



POLITECNICO  
MILANO 1863

DIPARTIMENTO DI MECCANICA



## An adaptive policy for on-line Energy-Efficient Control of machine tools under throughput constraint

Frigerio, N.; Cornaggia, C. F. A.; Matta, A.

This is a post-peer-review, pre-copyedit version of an article published in JOURNAL OF CLEANER PRODUCTION. The final authenticated version is available online at:

<http://dx.doi.org/10.1016/j.jclepro.2020.125367>

This content is provided under [CC BY-NC-ND 4.0](https://creativecommons.org/licenses/by-nc-nd/4.0/) license



# An Adaptive Policy for On-Line Energy-Efficient Control of Machine Tools Under Throughput Constraint

Nicla Frigerio<sup>a,\*</sup>, Claudio FA Cornaggia<sup>a</sup> and Andrea Matta<sup>a</sup>

<sup>a</sup>Department of Mechanical Engineering, Politecnico di Milano, p.za Leonardo da Vinci, 20156, Milano, Italy

## ARTICLE INFO

**Keywords:**  
energy efficiency  
optimal control  
machine learning  
manufacturing automation

## ABSTRACT


Controlling the machine power state by switching off/on the machine when idle is one of the most promising energy efficient measure for machining processes. Part arrival process is affected by uncertainty and acquiring knowledge to obtain a proper and updated control model is difficult in industrial practice. Hence, control policies should be connected to the shop floor exploiting data acquired on-line. This work extends an on-line time-based policy recently proposed in the literature by including constraints on machine performance. A novel optimization algorithm is proposed to minimize energy consumption while assuring a target production rate and mitigating the risk of incurring in unexpected high energy consumption. Moreover, the policy is also broadened to autonomously adapt the control when the arrival process is non-stationary in time. The benefits of the proposed algorithms are assessed by means of realistic **simulated** cases and are around 25% of the energy consumed in idle states. Differently from existing studies dealing with the off-line problem, the proposed algorithm learns on-line while acquiring information from the real system.

Symbol	Description
$t_p$	mean processing time
$s = \{b, id, sb, su\}$	machine states: $b$ busy, $id$ idle, $sb$ standby, $su$ startup
$w_s$	machine power in state $s$
$X$	machine idle time with pdf $f_X(x)$ and CDF $F_X(x)$
$t_a$	mean idle time
$t_{su}$	startup duration
$w_q$	holding power
$\tau = \{\tau_{off}, \tau_{on}\}$	vector of machine control parameters
$\varphi(x, \tau)$	machine energy demand
$\psi(x, \tau)$	part waiting time
$\phi(x, \tau)$	energy consumed in a cycle
$g(\tau)$	expected energy consumed in a cycle
$H(\tau)$	expected waiting time
$\theta(\tau)$	expected throughput
$\mathbf{x} = \{x_i   i = 1, \dots, n\}$	vector of $n$ observed data
$\hat{f}_X(x \mathbf{x})$	estimated distribution of machine starvation times
$\varepsilon$	maximum admissible throughput reduction
$\delta$	energy risk probability limit
$PES$	percentage energy saving
$PTL$	percentage throughput loss
$ER$	energy risk

**Table 1:** Notation table

1

\*Corresponding author

 nicla.frigerio@polimi.it (N. Frigerio)

ORCID(s): 0000-0001-8146-9772 (N. Frigerio); 0000-0003-3902-2007

(A. Matta)

<sup>1</sup>Abbreviation list: EEC Energy Efficient Control; EES Energy Efficient Scheduling; iid independent and identically distributed; pdf probability density function; CDF Cumulative Density Function; AO Always On; SP Switching Policy; KDE Kernel Density Estimator; CPD Change Point Detection; RuLSIF Relative unconstrained Least-Squares Importance Fitting; CNC Computer Numeric Control.

## 1. Introduction

According to the U.S. Energy Information Administration (2019), the amount of energy absorbed by the industrial sector in 2018 accounts for more than 50% of the world energy consumption. Looking to the future, the long-term projections provided show that the gross output from industrial activities is supposed to double between 2018 and 2050, resulting in an increase of industrial energy consumption. Further, manufacturing plants are facing increasing pressure to reduce their carbon footprint, driven by concerns related to energy costs and climate change.

In addition, the topic of energy efficiency in manufacturing has gained an increasing prominence within the scientific community. As one of the most promising measure for machine tools, a proper control of machines standby can be applied to improve energy efficiency (Zhou et al. (2016), Sihag and Sangwan (2020)). Herewith, we refer to this control as Energy-Efficient Control (EEC). According to a recent EU report on energy efficiency and energy saving potential in industry (Chan et al. (2015)), a total of 91 energy saving opportunities which are technically feasible for the sector were accounted. Among them, integrated control systems account for the 14% of the total sector technical potential to obtain high efficiency equipment and the increased uptake of energy management systems accounts for an additional 4%. EEC potential is part of such share.

In the last decade, several machine control strategies have been defined to properly switch off/on production resources while not working on parts. The selection of an effective control is not trivial because a startup procedure is commonly required to resume the operational readiness of a machine tool. EEC strategies must be properly tuned to pursue energy efficiency while satisfying one or more criteria (e.g., target service level).

Nowadays machine tools have power saving (standby) modes to be used when the part flow is interrupted; nevertheless, very frequently, the standby mode is not exploited and machine tool users do not profit of the available standby modes. When machine standby is used, the control parameters are selected manually according to simple experience-based analyses. Commonly, parameters are equal on all machines and are not updated when the production changes resulting in a lack of effectiveness or efficiency.

The complexity and the variability of real manufacturing systems are difficult to be handled in the practice. Indeed, the part arrival process at a certain machine is stochastic due to process variability, failures to upstream processes and machines, or production changes and may vary in time. To cope with this intrinsic unknown stochasticity, an effective energy efficient policy requires an experimental campaign that allows for the fitting of the control model. This process can require high effort for a practitioner and it must be repeated for each controlled resource in correspondence of any production change. Due to the high management effort, EEC strategies are rarely applied or do not reach their full potential.

This work deals with the on-line EEC of machine state during idle periods and we specifically refer to machine tools executing machining operations, although the approach is flexible and can be applied also to other resources. The control model is created by extracting information from data concurrently collected from the shop floor. Careful attention is devoted to the identification of a solution that also addresses a production target. The exploitation of on-line data learning approaches would allow to better cope with the intrinsic unknown stochasticity of production systems, being autonomous and flexible. The proposed approach and algorithms reduce the barriers for a practical implementation.

## 2. Literature and Contribution

A key towards cleaner manufacturing is to apply measures to improve production energy efficiency. Due to the wide range of manufacturing activities, technologies and industries, strategic measures may be applied at different levels and may affect different manufacturing layers. A systematic overview has been proposed by Dufloy et al. (2012) where measures at different levels are discussed. Similar reviews can be found in Devoldere et al. (2007), Zhou et al. (2016), Yoon et al. (2015), and Sihag and Sangwan (2020).

Focusing at machine level, improvements can be achieved with four main measures as in Figure 1. The reuse of energy includes kinetic energy recovery and thermal management. Machine design improves energy transformations and machine components (e.g., friction reduction, weight minimization). Further, to reduce energy in input includes the control of processes and the control of machine states. Indeed, machine tools consume energy while working on parts, i.e., *process-related energy*, but also while the machine is idle (Dahmus and Gutowski (2004), Yoon et al. (2015)). The *non processing energy* is related to some machine components that keep executing their functions although the machine is not producing. For example, auxiliaries (e.g., chiller unit, hydraulic unit) allow to keep the machine in a ready-for-process state enabling machine tool cooling/heating, waste handling, and other machine conditions such that, whenever a part arrives, the part program can immediately start. The selection of proper process parameters (e.g., parameters optimization, trajectory improvement) or alternative processes (e.g., dry processes, different lubricants) aims at optimizing the process-related energy. The EEC of machine state instead aims at reducing the non processing energy consumed when the part flow is interrupted toward a better use of machine auxiliary equipment.

### 2.1. EEC related literature

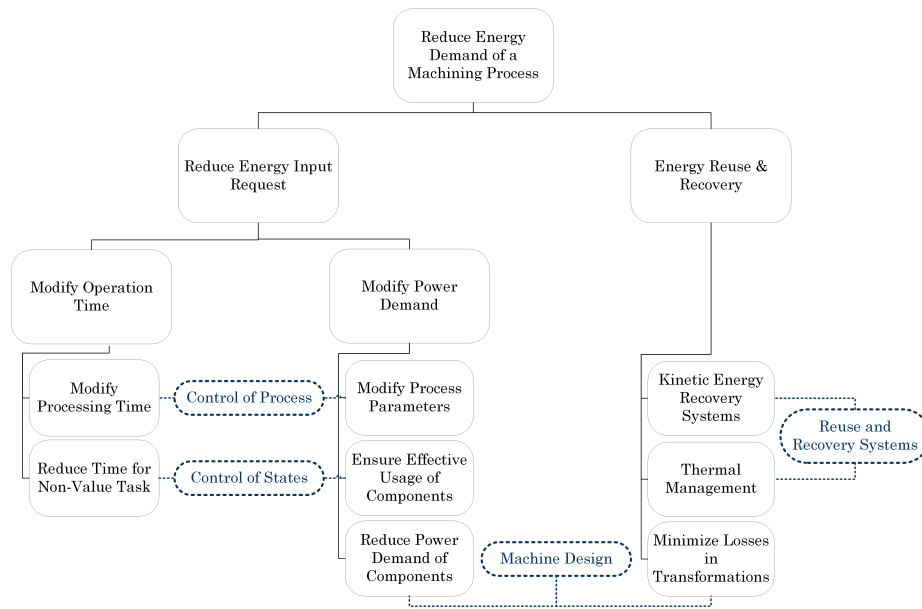


Figure 1: Classification of energy efficient measures at machine level.

In EEC policies, a *switch off* command is issued to deactivate some auxiliary units and the machine is triggered into a low power consumption state where the service is interrupted. Operational readiness is resumed with a *switch on* command. Often in manufacturing equipment, switch off/on transitions, i.e., *closedown* and *startup*, may require a certain amount of time impacting on system production rate.

EEC differs from Energy Efficient Scheduling (EES) because it refers to a different level in the production planning and control hierarchy. EES plans jobs schedule and off/modes to machines over a specific period of time before production starts. A recent and complete review on EES literature can be found in Gahm et al. (2016). Recent examples of EES can be found in Lee et al. (2017) and Materi et al. (2020). EEC provides switch off/on policies to be applied in real-time during production progress. State transitions are triggered with ad-hoc control rules, usually expressed as *IF-THEN* terms.

In EEC, control policies exploit several types of real-time information. *Buffer-based* policies introduce buffer occupancy thresholds such that machines can be controlled (Chen et al. (2011); Guo et al. (2013); Frigerio and Matta (2016); Duque et al. (2019); Jia et al. (2016); Wang et al. (2019)). *Time-based* policies set thresholds on machine idle time to define when service should be interrupted and resumed (Mouzou et al. (2007); Wang et al. (2013); Frigerio and Matta (2015); Squeo et al. (2019); Marzano et al. (under review)). Buffer-based policies have higher potential savings compared to time-based policies, but they are more complex to be developed and managed. Despite their simplicity, time-based policies are effective and can potentially be applied easily to a wide range of real-cases.

