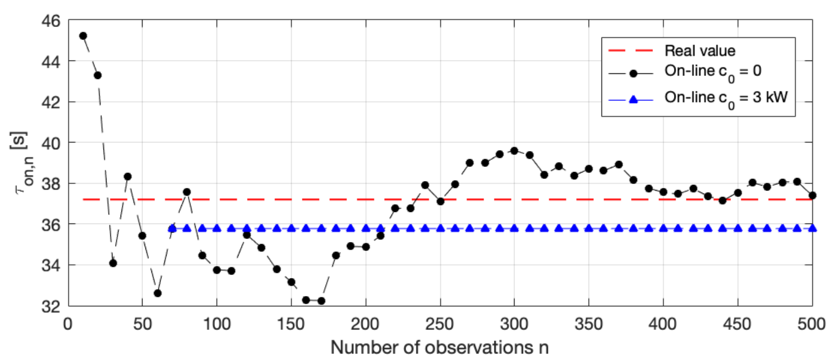


Fig. 9: Implemented control parameter $\tau_{\text{on},n}$ with the on-line policy (10 replications, M1, Erlang distribution, MLE method). Once the control is applied, $\tau_{\text{off},n} = 0$.

TABLE V: Setting of the algorithm parameters.

Factor	Level 1	Level 2	Level 3
α	0.5%	5%	35%
c_0	17 kJ	73 kJ	
γ	0.02	0	

VII. An Application at System Level

We consider an assembly line with single-machine stations as in the literature [27]. The proposed algorithm (with KDE method) is applied to all machines and the control at machine m minimizes the expected energy consumption of the machine. The effect at machine and line levels is investigated.

TABLE VI: Effects of factors γ, α and c_0 over the mean number of changes after 500 observations, and mean number of observations before implementing the control n_{initial} (10 replications, M2, Weibull distribution 2, KDE method).

γ	α	c_0 [kJ]	N. of changes	n_{initial}
0	0.05	17	1.7	36
0	0.05	73	1.0	47
0	0.35	17	1.8	36
0	0.35	73	1.0	53
0.02	0.005	17	1.3	76
0.02	0.005	73	1.0	203
0.02	0.35	17	2.3	36
0.02	0.35	73	1.3	50

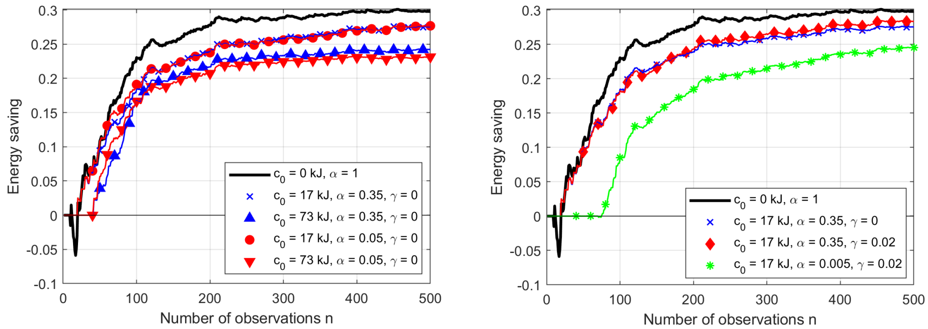


Fig. 10: Sample-based saving with on-line IP policy compared to AO policy with different algorithm setting (10 replications, M2, Weibull distribution 2, KDE method).

The line has 8 stations, splits in two parallel branches ($m = 3, 4$ and $m = 5, 6$) and merges before $m = 7$. Machine processing time is 60 seconds except for $m = 3, 4, 5, 6$ where the time is 120 seconds. We apply the control to a scenario where $m = 5, 6$ are not working: the line becomes highly unbalanced and EEC might save energy because of low utilization of not-bottleneck machines.

Machine power requests and startup times are in Table VII, and buffer capacities are $B_2 = 60, B_3 = 80, B_3 = 25, B_4 = 35$ and, $B_6 = 45$. When a machine is blocked, its power is null. Operational dependent failures are considered (failures and repairs are exponentially distributed with MTBF and MTTR as in Table VII).

A discrete event model of the system is created in Simulink (Matlab environment) and the on-line policy is applied at each machine. After the first 500 observations, the algorithm operates with a moving window of 500 observations, keeping the most recent ones. The control is applied with $\gamma = 0$, $\alpha = 0.05$, and C_0 as in Table VII. The simulation starts with an empty system and all machines in idle state. We perform 8 simulation runs of 5600 parts.

The line, under the AO policy, consumes on average 1438±97 kJ/part with a throughput of 25.98±0.13 part/h. The bottleneck is machine $m = 3$ with a utilization of 95%. The average NPE consumed by each machine is reported in Table VIII. When the AO policy is applied, machines $m = 3$ and $m = 4$ consume less NPE energy, because of the low probability of being idle.