During control parameters (i.e., thresholds) optimization, the objective function always comprises an energy efficiency criterion. In addition, a production rate target is frequently set so as the EEC does not jeopardize the service level of the system (Chen et al. (2011); Guo et al. (2013); Frigerio and Matta (2015, 2016)). Other performance of interest can be also included, either as additional objectives or as constraints: as example, Frigerio and Matta (2015) included the number of switch off/on actions per produced part to limit machine wear.

5

EEC necessarily deals with the stochastic processes involved in the system. The literature assumes the stochastic processes to be known, solving the problem with an *off-line* approach. Hence, in all cited works, it is implicitly assumed that the probability distributions have been fitted on the basis of large data sets previously collected from the system. As exception, a recent work addresses the problem *on-line* (Marzano et al. (under review)). The proposed algorithm progressively fits machine starvation times from acquired data and controls the machine with properly optimized time thresholds.

## 2.2. Contribution

The literature analysis points out that EEC policies are generally formulated as off-line problems. Also, the need to enrich the optimization problem with constraints on performance of interest clearly emerges. The possibility to autonomously adapt the control in response to dynamic changes in the system is rarely addressed in EEC literature, despite being a clear requirement from industry.

An on-line time-based EEC policy is the focus of this paper, following the recent approach in Marzano et al. (under review) where machine requests are estimated through on-line collected data. The literature is extended in two directions

and the on-line problem is enriched by: (i) constraints related to production criteria and effectiveness of the control, (ii) the capability of adapting to arrival processes non-stationary in time.

A novel optimization algorithm is developed to select the optimal control when constraints limit the feasibility region for possible solutions. Specifically, a constraint on the expected throughput is included. Moreover, since control parameters are optimized based on expectations, an additional constraint accounts for the risk of deteriorating machine energy consumption over the single service request. Further, a change point detection method is included to identify variations in the part arrival process. This enables to autonomously adapt the control in response to the dynamic behavior of the upstream portion of the system.

The remainder of the paper is organized in 6 sections. Section 3 includes system description and model. The constrained on-line approach is described in Section 4 and Section 5 is devoted to the detection of changes in the stochastic process involved in the problem. Numerical results follow in Section 6 and Section 7 concludes the work.

### 3. System description and control policy

We consider a single machine whose state can be controlled for energy saving purposes. In the following, we describe the control policy applied and the model used.

#### 3.1. Machine states and assumptions

The machine is *busy* ( $b$ ) while working on parts, whilst it is *idle* ( $id$ ) in ready-for-process conditions. The machine can be triggered in a low power request state, i.e., the *standby* ( $sb$ ). Nevertheless, while in standby, the service is interrupted. The machine must pass through the *startup* ( $su$ ) state to switch from the standby state to the idle state. The startup is needed to resume the operational readiness of inactive components.

We assume that the machine is not saturated; hence, it might starve of raw parts. An upstream mechanism manages the arrival process at the machine and parts are released to the machine only when it is not busy. Further assumptions follow:

- The machine is never blocked.
- Processing times are assumed to be random variables with mean  $t_p$ .
- The power  $w_s$  requested in state  $s = \{b, id, sb, su\}$  is constant. Also,  $w_{su} > w_{id} > w_{sb} \geq 0$  and  $w_b > w_{id}$ . This assumption realistically represents manufacturing equipment that commonly requires high power.
- The duration of the startup  $t_{su}$  is deterministic.
- Machine idle times are iid and represented by the random variate  $X > 0$ , which is distributed accordingly to a probability density function (pdf)  $f_X(x)$  with mean  $t_a$ . The realization of  $X$  is denoted with  $x$ , which also constitutes the arrival time realization in a *cycle*.
- The holding power  $w_q$  is absorbed by the external equipment (e.g. part-handling system or heated/cooled buffer) for keeping the part until machine operational readiness is resumed.

#### 3.2. Time-based control policy

In common practice, the machines are not controlled for energy purposes and they are always kept idle while waiting for parts. This policy is known as *Always on* (AO) policy where, focusing on the time frame lasting from the departure of a part ( $t = 0$ ) until the departure of the next one, idle and busy states alternate.

The *Switching Policy* (SP) (Frigerio and Matta (2015)) is employed to control machine state for energy efficiency. The switch-off command occurs while the machine starves, and the switch-on occurs after a while or when the part arrives. Time-based thresholds  $\tau_{off}$  and  $\tau_{on}$  are used to trigger state transitions:

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

Controlled transitions are: (i) the switch off transition from the idle state to the standby state when  $t = \tau_{off}$ , and (ii) the switch on transition from the standby state to the startup state when  $t = \tau_{on}$  or upon part arrival  $t = x$ .

Note that the busy state, and its expected energy consumption  $w_b \cdot t_p$ , does not depend on the control parameters. Therefore, it is not included in our model. Let us define a *cycle* as the time frame lasting from the departure of a part ( $t = 0$ ) until the process of the next one begins.

Since the time spent in a certain state during a cycle is the output of a stochastic process, the *expected value of the energy consumed in a cycle* is the objective function to be minimized when tuning the control parameters  $\tau = \{\tau_{off}, \tau_{on}\}$ . In this respect, SP is effectively delaying the switch-off command when the probability of part arrival is high. Also, the switch-on command occurs when the part arrives or in advance to begin the startup before the actual arrival.

### 3.3. System model

We briefly describe the model used as from the literature (Frigerio and Matta (2015)). Given control parameters  $\tau$  and the occurrence of arrival in a cycle  $x$ , denote  $\varphi(x, \tau)$  the *machine energy demand* and  $\psi(x, \tau)$  the *part waiting time*. The *energy consumed in a cycle*  $\phi(x, \tau)$  is obtained by summing machine energy demand and the holding energy consumed to keep the part waiting for machine readiness:

$$\phi(x, \tau) = \varphi(x, \tau) + w_q \cdot \psi(x, \tau). \quad (1)$$

Given starvation time distribution  $f_X(x)$ , the *expected value of the energy consumed in a cycle*  $g(\tau)$  and the *expected waiting time*  $H(\tau)$  are:

$$g(\tau) = \mathbb{E}_X[\phi(x, \tau)] = \int_0^\infty \phi(x, \tau) f_X(x) dx$$

$$H(\tau) = \mathbb{E}_X[\psi(x, \tau)] = \int_0^\infty \psi(x, \tau) f_X(x) dx.$$

As a consequence, the *expected throughput*  $\theta(\tau)$  can be expressed as:

$$\theta(\tau) = \frac{1}{t_a + t_p + H(\tau)}.$$

Trivially, with AO policy  $g_{AO} = g(\infty, \infty) = w_{id} \cdot t_a$  and  $H_{AO} = H(\infty, \infty) = 0$ . For more details on the model, refer to Appendix A.

## 4. On-line time-based control policy with constraints

In *off-line* policies, the pdf  $f_X(x)$  is assumed to be known; whilst in *on-line* policies, an estimator of the pdf is provided, given a set of  $n$  observed arrivals  $\mathbf{x} = \{x_i | i = 1, \dots, n\}$ . The on-line approach, firstly proposed in Marzano et al. (under review), iteratively addresses three phases:

1. A *learning phase*, in which the non-parametric Kernel Density Estimation (KDE) method (Parzen (1962)) is used to provide the estimated distribution of machine starvation times  $\hat{f}_X(x|\mathbf{x})$ . More details are provided in Appendix B.
2. An *optimization phase*, which solves the *on-line EEC problem* and provides the control  $\tau_n = \{\tau_{\text{off},n}, \tau_{\text{on},n}\}$ .
3. An *implementation phase*, which focuses on the uncertainty of the estimates and is employed to decide whether the new control parameters  $\tau_n$  should be applied or if it is better to wait for more observations to accumulate.

This work focuses on the optimization and the implementation phases. The on-line EEC problem is enriched with constraints related to the effects of the control on system performance (Section 4.1), and a novel optimization algorithm is presented (Section 4.2). Further, besides improving the overall robustness of the implementation phase, an additional condition is introduced to ensure that the implemented control always satisfies the constraints (Section 4.3).

### 4.1. Constrained optimization problem

The following *on-line EEC problem* is proposed:

$$\tau_n = \arg \min_{\tau} \hat{g}(\tau) \quad (2)$$

$$\text{Subject to: } \hat{\theta}(\tau) \geq (1 - \varepsilon) \hat{\theta}_{AO} \quad (3)$$

$$\hat{\mathbb{P}}[\phi(x, \tau) > \phi_{AO}(x)] \leq \delta \quad (4)$$

$$\tau_{\text{on}} > \tau_{\text{off}} \quad (5)$$

$$\tau_{\text{off}}, \tau_{\text{on}} \in \mathbb{R}_0^+ \quad (6)$$

where estimator  $\hat{f}_X(x|\mathbf{x})$  is used to obtain expected system performance and probabilities.

The objective function (2) to be minimized is the expected value of the energy consumed in a cycle  $\hat{g}(\tau)$ . Constraint (3), referred to as *throughput constraint*, sets a lower bound for the expected throughput  $\hat{\theta}$ , where  $\varepsilon \in [0, 1]$  is the maximum admissible throughput reduction with respect to that achieved by the AO policy, i.e.,  $\hat{\theta}_{AO} = 1/(t_p + \hat{t}_a)$ . The *energy risk constraint* (equation (4)) limits up to  $\delta \in [0, 1]$  the probability of incurring in an energy consumption higher than that achieved by the AO policy, i.e.,  $\phi_{AO}(x) = w_{id} \cdot x$ . This is related to starvation time distribution tails which might incur in high energy consumption within a cycle. Constraint (5) and (6) define the domain of decision variables.



#### 4.1.1. A lower bound for $\tau_{on}$

Once the machine is switched off at  $\tau_{off}$  the machine should stay in the standby state for a minimum sojourn time such that the policy can be advantageous, resulting in a lower bound for control parameter  $\tau_{on}$ .

Therefore, the feasible set of solutions defined by problem constraints (5) and (6) can be successfully reduced by leveraging the following property:

**Property 4.1.** Given  $\tau_{off}$ , the switch on transition must be triggered at:

$$\tau_{on} > \tau_{off} + \xi \quad (7)$$

where  $\xi$  is a critical sojourn duration defined as follows:

$$\xi \stackrel{\text{def}}{=} \frac{w_{su} - w_{id}}{w_{id} - w_{sb}} t_{su}. \quad (8)$$

*Proof.* The proof of property 4.1 is carried out graphically comparing function  $\phi(x, \tau)$  and the function obtained with AO policy, i.e.,  $\phi_{AO}(x)$ .

Trivially, in order to be effective, control parameters must ensure that  $\phi(x, \tau) < \phi_{AO}(x)$  for a certain set of occurrences  $x \in \Omega$ ; hence set  $\Omega$  must be not empty.

Referring to Figure 2, denote  $P1$  the point with  $x = \tau_{on} + t_{su}$  on the SP curve, and  $P2$  that on AO curve. In Figure 2a,  $P1$  is above  $P2$  and  $\Omega \equiv \emptyset$ . In Figure 2b,  $P1$  is below  $P2$  and  $\Omega \equiv \{x > \tilde{x}\}$  where  $\tilde{x}$  is the abscissa of the intersection among the two curves  $\phi(x, \tau)$  and  $\phi_{AO}(x)$ .

The critical condition is obtained when  $P1$  and  $P2$  coincide; therefore:

$$\phi(\tau_{on} + t_{su}, \tau) = \phi_{AO}(\tau_{on} + t_{su}). \quad (9)$$

From equation (9), we obtain:

$$w_{id} \cdot \tau_{off} + w_{sb}(\tau_{on} - \tau_{off}) + w_{su} \cdot t_{su} = w_{id}(\tau_{on} + t_{su})$$

$$\tau_{on} = \tau_{off} + \frac{w_{su} - w_{id}}{w_{id} - w_{sb}} t_{su}$$

By implication, the *critical sojourn duration in the standby state*  $\xi$  is as in equation (8) and the proof is completed.  $\square$

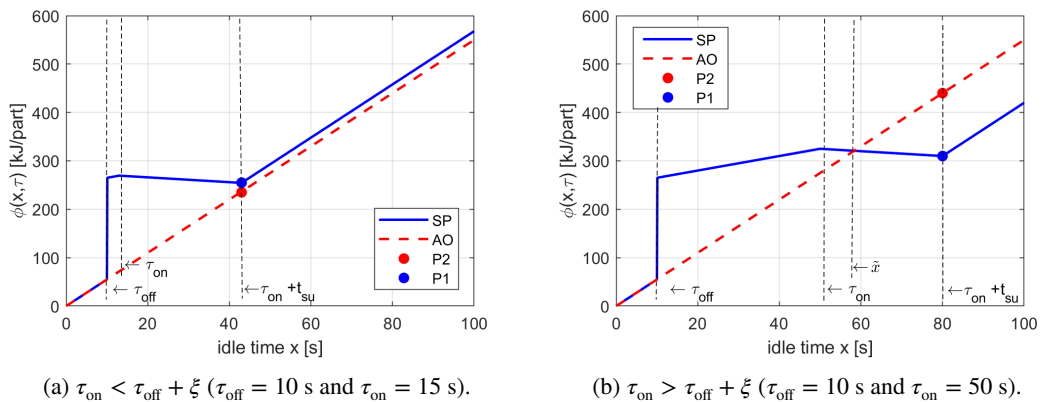
The new constraint (7) states that the planned sojourn duration in the standby state (i.e.,  $\tau_{on} - \tau_{off}$ ) must ensure that the energy consumed in the standby and startup states is lower than what would have been absorbed if the machine had not been controlled. In other words, given  $\tau_{on} > \tau_{off} + \xi$ , we obtain:

$$\phi(\tau_{on} + t_{su}, \tau) < \phi_{AO}(\tau_{on} + t_{su})$$

$$w_{sb}(\tau_{on} - \tau_{off}) + w_{su} \cdot t_{su} < w_{id}(\tau_{on} + t_{su} - \tau_{off}).$$

The domain of decision variables  $\mathcal{U}$  is obtained by combining equations (6) and (7):

$$\mathcal{U} \stackrel{\text{def}}{=} \{\tau \in \mathbb{R}^2 \mid \tau_{on} > \tau_{off} + \xi, \tau_{off} \geq 0\}. \quad (10)$$



**Figure 2:** Energy consumed per cycle under AO and SP: case where  $P2 < P1$  (a), and case  $P2 > P1$  (b)

. Machine data are:  $w_{id} = 5.5$  kW,  $w_{sb} = 1.5$  kW,  $w_{su} = 6.5$  kW,  $t_{su} = 30$  s, and  $w_q = 0.5$  kW.

#### 4.1.2. Throughput constraint analysis

Constraint (3) can be equivalently expressed as a constraint on the estimated expected part waiting time  $\hat{H}(\tau)$ :

$$\hat{H}(\tau) \leq \hat{H}_{\max} \stackrel{\text{def}}{=} \frac{\varepsilon}{1-\varepsilon} (t_p + \hat{t}_a). \quad (11)$$

As in Frigerio and Matta (2015),  $\hat{H}(\tau)$  monotonically decreases over  $\tau_{\text{off}}$  and monotonically increases over  $\tau_{\text{on}}$ . Trivially, the maximum waiting time occurs for  $\tau = \{0, \infty\}$ , when the startup is triggered by part arrival and every part waits  $t_{\text{su}}$  before being processed. Hence, for  $\hat{H}_{\max} \geq t_{\text{su}}$  constraint (11) is not binding.

#### 4.1.3. Energy risk constraint analysis

With reference to Figure 2b, constraint (4) can be computed as follows:

$$\hat{F}_X[\tilde{x}(\tau)|\mathbf{x}] - \hat{F}_X(\tau_{\text{off}}|\mathbf{x}) \leq \delta \quad (12)$$

where  $\hat{F}_X(x|\mathbf{x})$  is the estimated cumulative density function of the starvation times, while  $\tilde{x}(\tau)$  indicates the abscissa of the intersection among  $\phi(x, \tau)$  and  $\phi_{\text{AO}}(x)$  (see Figure 2b). Moreover, the following property holds:

**Property 4.2.** *The probability  $\hat{\pi}(\tau) = \hat{\mathbb{P}}[\phi(x, \tau) > \phi_{\text{AO}}(x)]$  is non-decreasing in  $\tau_{\text{on}}$  and reaches the maximum in  $\tau_{\text{on}} = \tau_{\text{off}} + \varpi$ . Where  $\varpi$  is defined as:*

$$\varpi \stackrel{\text{def}}{=} \frac{w_{\text{su}} + w_q}{w_{\text{id}} - w_{\text{sb}}} t_{\text{su}}.$$

*Proof.* Based on equation (12), the partial derivative of  $\hat{\pi}(\tau)$  with respect to  $\tau_{\text{on}}$  is non negative. In details:

$$\frac{\partial \hat{\pi}(\tau)}{\partial \tau_{\text{on}}} = \frac{\partial \hat{F}_X(x|\mathbf{x})}{\partial x} \Big|_{x=\tilde{x}(\tau)} \cdot \frac{\partial \tilde{x}(\tau)}{\partial \tau_{\text{on}}}. \quad (13)$$

The first term is non-negative by definition. Whilst, for the second term, we need to analyze the intersection abscissa  $\tilde{x}(\tau)$ . Firstly,  $\tilde{x}(\tau) = \infty$  if  $\tau_{\text{on}} < \tau_{\text{off}} + \xi$ ; nevertheless, this case never happens when constraint (7) is verified. In other cases, solving  $\phi(x, \tau) = \phi_{\text{AO}}(x)$  for  $x > \tau_{\text{off}}$  we obtain:

$$\tilde{x}(\tau) = \begin{cases} \frac{w_{\text{id}} \cdot \tau_{\text{off}} + w_{\text{su}} \cdot t_{\text{su}} + w_{\text{sb}}(\tau_{\text{on}} - \tau_{\text{off}}) + w_q(\tau_{\text{on}} + t_{\text{su}})}{w_{\text{id}} + w_q} & \text{if } C_1 \\ \tau_{\text{off}} + \varpi & \text{if } C_2 \end{cases} \quad (14)$$

with conditions  $C_i | i = 1, 2$  depending on  $\tau$ . We have:

- $C_1$  :  $\tau_{\text{on}} \in (\tau_{\text{off}} + \xi; \tau_{\text{off}} + \varpi]$ ;
- $C_2$  :  $\tau_{\text{on}} > \tau_{\text{off}} + \varpi$ .

According to equation (14), the abscissa of intersection  $\tilde{x}(\tau)$  is trivially non-decreasing in  $\tau_{\text{on}}$ , and there exists a limit sojourn duration in the standby state  $\varpi$  beyond which  $\tilde{x}(\tau)$  is constant in  $\tau_{\text{on}}$ . As a consequence, the second term of equation (13) is non-negative and the proof is completed.  $\square$

Trivially, the maximum probability occurs for  $\tau_{\text{on}} = \tau_{\text{off}} + \varpi$  and when it happens to be smaller than  $\delta$ , constraint (4) is not binding.

## 4.2. Solving algorithm

The solving algorithm is now detailed, whose rationale is to select the best solution among:

1. The local solutions of the unconstrained optimization problem, obtained relaxing (3) and (4);
2. The minimum of  $\hat{g}(\tau)$  along the boundary of the feasibility region.

It is noteworthy that, in the practice, constraints (3) and (4) does not have the same importance and are rarely addressed simultaneously. In most of practical problems, the fulfillment of a target throughput is the most important requirement, hence constraint (4) becomes less important and could be relaxed. Vice versa, in other cases, constraint (3) could be relaxed giving priority to constraint (4). Hence, the proposed algorithm can be used to address the throughput constraint (3) and the energy risk constraint (4) separately, despite the main idea is common.

Let us consider the optimization problem in equations (2)–(6) relaxing constraint (4). We obtain:



**Step 1** Solve the unconstrained problem, minimizing the objective function (2) within set  $\mathcal{U}$ , see equation (10). Denote the optimal solution with  $\tau_{unc}$ . The algorithm in Appendix C is used (Frigerio and Matta (2015)).

**Step 2** If the following condition holds:

$$\hat{H}(\tau_{unc}) \leq \hat{H}_{max},$$

the unconstrained solution is feasible and the algorithm stops giving  $\tau_n = \tau_{unc}$ . Otherwise, create set  $D_{con} \subset D_{unc}$  of feasible solutions using constraint (3) and continue.

**Step 3** Define problem boundary  $\Theta$  as the set of points meeting the throughput constraint (3) at equality:

$$\Theta \stackrel{\text{def}}{=} \{\tau \in \mathcal{U} \mid \hat{H}(\tau) = \hat{H}_{max}\}.$$

**Step 4** Select problem solution  $\tau_n$ :

$$\tau_n = \min_{\tau \in \{D_{con} \cup \Theta\}} \hat{g}(\tau). \quad (15)$$

The flowchart of the optimization algorithm is in Figure 3.

We reformulate Step 2 and Step 3 while considering constraint (4) instead of (3). We obtain:

**Step 2 (II)** If the following condition holds:

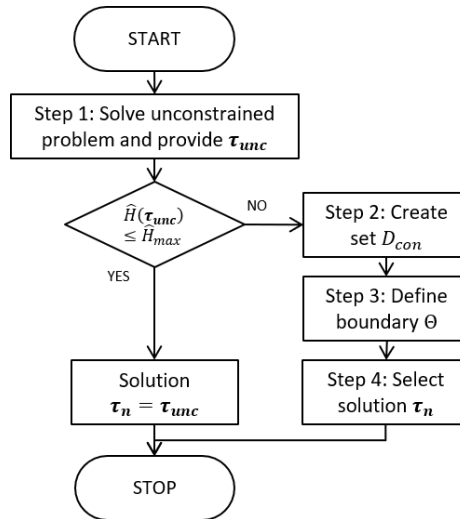
$$\hat{\mathbb{P}}[\phi(x, \tau_{unc}) > \phi_{AO}(x)] \leq \delta,$$

the unconstrained solution is feasible and the algorithm stops giving  $\tau_n = \tau_{unc}$ . Otherwise, create set  $D_{con} \subset D_{unc}$  of feasible solutions using constraint (4) and continue.