With the on-line policy, the machines are controlled and the line consumes on average 991±40 kJ/part with a saving of 31% with respect to the AO case. The production rate is reduced of 1.3%. It is noteworthy that, after an initial transitory, machines are controlled with a *Switch-off* policy, i.e., $\tau^*_{\text{off}} = 0$ and $\tau^*_{\text{on}} = \infty$, except for machine $m = 3$ that is kept always on. Mainly, this is motivated by the short startup duration of the machines ($t_{\text{su}} = 6$ s). Table VIII reports a comparison among energy consumption with AO and IP policies. Despite the policy is applied locally, the overall effect is promising.

TABLE VII: Production line input data ($w_q = 0$ for all buffers) and algorithm parameter C_0. Machines $m = 5, 6$ do not work.

	Power [kW]			Time [s]			
m	w_1	w_2	w_3	t_{su}	MTBF	MTTR	C_0 [kJ]
1	3.6	14.4	8.4	6	6000	480	10
2	1.5	6	3.5	6	3600	180	2.5
3	3	12	7	6	4800	540	1
4	5.1	20.4	11.9	6	7920	600	1.5
7	1.8	7.2	4.2	6	6480	360	5
8	1.5	6	3.5	6	6000	300	3.5

VIII. Conclusions and Future Developments

Numerical results show that an on-line approach can be advantageous while off-line approaches are still in the learning phase. Also, it has been shown that the proposed on-line approach can effectively be applied in real cases by the use of a general estimation method (KDE). By tuning the algorithm parameters, the proposed policy is able to cover from the risk of unexpected high energy consumption and to limit the number of changes in control parameters over time.

Currently, the computational time required when the number of observations increases might be significant. Therefore, future effort will be devoted to decrease the response time of the algorithm. The proposed algorithm can adapt to not-unimodal distributions of the idle times. However, new policies should be evaluated to better cope with multiple modes of the stochastic process. Future development will also be devoted to include a target production rate and non homogeneous arrival processes such that the distribution might change in time and new features should be included in the algorithm to properly adjust the control. In this sense, the control problem might be addressed within a dynamic programming framework since decisions occur in time and sub-problems can be solved in a recursive manner. The effect of the policy at system level will be studied on an extensive set of case studies, both in terms of energy and productivity criteria.

Acknowledgment

The research has been partially funded by Regione Emilia under the project: Hi-score.it - Progetto POR-FESR 2014-2020, ASSE 1 Ricerca e Innovazione - Azione 1.2.2.

TABLE VIII: Numerical results for AO and IP policies applied at machines of a production line (8 replications).

Energy [kJ/part] at machine m	AO ($n = 500$)	AO ($n = 3000$)	AO ($n = 5600$)	IP ($n = 500$)	IP ($n = 3000$)	IP ($n = 5600$)
$m = 1$	257±19	523±21	549±27	271±17	379±14	349±9
$m = 2$	167±8	232±10	238±12	176±7	214±14	191±17
$m = 3$	8±11	4±2	2±1	14± 15	5± 2	2± 1
$m = 4$	74±17	94±27	107±34	82±15	72±14	76±11
$m = 7$	306±20	295±10	295±12	274±31	206±20	195±20
$m = 8$	255±15	246±9	246±11	256±29	205±27	178±20
Line Energy [kJ/part]	1067±90	1394±80	1438±97	1073±80	1083±36	991±40
PR [part/h]	25.21± 0.77	25.8±0.49	25.98±0.13	24.05±0.83	25.39±0.54	25.64±0.13

References

[1] Dahmus, J.B., Gutowski, T.G., (2010). An environmental analysis of machining, Manufacturing Engineering and Materials Handling Engineering, 643–652.

[2] Gahm, C. Denz, F., Dirr, M. and Tuma, A., (2016). Energy-efficient scheduling in manufacturing companies: a review and research framework. European Journal of Operational Research, 248(3), 744–757.

[3] Frigerio, N. and Matta, A., (2015). Energy-efficient control strategies for machine tools with stochastic arrivals. IEEE Transactions on Automation Science and Engineering, 12(1), 50–61.

[4] Frigerio, N. and Matta, A., (2014). Energy efficient control strategy for machine tools with stochastic arrivals and time dependent warm-up. Procedia CIRP, 15, 56–61.

[5] Fysikopoulos, A., Pastras, G., Vlachou, A. and Chryssolouris, G., (2014). An approach to increase energy efficiency using shutdown and standby machine modes, IFIP Advances in Information and Communication Technology AICT-439 (Part II), 205–212.

[6] Li, W., Zein, A., Kara, S., and Herrmann, C., (2011). An Investigation into Fixed Energy Consumption of Machine Tools. 18th CIRP Conference on LifeCycle Engineering, Braunschweig, 268–273.

[7] Weinert, N., Chiotellis, S. and Seliger. G., (2011). Methodology for planning and operating energy-efficient production systems, CIRP Annals - Manufacturing technology, 60(1), 41–44.

[8] Mouzon, G., Yildirim,M.B. and Twomey, J., (2007). Operational methods for minimization of energy consumption of manufacturing equipment. International Journal of Production Research, 45(18-19), 4247–4271.