**Step 3 (II)** Define problem boundary  $\Phi$  as the set of points meeting the energy risk constraint (4) at equality:

$$\Phi \stackrel{\text{def}}{=} \{\tau \in \mathcal{U} \mid \hat{\mathbb{P}}[\phi(x, \tau) > \phi_{AO}(x)] = \delta\}. \quad (16)$$

Trivially Step 4 finds problem solution  $\tau_n$  solving equation (15) constrained by  $\Phi$  (16).



**Figure 3:** Flowchart of the optimization algorithm when throughput constraint is considered. When risk constraint is considered, Step 2 and Step 3 are substituted with Step 2 (II) and Step 3 (II), respectively.

### 4.3. Implementation phase

Denote  $\tau_{impl,n-1}$  as the control parameters employed when observation  $n$  is collected (trivially  $\tau_{impl,0} = \{\infty, \infty\}$ ). At new iteration, new control  $\tau_n$  is found in the optimization phase given the updated pdf estimate  $\hat{f}_X(x|\mathbf{x})$ . The control applied to the machine is changed from  $\tau_{impl,n-1}$  to  $\tau_n$  if at least one of the following conditions holds:

- I1. Control parameters  $\tau_{impl,n-1}$  are not anymore feasible according to constraints (3) and (4);

I2. The new control parameters improve significantly the objective function.

Condition I1 checks the feasibility of control parameters  $\tau_{\text{impl},n-1}$ , considering the updated pdf estimate  $\hat{f}_X(x|\mathbf{x})$ . Since more observations are used, the variability on the estimate is reduced. I1 holds if at least one of the following requirements is met:

- $\hat{\theta}(\tau_{\text{impl},n-1}) < (1 - \epsilon)\hat{\theta}_{\text{AO}}$ ;
- $\hat{\mathbb{P}}[\phi(x, \tau_{\text{impl},n-1}) > \phi_{\text{AO}}(x)] > \delta$ .

Condition I2 focuses on the mean improvement  $\mu_Z$  obtained with control  $\tau_n$  with respect to the objective function obtained by  $\tau_{\text{impl},n-1}$ . Specifically,  $\mu_Z$  is the expected value of the following random variable  $Z$ :

$$Z = \phi(X, \tau_{\text{impl},n-1}) - \phi(X, \tau_n).$$

Being unknown,  $\mu_Z$  is estimated using the sample mean obtained from observations  $\mathbf{x}$ . I2 is verified if the null hypothesis  $H_0 : \mu_Z \leq c_n$  is rejected. Parameter  $c_n$  is a discounted implementation cost defined as:

$$c_n = c_0 \cdot (1 - \gamma)^n$$

where  $\gamma \in [0, 1]$  is a discount factor. Since random variable  $Z$  is generally unlikely to be iid normal, a  $(1 - \alpha)\%$  bias-corrected and accelerated bootstrap confidence interval is constructed to assess the null hypothesis (Efron (1987)). This confidence interval is very simple to form and not necessarily symmetric.

## 5. Variations in stochastic processes

Manufacturing environments are highly dynamic, e.g., switches between different production batches, unreliable resources, continuous improvement interventions. Therefore, the arrival of parts at a single machine is generally non-stationary in the long range and the on-line EEC should adjust in response to this dynamic behavior.

We assume that the starvation times distribution  $f_X(x)$  changes at unknown points in time, i.e., *change points*, and the approach is extended to ensure an effective adaptation. Hence, the algorithm is enriched with a *detection phase* to be executed before the *learning phase* to identify significant variations in  $f_X(x)$  (see Figure 4). As a consequence, once the *change point* is detected, the previous observations are discarded so that the estimates are adapted to the current stochastic process.

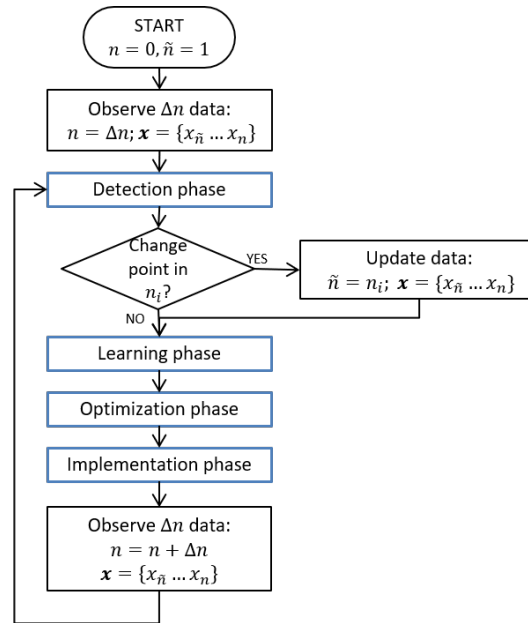


Figure 4: Flowchart of the on-line EEC algorithm. We assume  $\Delta n$  new data are observed at each iteration.

Several *Change Point Detection* (CPD) approaches are available in the literature (Aminikhanghahi and Cook (2016)). Among the unsupervised, on-line and non-parametric ones, we selected the *Relative Unconstrained Least-Squares Importance Fitting* (RuLSIF) method (Liu et al. (2013)). According to the related literature, RuLSIF method yields strong accuracy and robustness. Further, the Matlab implementation of the method is directly provided by the authors.

The idea behind RuLSIF is to compare the probability distributions of two data sets, respectively before and after a candidate observation  $x_i$ , and to quantify the dissimilarity with a *change point score*  $\sigma(x_i)$ . The higher  $\sigma(x_i)$ , the more likely  $x_i$  is a change point. The following conditions defines  $x_i$  as a change point:

$$\begin{cases} \sigma(x_{i-1}) < \sigma(x_i) \\ \sigma(x_i) > \sigma(x_{i+1}) \\ \sigma(x_i) \geq \eta_d \end{cases}$$

where  $\eta_d$  is a threshold to filter out false alarms.

In details, the method is illustrated in Figure 5. Let  $\mathbf{X}(i)$  be the subset of  $k_d$  observations starting with  $x_i$ :

$$\mathbf{X}(i) = [x_i, x_{i+1}, \dots, x_{i+k_d-1}].$$

Moreover, denote  $\mathcal{X}(i) \in \mathbb{R}^{k_d \times n_d}$  as the matrix of  $n_d$  consecutive subsets starting with  $\mathbf{X}(i)$  in the first column:

$$\mathcal{X}(i) = [\mathbf{X}(i)^T, \mathbf{X}(i+1)^T, \dots, \mathbf{X}(i+n_d-1)^T].$$

The RuLSIF method estimates the symmetrized  $\alpha_d$ -relative Pearson divergence ( $PE$ ) between  $\mathcal{X}(i)$  and  $\mathcal{X}(i-n_d)$  and returns  $\sigma(x_i)$  as:

$$\sigma(x_i) = \widehat{PE}_{\alpha_d}(P_{i-n_d} || P_i) + \widehat{PE}_{\alpha_d}(P_i || P_{i-n_d})$$

where  $P_{i-n_d}$  and  $P_i$  are the true probability distributions of samples  $\mathcal{X}(i-n_d)$  and  $\mathcal{X}(i)$ , and  $\widehat{PE}$  is the estimated  $PE$  based on collected data. As a property,  $\sigma(x_i) \in [0, \frac{1}{\alpha_d}]$  (Liu et al. (2013)).

RuLSIF parameters must be calibrated in order to ensure an effective and responsive CPD while reducing false alarms. In detail,  $n_d$  and  $k_d$  influence the responsiveness of the approach (i.e., detection delay). Specifically,  $n_d + k_d - 2$  observations must be inspected ahead of the candidate change point  $x_i$  to return  $\sigma(x_i)$ .  $\alpha_d$  defines the smoothness and the domain of the change point score  $\sigma(x_i)$ . Threshold  $\eta_d \in (0, \frac{1}{\alpha_d})$  is the most sensitive parameter, since it affects both false alarms and detection capability of the method. These parameters will depend on the specific application and user willingness at risk.

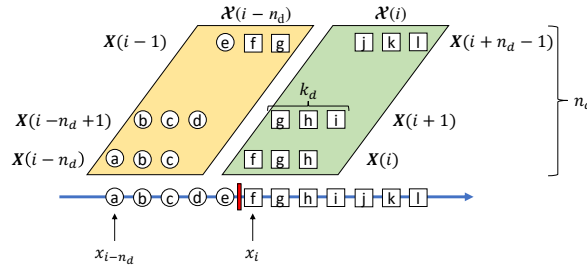


Figure 5: Illustrative example of RuLSIF method (Liu et al. (2013)).

## 6. Numerical results

The benefits of the proposed policy are analyzed in this section. Specifically, algorithm results are studied comparing the obtained solution with the AO policy. Given a set of  $n$  observed arrivals  $\mathbf{x} = \{x_i | i = 1, \dots, n\}$ , we define the following sample-based KPIs:

- Percentage energy saving:

$$PES = \frac{\sum_{i=1}^n \phi_{AO}(x_i) - \sum_{i=1}^n \phi(x_i, \tau_{impl, i-1})}{\sum_{i=1}^n \phi_{AO}(x_i)} \cdot 100;$$

- Percentage throughput loss:

$$PTL = \frac{\sum_{i=1}^n \psi(x_i, \tau_{impl, i-1})}{\sum_{i=1}^n [x_i + t_p + \psi(x_i, \tau_{impl, i-1})]} \cdot 100;$$

Scenario	Machine	$f_X(x)$	$\varepsilon$	$\delta$	$\tau^*$ [s]	$g(\tau^*)$ [kJ/part]	$\theta(\tau^*)$ [parts/h]	$\pi(\tau^*)$
M1-0-0 (AO)	M1	Erlang(3,0.037)	0	0	$\{\infty, \infty\}$	445.95	9.45	0
M1-1-1	M1	Erlang(3,0.037)	1	1	$\{0, \infty\}$	331.62 (-25.64)	8.76 (-7.30)	0.308
M1-0.05-1	M1	Erlang(3,0.037)	0.05	1	$\{0, 78.64\}$	348.85 (-21.77)	8.97 (-5.00)	0.308
M1-0.02-1	M1	Erlang(3,0.037)	0.02	1	$\{0, 32.42\}$	397.61 (-10.84)	9.26 (-2.00)	0.241
M1-1-0.27	M1	Erlang(3,0.037)	1	0.27	$\{0, 41.13\}$	383.45 (-14.01)	9.20 (-2.63)	0.270
M1-1-0.22	M1	Erlang(3,0.037)	1	0.22	$\{0, 25.78\}$	410.54 (-7.94)	9.30 (-1.53)	0.220

**Table 2:** Scenarios description and off-line results. The percentage relative difference against AO policy is reported within braces for  $g(\tau^*)$  and  $\theta(\tau^*)$ .

Machine	$w_{sb}$ [kW]	$w_{id}$ [kW]	$w_{su}$ [kW]	$w_q$ [kW]	$t_{su}$ [s]
M1	1.5	5.5	6.5	0.5	30
M2	0.52	5.35	6.08	1	24
M3	0.8	4.3	9.3	0	25

**Table 3:** Machine parameters.

- Energy risk:

$$ER = \frac{1}{n} \sum_{i=1}^n \mathcal{I}[\phi(x_i, \tau_{impl, i-1}) > \phi_{AO}(x_i)]$$

where indicator function  $\mathcal{I}[\cdot] = 1$  whenever the energy consumed in a cycle with the on-line control is higher than that obtained with the AO policy.

Several simulated scenarios are investigated varying the controlled machine, the arrival distribution and problem constraints. Despite the algorithm can generally iterate at each new observation, for computational saving it herewith iterates every  $\Delta n = 10$  collected observations. Further, each [computer simulation](#) experiment is replicated 10 times. The implementation phase works with the following setting:  $C_0 = 10$  kJ/part,  $\alpha = 0.05$ ,  $\gamma = 0.006$ .

The algorithm has been implemented in Matlab<sup>®</sup>R2018b and results have been obtained [by computer simulation](#) on a laptop with 2.4GHz i-5 Intel Core and 8GB RAM (1600MHz DDR3).

## 6.1. Results of the control policy with constraints

The goal of these {computer experiments is to compare the unconstrained solution with the constrained one. Evaluated scenarios are described in Table 2 where the code "M1- $\varepsilon$ - $\delta$ " includes the controlled machine (M1, with parameters as in Table 3 and deterministic processing time  $t_p = 300$  s), the starvation time distribution (Erlang-3 with rate  $\lambda = 0.037$  and mean  $t_a = 81.08$  s), the maximum throughput loss  $\varepsilon$  (see equation (3)) and the maximum risk  $\delta$  (see equation (4)). In more detail, we analyze the unconstrained scenario (M1-1-1), two scenarios with constrained throughput (M1- $\varepsilon$ -1), and two with constrained energy risk (M1-1- $\delta$ ), such that in each scenario the constraint is binding.

A production run of 2500 parts is considered. For comparison purpose, the collected starvation times have been sampled from  $f_X(x)$  using common random numbers (Law (2015)), such that the variability among scenarios is reduced.

### 6.1.1. A note on off-line results

The results of the off-line control problem, assuming the starvation time distribution is known, are collected in Table 2. Denote the theoretical optimal control of a certain policy with  $\tau^*$ . Note that the comparison with the off-line solution is always reported in next sections, despite the off-line SP cannot actually be applied while observing data.

The AO policy obtains an objective function value of  $g(\infty, \infty) = g_{AO} = 445.95$  kJ/part and an expected throughput of  $\theta(\infty, \infty) = \theta_{AO} = 9.45$  parts/h. Relaxing constraints (3) and (4), the off-line SP achieves 25.64% of savings on the objective function, a throughput loss of 7.30% with respect to the AO policy and the probability of having an increased energy is 0.308.

When the problem becomes constrained, the switch-on command is progressively anticipated reducing the amount of energy saving. Indeed for scenarios M1- $\varepsilon$ -1, the throughput is decreasing in  $\tau_{on}$  because of the probability that an incoming part finds the machine in the standby or startup state (Frigerio and Matta (2015)). For M1-1- $\delta$ , according to property 4.2, a similar consideration also holds for constraint (4).

Further, scenarios M1-1-1 and M1-0.05-1 result in the same energy risk ( $\pi(\tau^*) = 0.308\%$ ) because the planned sojourn duration in the standby state is greater than the limit duration  $\varpi = 52.50$  s ( $\tau_{off}^* = 0$ ). In other scenarios, probability  $\pi(\tau^*)$  is increasing in  $\tau_{on}$ .

Scenario	Performance	Observation number				
		$n = 50$	$n = 250$	$n = 500$	$n = 1000$	$n = 2500$
M1-0-0 (AO)	Energy [kJ/part]	455.37±22.07	445.06±12.79	445.56±7.42	446.75±6.48	445.54±3.11
	Throughput [parts/h]	9.41±0.10	9.45±0.06	9.45±0.03	9.44±0.03	9.45±0.01
M1-1-1	PES [%]	17.36±2.34	23.46±1.21	24.42±0.77	25.07±0.70	25.30±0.34
	PTL [%]	4.60±0.84	6.55±0.26	6.81±0.21	6.99±0.12	7.14±0.08
	ER	0.148±0.051	0.278±0.011	0.294±0.012	0.298±0.012	0.302±0.007
M1-0.05-1	PES [%]	15.93±2.15	20.18±1.26	21.21±0.84	21.84±0.63	21.84±0.44
	PTL [%]	3.56±0.90	4.69±0.29	4.80±0.17	4.90±0.09	5.00±0.05
M1-0.02-1	PES [%]	5.82±2.65	10.29±1.41	10.68±1.08	10.95±0.85	10.95±0.58
	PTL [%]	0.87±0.49	1.82±0.08	1.94±0.08	1.97±0.06	2.00±0.04
M1-1-0.27	PES [%]	10.37±4.46	16.31±3.12	15.72±2.36	15.25±1.77	14.66±1.18
	ER	0.112±0.059	0.252±0.013	0.264±0.012	0.268±0.008	0.269±0.005
M1-1-0.22	PES [%]	4.63±2.33	8.56±2.47	8.54±2.21	8.50±1.91	8.20±1.48
	ER	0.078±0.046	0.194±0.012	0.210±0.011	0.216±0.066	0.219±0.051

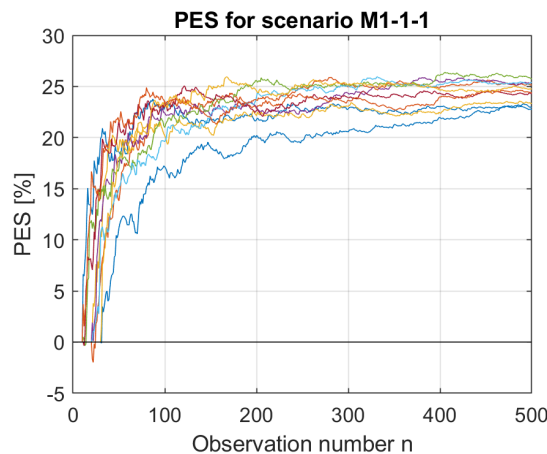
**Table 4:** Sample-based comparison of the on-line algorithm for different scenarios (95% CI of 10 replications).

### 6.1.2. On-line results

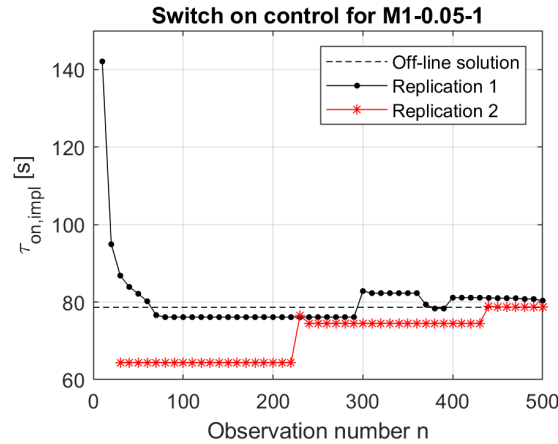
Scenarios described in Table 2 are herewith solved by applying the on-line approach while observations are collected. Results are thoroughly collected in Table 4 and show that the on-line approach achieves good performances in terms of *PES* despite a limited set of observations is available. This constitutes a significant advantage with respect to an off-line approach that requires, instead, a large learning period to collect observations without controlling the system.

Algorithm performance improves on average when more observations are collected. From replication to replication, the applied control differs and it might happen that some replications obtain negative energy saving (or high throughput loss) while few observations are available. As an example, Figure 6 represents the *PES* for scenario M1-1-1. A more conservative implementation phase (i.e., high  $C_0$ , low  $\alpha$ , and low  $\gamma$ ) would help in this direction. As a drawback, the control application would be delayed and energy saving on the long run reduced.

After the initial transitory, the switch-off control parameter remains  $\tau_{\text{off,impl}} = 0$  whilst the switch-on control significantly varies from replication to replication according to the estimated  $\hat{f}_X(x|\mathbf{x})$  to maintain good performance. Also, it tends to the off-line solution. As an example, Figure 7 represents the control applied for two different replications of M1-0-05-1.



**Figure 6:** PES in scenario M1-1-1 (10 replications). The first 500 observations are reported.



**Figure 7:** Implemented switch on parameter  $\tau_{\text{on,impl}}$  for scenario M1-0.05-1. The first 500 observations are reported.

### 6.1.3. Effect of the constraints

Focusing on the constrained scenarios (off-line results in Table 2 and on-line results in Table 4), the *PTL* and the *PER* satisfy the imposed targets, respectively  $\varepsilon$  and  $\delta$ . The introduction of a binding constraint decreases the amount of *PES* with respect to unconstrained scenario M1-1-1. This behavior is not surprising, since the constrained solutions are a trade-off among different conflicting objectives.

As in Figure 8, the on-line approach progressively increases the *PES* while the number of observations increases. All constrained scenarios appear more conservative despite being on average feasible. Furthermore, the effect seems to be non-linear in the constraint target value  $\varepsilon$ . As  $\varepsilon$  decreases, the *PES* decreases more significantly and the AO policy will become the optimal. Similarly for  $\delta$ .

### 6.1.4. A note on computational effort

Computational times per iteration are collected in Table 5. The learning phase, which does not vary among scenarios, requires more time as the number of collected observations increases. However, it is still performed in a reasonable time considering the application field. The optimization phase depends both on the amount of collected observations and on the typology and setting of problem constraints and constitutes most of the computational effort. The time required for the implementation phase is not reported because it is not significantly affected by the number of observations, it is almost constant among scenarios, and it can be considered negligible (i.e.,  $<1$  s).

It is noteworthy that the total computational time for an iteration is quite limited in all tested scenarios. This indicates that the proposed algorithm features fast response time laying the foundations for a real-time applicability.

## 6.2. Control adjustment to production dynamics

The goal of these [computer simulation](#) experiments is to assess the benefits of detecting changes in the upstream production process (i.e., detection phase of the algorithm), while controlling the machine for energy saving purposes.

We consider a real CNC (Computer Numerical Control) machining center (M2), characterized by the use of a power meter that elaborates the three-phase voltages and the linked current, measured through LEM sensors, as in Table 3. The machine is a COMAU SmartDrive700L machining center equipped with 5 linear axes, a working cube of 700mmx700mmx800mm, a HSK63 horizontal spindle (31.5kW) and 110kW of installed power. The machine commonly executes machining operations (milling, drilling, finishing) for powertrain applications. According to the information provided by the company, the processing time is assumed deterministic and equal to  $t_p = 168$  s and the maximum admissible throughput reduction is  $\varepsilon = 0.02$ . Also, we relax the energy risk constraint (i.e., scenario M2-0.02-1).