[9] Frigerio, N. and Matta, A., (2016). Analysis on energy efficient switching of machine tool with stochastic arrivals and buffer information. IEEE Transactions on Automation Science and Engineering, 13(1), 238–246.

[10] Li, Y., Chang, Q., Ni. J. and Brundage, M., (2018), Event-Based Supervisory Control for Energy Efficient Manufacturing Systems. IEEE Transactions on Automation Science and Engineering, 15(1), 92–103.

[11] Su, W., Xie, X., Li, J., Zheng, L., Feng, L., (2017). Reducing energy consumption in serial production lines with Bernoulli reliability machines, International Journal of Production Research, 55(24), 7356–7379.

[12] Wang, J., Feng, Y., Fei, Z., Li, S., and Chang, Q., (2017). Markov chain based idle status control of stochastic machines for energy saving operation, IEEE International Conference on Automation Science and Engineering, 1019-1023, Xian, P.R. China.

[13] Jia, Z., Zhang, L., Arinez, J. and Xiao, G., (2016). Performance analysis for serial production lines with Bernoulli machines and real-time WIP-based machine switch-on/off control. International Journal of Production Research, 54(21), 6285–6301.

[14] Chen, G., Zhang, L., Arinez J., Biller, S., (2011). Feedback control of machine startup for energy-efficient manufacturing in Bernoulli serial lines, IEEE International Conference on Automation Science and Engineering, Trieste, Italy, 666–671.

[15] Guo, X., Zhou, S., Niu, Z., Kumar, P., (2013). Optimal wake-up mechanism for single base station with sleep mode, IEEE Int. Teletraffic Congress, 1–8.

[16] Maccio, V.J. and Down, D.G., (2015). On optimal policies for energy-aware servers. Performance Evaluation, 90, 36–52.

[17] Frigerio, N., and Matta, A., (2019). Analysis of Production Lines with Switch-off/on Controlled Machines, Thiede S., Herrmann C. (eds) Eco-Factories of the Future, Springer International Publishing.

[18] Brundage, M.P., Chang, Q., Li, Y., Xiao, G. and Arinez, J., (2014). Energy efficiency management of an integrated serial production line and HVAC system, IEEE Transactions on Automation Science and Engineering, 11(3), 789–797.

[19] Marzano, L., Frigerio, N., Matta, A., (2019). Energy Efficient State Control of Machine Tools: a Time-Based Dynamic Control Policy, IEEE International Conference on Automation Science and Engineering, Vancouver, Canada, 596–601.

[20] Powell, W. and Ryzhov, I., (2012). Optimal learning, Wiley.

[21] Munkhammar, J., Mattsson, L. and Ryden, J., (2017). Polynomial probability distribution estimation using the method of moments, PloS one, 12(4), e0174573. doi:10.1371/journal.pone.0174573.

[22] Cramer, H., (1946). Mathematical Methods of Statistics, Princeton University Press.

[23] Rosenblatt, Murray, (1956). Remarks on Some Nonparametric Estimates of a Density Function. Annals of Mathematical Statistics, 27(3), 832–837.

[24] Parzen, Emanuel, (1962). On Estimation of a Probability Density Function and Mode. Annals of Mathematical Statistics, 33(3), 1065–1076.

[25] Bowman, A., (1984). An alternative method of cross-validation for the smoothing of density estimates, Biometrika, 71(2), 353–360.

[26] Law A., Simulation Modeling and Analysis, McGraw Hill, (2015).

[27] Wang, J., Fei, Z., Chang, Q. et al., (2019). Multi-state decision of unreliable machines for energy-efficient production considering work-in-process inventory. International Journal of Advanced Manufacturing Technology, 102, 1009-1021.

Nicla Frigerio She obtained the title of PhD in Mechanical Engineering from Politecnico di Milano in 2016 with a thesis focused on the energy efficient control of manufacturing systems. Currently, she is Assistant Professor (RTDa) at the Manufacturing and Production Systems Research Line of Politecnico di Milano Mechanical Engineering Department. She collaborated with Purdue University (USA), and with Shanghai Jiao Tong University, Shanghai (China). She is a member of AITeM Italian Association of Manufacturing Technologies and of IEEE - Institute of Electrical and Electronics Engineers.

Lorenzo Marzano He received the M.Sc. degree in Mechanical Engineering from Politecnico di Milano, Milan, Italy, in 2018. His research was focused on energy-efficient policies for switching off/on manufacturing equipment with particular interest in learning algorithms and estimation methods. He is currently working with a consultancy company.

Andrea Matta He is Full Professor of Manufacturing at Department of Mechanical Engineering of Politecnico di Milano, Italy. He graduated in Industrial Engineering at Politecnico di Milano where he develops his teaching and research activities since 1998. He was Distinguished Professor at the School of Mechanical Engineering of Shanghai Jiao Tong University, China, from 2014 to 2016. He was awarded with the Shanghai One Thousand Talent and Eastern Scholar in 2013. His research area includes analysis, design and management of manufacturing and health care systems. He is Editor–in–Chief of Flexible Services and Manufacturing journal. He has published more than 130 papers on international and national journals/conference proceedings.