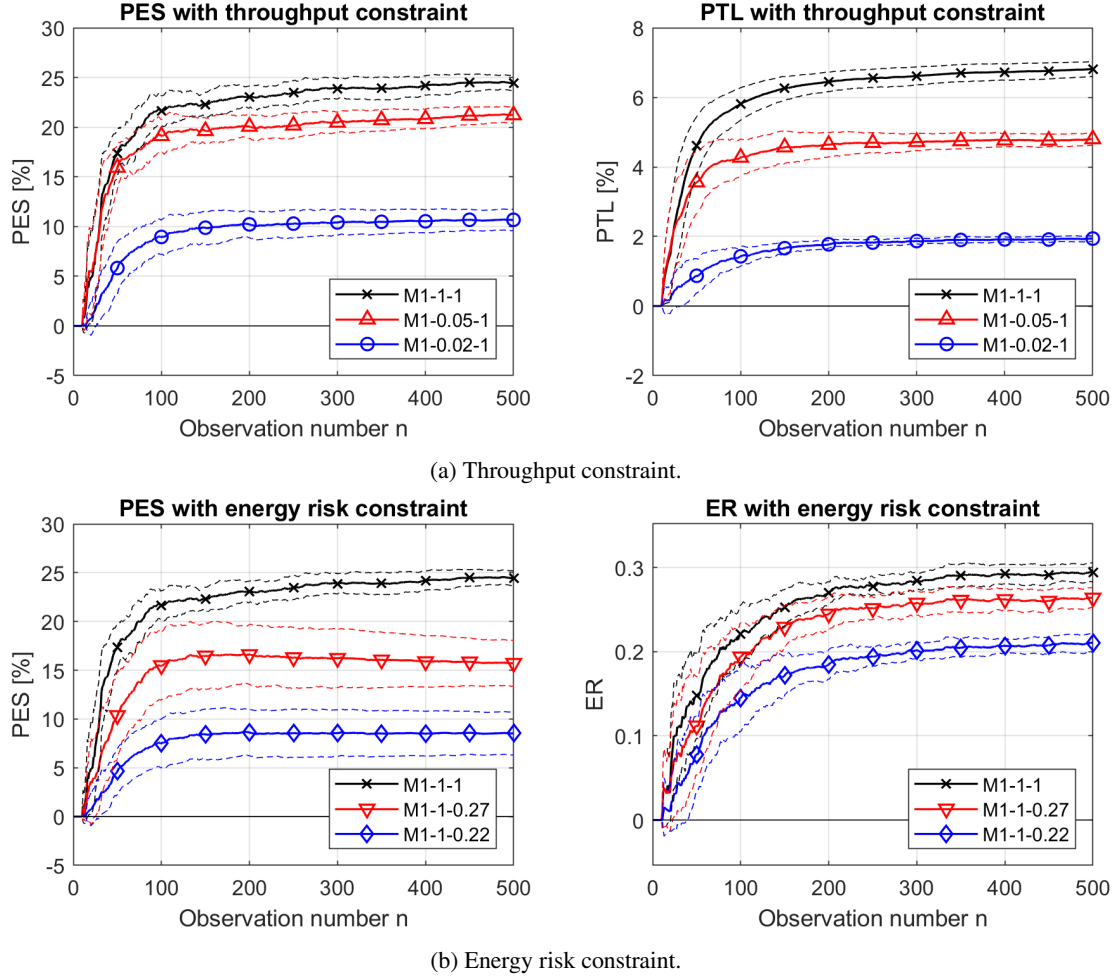
A production run of 600 parts is considered and we assume that the pdf  $f_X(x)$  changes at observations  $i = 201$  and  $j = 401$ . A Weibull distribution  $f_1(x)$  with shape  $k = 5$  and scale  $\beta = 49.011$  (i.e.,  $t_{a,1} = 45$  s) is employed when  $n \in [1, 200]$  and  $n \in [401, 600]$ . The following pdf  $f_2(x)$  with mean  $t_{a,2} = 46$  s is instead employed when  $n \in [201, 400]$ :

$$f_2(x) = \mathcal{G}(0, 7.5) + 0.1\mathcal{G}(70, 5) + 0.4\mathcal{G}(90, 10); \forall x \geq 0$$

where  $\mathcal{G}(m, v)$  is a Gaussian distribution with mean  $m$  and standard deviation  $v$  conveniently truncated to have  $x \geq 0$ . Function  $f_2(x)$  is in Figure 9 and it is characterized by a mode in  $x = 0$  followed by a second bi-modal peak. Despite the differences in shape and variance, distribution  $f_1(x)$  and  $f_2(x)$  have a similar mean; hence, the change point is very difficult to handle.

Assuming that  $f_1(x)$  and  $f_2(x)$  are known, the AO policy obtains respectively  $g_{\text{AO},1} = 240.75$  kJ/part and  $g_{\text{AO},2} = 246.06$  kJ/part. In addition, the achievable throughput is respectively  $\theta_{\text{AO},1} = 16.90$  parts/h and  $\theta_{\text{AO},2} = 16.82$  parts/h. The off-line





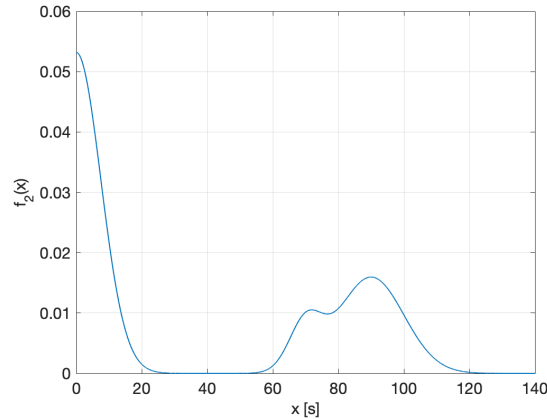
**Figure 8:** Sample-based comparison among scenarios  $M1-\varepsilon-\delta$ . The first 500 observations are reported. Full line represents the mean and dotted lines represent the respective 95% CI (10 replications).

Phase	Scenario	Observation number				
		$n = 50$	$n = 250$	$n = 500$	$n = 1000$	$n = 2500$
Learning	$M1-\varepsilon-\delta$	$0.04 \pm 0.03$	$0.18 \pm 0.05$	$0.41 \pm 0.12$	$1.20 \pm 0.30$	$5.19 \pm 1.78$
	$M1-1-1$	$1.51 \pm 0.18$	$5.51 \pm 0.82$	$7.16 \pm 0.52$	$8.23 \pm 0.61$	$11.35 \pm 0.84$
Optimization	$M1-0.05-1$	$2.67 \pm 0.14$	$8.09 \pm 0.80$	$10.03 \pm 0.43$	$12.03 \pm 0.55$	$16.05 \pm 2.11$
	$M1-0.02-1$	$3.08 \pm 0.14$	$8.21 \pm 1.01$	$10.05 \pm 0.65$	$11.60 \pm 0.40$	$15.67 \pm 1.32$
	$M1-1-0.27$	$2.05 \pm 0.59$	$9.29 \pm 1.86$	$12.91 \pm 1.33$	$13.80 \pm 1.02$	$18.99 \pm 1.06$
	$M1-1-0.22$	$2.87 \pm 0.23$	$9.83 \pm 1.67$	$12.87 \pm 1.40$	$14.00 \pm 1.19$	$18.91 \pm 0.96$

**Table 5:** Computational times [s] per iteration according to the tested scenario: learning and optimization phases (95%CI of 10 replications).

constrained policy achieves 24.36% and 36.47% of savings on the objective function with the theoretical optimal controls  $\tau_1^* = \{0, 21.55\}$  s and  $\tau_2^* = \{13.04, 65.48\}$  s.

In the following, we compare the results obtained with (namely "On-line+CPD" algorithm) and without (namely "On-line" algorithm) the detection phase. Common random numbers are used to sample starvation times in simulation experiments. The considered implementation phase setting is:  $C_0 = 10$  kJ/part,  $\alpha = 0.05$  and  $\gamma = 0.006$ .



**Figure 9:** Probability density function  $f_2(x)$ .

Actual change point	Located mean	Located range
$i = 201$	197.20	[190,201]
$j = 401$	396.70	[390,403]

**Table 6:** Location of detected change points (10 replications).

### 6.2.1. Accuracy in change point detection

In order to provide an accurate and responsive performance, the RuLSIF method is employed with the following parameter combination:

$$\{n_d, k_d, \alpha_d, \eta_d\} = \{30, 5, 0.01, 25\}. \quad (17)$$

The parameters have been chosen after a calibration phase.

With reference to Section 5, the considered setting entails a delay of  $n_d + k_d - 2 = 33$  observations in change point detection. In other words, the change point score of observation  $i$  can be computed only when  $n \geq i + 33$ . The detection phase returns the change point locations in Table 6, which vary according to the considered sample path. In some cases, the identified change point precedes the actual one.

### 6.2.2. CPD effect on on-line results

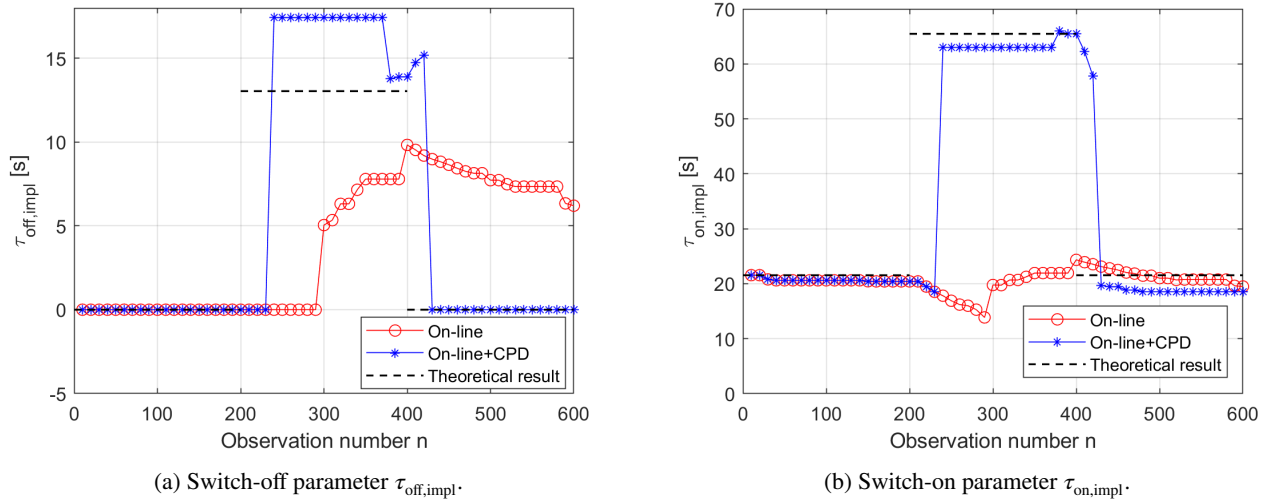
The behavior of the algorithm enriched with CPD can be appreciated from the control applied in one replication, as reported in Figures 10a and 10b. Initially, the two algorithms ("On-line" and "On-line+CPD") behave identically and control parameters are close to the theoretical ones. When the first change in  $f_X(x)$  occurs at  $n = 201$ , the estimation of  $f_2(x)$  becomes biased due to the observations from  $f_1(x)$  and the algorithms start decreasing  $\tau_{\text{on,impl}}$ . At  $n = 240$  the "On-line+CPD" algorithm detects a change point in  $i = 201$  and discards previous observations  $x_i$  with  $i \in [1, 200]$ . The learning phase now focuses on a homogeneous subset and control parameters are closer to the theoretical ones. Similarly for the second change point identified at  $j = 395$  when  $n = 430$ . The "On-line" algorithm, instead, becomes strongly biased, since the algorithm learns from data belonging to two different populations.

The  $PES$  and  $PTL$  obtained by algorithms "On-line" and "On-line+CPD" are in Figures 11a and 11b. Both algorithms improve the AO scenario but the CPD significantly enhances the performance.

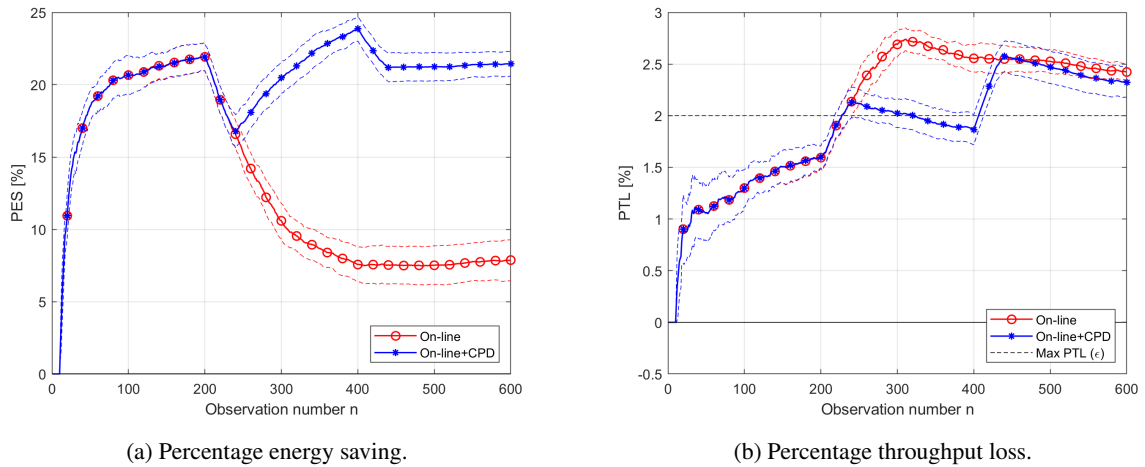
Initially, the two algorithms behave identically; whilst, when the first change in  $f_X(x)$  occurs at  $n = 201$ , the performance of the "On-line" algorithm drops and  $PES$  decreases (Figure 11a). The detection phase allows to reverse the initially declining trend of  $PES$  for the "On-line+CPD" algorithm, and maintains a good performance despite the variation in the starvation process. Similarly for the second change point. Over the observation period, this entails a  $PES$  which is, on average, double with respect to that obtained without the detection phase.

In Figure 11b, the performance detriment of "On-line" algorithm is also visible. After an initial peak of  $PTL$ , the "On-line" algorithm reacts and adjust the control parameters to maintain a feasible solution. This reaction is faster for the "On-line+CPD" algorithm since the detection phase contributes to the control adaptation.

The computational effort required for the detection phase is negligible (i.e.,  $< 0.05$  s) and it is not affected by observed data  $n$ : observations are indeed processed within a sliding window of fixed size  $2n_d + k_d - 1$ . In terms of total computational effort of the algorithm, when a change point is detected, a certain portion of observations is discarded and the learning and optimization phases become faster for next iterations.



**Figure 10:** Implemented control parameters for on-line algorithm with vs without the detection phase. Scenario M2-0.02-1, 1 replication. Actual change points at observations  $n = 201$  and  $n = 401$ .



**Figure 11:** Comparison among on-line algorithm with ("On-line+CPD") and without ("On-line") detection phase (scenario M2-0.02-1, 10 replications). Dotted lines represent the 95%CI among the mean.

### 6.3. A realistic production case

The goal of this section is to assess algorithm performance at higher level, showing the robustness of our approach in more complex environments of application. We consider a two-stage manufacturing system decoupled by an infinite intermediate buffer of parts, where the second stage is controlled for energy saving purposes. The first stage is composed of five parallel and equal machines followed by a single-server stage.

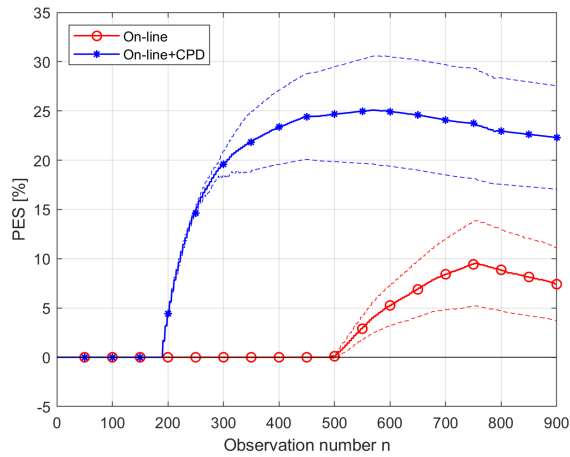
All machines are of type M3 as in Table 3 and parameters have been obtained after ad-hoc measurements. Random processing times are used to model microstoppages at machines: Uniform(415, 435) s is used for machines at the first stage and Uniform(77.5, 82.5) s for the second stage.

A production run of 900 parts is simulated, assuming that from observation  $i = 151$  to  $j = 751$  one of the parallel machines is not producing due to a severe failure requiring a long maintenance intervention. Expected system throughput is 42.35 parts/h, reduced to 33.88 parts/h when only four machines are working in parallel. The temporary reduction of 20% in the throughput leads to an energy saving potential at the second stage. Note that the first stage is the bottleneck of the system and, for this reason, machines are not controlled. The machine at the second stage has a utilization of 95%, which theoretically decreases to around 75% when one of the upstream machines fails.

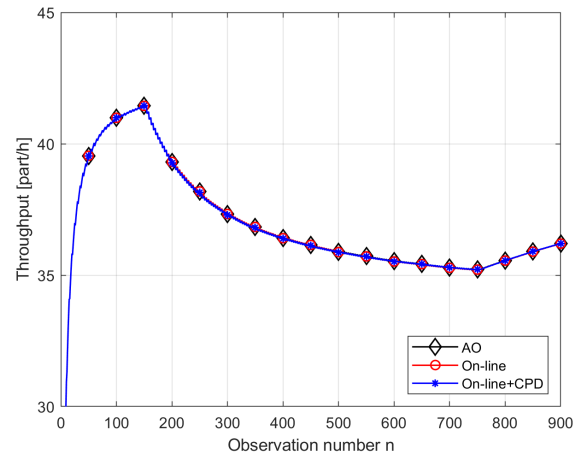
The AO policy consumes on average  $26.68 \pm 0.37$  kJ/part when five parallel machines are working and  $112.03 \pm 0.30$  kJ/part when only four work (over the whole observation period:  $83.58 \pm 0.28$  kJ/part). This consumption is related to the starvation of the second-stage machine, because the first stage is never starved of raw parts and never blocked. The RuLSIF method

uses parameters as in equation (17).

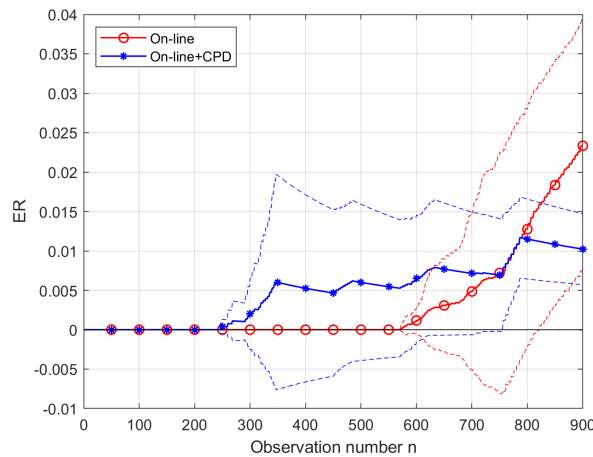
Results in terms of  $PES$ , system throughput and  $ER$  are in Figure 12. Initially, the "On-line" algorithm does not control the machine keeping an AO policy. Then, after  $n = 500$ , it controls the machine with  $\tau_{\text{off}} = 0$  s and  $\tau_{\text{on}} \approx 126$  s. Change points are detected along the observation period and the "On-line+CPD" algorithm starts sooner in controlling the machine. This can be noticed in Figure 12a, where the savings obtained are significantly higher than those of the "On-line" algorithm. As an example, the control applied to one replication is reported in Figure 13. Three change points are detected: the first ( $n = 185$ ) identifies the failure at first stage, the second ( $n = 751$ ) the repair, and the third ( $n = 793$ ) is actually a false alarm. Initially, the AO policy is implemented ( $\tau = \{\infty, \infty\}$ ), then the machine is controlled because the starvation mean increases, and at last the AO policy is implemented.



(a) Percentage energy saving.

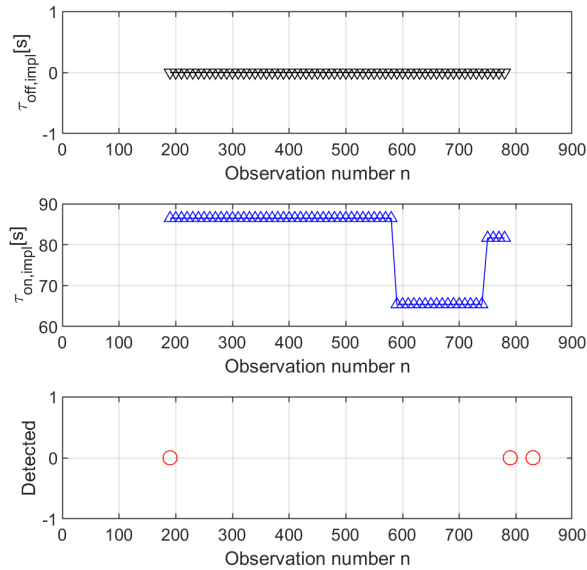


(b) System throughput.



(c) Energy risk.

**Figure 12:** Sample-based comparison of results obtained for M3 in the two-stage scenario. Full line represents the mean and dotted lines represent the respective 95% CI (10 replications).



**Figure 13:** "On-line+CPD" control for M3 in the two-stage scenario. Example of one replication.

As in Figure 12b, system throughput decreases after observation  $i = 151$  because of the failures at the first stage and recovers after observation  $j = 751$  when the service is resumed. This behavior is not affected by the control because the intermediate buffer absorbs the variation in queue length. Indeed, the average buffer occupancy is 0.80 parts (up to a maximum of 4 parts) in the AO case, and 0.91 parts (up to a maximum of 4 parts) in the "On-line+CPD" case. In addition, the assumption of infinite capacity is realistic and it is not affecting the blocking probability at the first stage. Thus, results of the throughput constrained problem are equal to those of the unconstrained problem.

The  $ER$  is quite variable because of the difficulties in estimating  $f_X(x)$ , but the maximum  $ER$  obtained in a replication is 6% which represents a very low energy risk. Nevertheless, as in Figure 12c, the CPD maintains a lower  $ER$  given the better estimation of  $f_X(x)$ .

## 7. Conclusions and future developments

The objective of this work is to move a step forward in the energy efficient control of machine states under uncertainty in part arrival. Compared to existing studies, the proposed approach is highly flexible and can be applied on-line without an intensive effort for practitioners. Considering the widespread diffusion of machining processes in industry, our study can be of high impact for the sector. Specifically:

- In the analyzed [simulated](#) scenarios, the proposed approach reduces the energy consumption up to 25% with respect to the AO policy.
- The proposed algorithm solves the on-line problem of minimizing energy while assuring constraints on machine performance so as a target throughput.
- The proposed algorithm learns from acquired data and autonomously selects the control to be applied.
- When the arrival process is non-stationary in time, the proposed algorithm reacts effectively limiting the deterioration of performance.
- Low computational times ensure shop floor applicability.

As future development, the proposed algorithm can be generalized to include both constraints simultaneously. However, this might result in a severe increase of computational time which might be not suitable for practical application where, in addition, those constraints are rarely both binding.

Future effort will be devoted to the application of change point detection methods to other performance of interest. This may allow to identify variations in the system or in the external environment in a more effective way. Steps towards system level application will be pursued. Indeed, when model assumptions are not completely fulfilled, the algorithm might

be not efficient; for example, starvation times are hardly iid and can be affected by the applied control policy. Also, the detection method might fail. New methods should be investigated to better estimate and react to variations in the production environment.

## Appendix A. Machine model

The model has been proposed in the literature (Frigerio and Matta (2015)). *Machine energy demand*  $\varphi(x, \tau)$  follows:

$$\varphi(x, \tau) = \begin{cases} w_{id} \cdot x & \text{if } A_1 \\ w_{id} \cdot \tau_{off} + w_{sb}(x - \tau_{off}) + w_{su} \cdot t_{su} & \text{if } A_2 \\ w_{id} \cdot \tau_{off} + w_{sb}(\tau_{on} - \tau_{off}) + w_{su} \cdot t_{su} & \text{if } A_3 \\ w_{id} \cdot \tau_{off} + w_{sb}(\tau_{on} - \tau_{off}) + w_{su} \cdot t_{su} & \text{if } A_4 \\ + w_{id}(x - \tau_{on} - t_{su}) & \end{cases}$$

where  $A_i | i = 1, 2, 3, 4$  are the following conditions:

$$\begin{aligned} A_1 &= \{0 \leq x \leq \tau_{off}\} \\ A_2 &= \{\tau_{off} < x \leq \tau_{on}\} \\ A_3 &= \{\tau_{on} < x \leq \tau_{on} + t_{su}\} \\ A_4 &= \{x > \tau_{on} + t_{su}\}. \end{aligned}$$

Conditions  $A_i | i = 1, 2, 3, 4$  depend on the interaction between  $x$  and  $\tau$  and their probability of occurrence depends on  $f_X(x)$ . Similarly, we obtain the *part waiting time*  $\psi(x, \tau)$ :

$$\psi(x, \tau) = \begin{cases} 0 & \text{if } A_1 \\ t_{su} & \text{if } A_2 \\ \tau_{on} + t_{su} - x & \text{if } A_3 \\ 0 & \text{if } A_4. \end{cases}$$

Therefore, the *energy consumed in a cycle*  $\phi(x, \tau)$  is obtained by equation (1).

## Appendix B. Learning phase

Given a set of  $n$  observed arrivals  $\mathbf{x} = \{x_i | i = 1, \dots, n\}$ , the estimated distribution is:

$$\hat{f}_X(x|\mathbf{x}) = \frac{1}{nh} \sum_{i=1}^n \mathcal{K}\left(\frac{x - x_i}{h}\right)$$

where  $\mathcal{K}(\cdot)$  is the Gaussian Kernel function and  $h > 0$  is a smoothing parameter, referred to as bandwidth (Parzen (1962)). The implemented KDE method finds the optimal bandwidth  $h_n^*$  with the leave-one-out cross validation method (Bowman (1984)). Therefore, we use  $h = h_n^*$  that makes most likely the  $n$  data observed by maximizing the Likelihood function:

$$h_n^* = \arg \max_{h \in \mathbb{R}^+} \left\{ \prod_{i=1}^n \frac{1}{h \cdot (n-1)} \left[ \sum_{\substack{j=1 \\ j \neq i}}^n \mathcal{K}\left(\frac{x_i - x_j}{h}\right) \right] \right\}.$$

## Appendix C. Solving algorithm for the unconstrained problem

The unconstrained off-line optimization problem, obtained by relaxing constraints (3) and (4), has been addressed in the literature (Frigerio and Matta (2015)). In details, control parameters are independent and problem solution can be found. The algorithm proposed in the literature is herewith improved to address general  $\hat{f}_X(x|\mathbf{x})$  estimated after  $n$  observations:

**Step 1** Create set  $\mathcal{T}_{off}$  collecting the solutions of the following equation:

$$\frac{\partial \hat{g}(\tau)}{\partial \tau_{off}} = 0$$

and include also  $\tau_{off} = 0$  to account for potential solutions on the domain boundary.



**Step 2** Create set  $\mathcal{T}_{\text{on}}$  collecting the solutions of the following equation:

$$\frac{\partial \hat{g}(\boldsymbol{\tau})}{\partial \tau_{\text{on}}} = 0.$$

**Step 3** Create set  $D_{\text{unc}}$  of candidate problem solutions combining elements of sets  $\mathcal{T}_{\text{off}}$  and  $\mathcal{T}_{\text{on}}$  such that each pair  $\{\tau_{\text{off},i}, \tau_{\text{on},j}\}$   $\forall \tau_{\text{off},i} \in \mathcal{T}_{\text{off}}, \tau_{\text{on},j} \in \mathcal{T}_{\text{on}}$  belongs to the domain of decision variables  $\mathcal{U}$ , see equation (10). Note that candidate solution  $\{\infty, \infty\}$  represents the AO policy and must be included in  $D_{\text{unc}}$ .

**Step 4** Select:

$$\tau_n = \min_{\boldsymbol{\tau} \in D_{\text{unc}}} \hat{g}(\boldsymbol{\tau}).$$

## References

- Aminikhanghahi, S., Cook, D., 2016. A survey of methods for time series change point detection. *Knowledge and Information Systems* 51, pp. 339–367. doi:10.1007/s10115-016-0987-z.
- Bowman, A., 1984. An alternative method of cross-validation for the smoothing of density estimates. *Biometrika* 71, pp. 353–360. doi:10.2307/2336252.
- Chan, Y., Kantamaneni, R., Allington, M., 2015. Study on energy efficiency and energy saving potential in industry and on possible policy mechanisms. ICF Consulting Limited, London. 6, 2016.
- Chen, G., Zhang, L., Arinez, J., Biller, S., 2011. Feedback control of machine startup for energy-efficient manufacturing in bernoulli serial lines, in: *IEEE International Conference on Automation Science and Engineering*, pp. 666–671. doi:10.1109/CASE.2011.6042495.
- Dahmus, J.B., Gutowski, T.G., 2004. An environmental analysis of machining, in: *ASME international mechanical engineering congress and exposition*, pp. 643–652. doi:10.1115/IMECE2004-62600.
- Devoldere, T., Dewulf, W., Deprez, W., Willems, B., Duflou, J.R., 2007. Improvement potential for energy consumption in discrete part production machines, in: *Advances in life cycle engineering for sustainable manufacturing businesses*. Springer, pp. 311–316. doi:10.1007/978-1-84628-935-4\_54.
- Duflou, J.R., Sutherland, J.W., Dornfeld, D., Herrmann, C., Jeswiet, J., Kara, S., Hauschild, M., Kellens, K., 2012. Towards energy and resource efficient manufacturing: A processes and systems approach. *CIRP annals* 61, 587–609. doi:10.1016/j.cirp.2012.05.002.
- Duque, E.T., Fei, Z.C., Wang, J., Li, S.Q., Li, Y.F., 2019. Energy consumption control of one machine manufacturing system with stochastic arrivals based on fuzzy logic, in: *IEEE International Conference on Industrial Engineering and Engineering Management*, pp. 1503–1507. doi:10.1109/IEEM.2018.8607749.
- Efron, B., 1987. Better bootstrap confidence intervals. *Journal of the American Statistical Association* 82, pp. 171–185. doi:10.2307/2289144.
- Energy Information Administration, U., 2019. International energy outlook 2019, with projections to 2050. Available at: <https://www.eia.gov/ieo> (access Sept 2020).
- Frigerio, N., Matta, A., 2015. Energy-efficient control strategies for machine tools with stochastic arrivals. *IEEE Transactions on Automation Science and Engineering* 12, pp. 50–61. doi:10.1109/TASE.2014.2344507.
- Frigerio, N., 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, pp. 238–246. doi:10.1109/TASE.2015.2492600.
- Gahm, C., Denz, F., Dirr, M., Tuma, A., 2016. Energy-efficient scheduling in manufacturing companies: A review and research framework. *European Journal of Operational Research* 248, 744–757. doi:10.1016/j.ejor.2015.07.017.
- Guo, X., Zhou, S., Niu, Z., Kumar, P.R., 2013. Optimal wake-up mechanism for single base station with sleep mode, in: *Proceedings of the 2013 25th International Teletraffic Congress, ITC 2013*. doi:10.1109/ITC.2013.6662947.
- Jia, Z., Zhang, L., Arinez, J., 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, pp. 6285–6301. doi:10.1080/00207543.2016.1197438.
- Law, A., 2015. *Simulation modeling and analysis*. McGraw Hill.
- Lee, S., Do Chung, B., Jeon, H.W., Chang, J., 2017. A dynamic control approach for energy-efficient production scheduling on a single machine under time-varying electricity pricing. *Journal of Cleaner Production* 165, 552–563. doi:10.1016/j.jclepro.2017.07.102.
- Liu, S., Yamada, M., Collier, N., Sugiyama, M., 2013. Change-point detection in time-series data by relative density-ratio estimation. *Neural Networks* 43, pp. 72–83. doi:https://doi.org/10.1016/j.neunet.2013.01.012.
- Marzano, L., Frigerio, N., Matta, A., under review. An on-line policy for energy-efficient state control of manufacturing equipment. *IEEE Trans. on Automation Science and Engineering* (second submission).
- Materi, S., D'Angola, A., Renna, P., 2020. A dynamic decision model for energy-efficient scheduling of manufacturing system with renewable energy supply. *Journal of Cleaner Production*, 122028doi:10.1016/j.jclepro.2020.122028.
- Mouzon, G., Yildirim, M.B., Twomey, J., 2007. Operational methods for minimization of energy consumption of manufacturing equipment. *International Journal of Production Research* 45, pp. 4247–4271. doi:10.1080/00207540701450013.
- Parzen, E., 1962. On estimation of a probability density function and mode. *Ann. Math. Statist.* 33, pp. 1065–1076. doi:10.1214/aoms/1177704472.
- Sihag, N., Sangwan, K.S., 2020. A systematic literature review on machine tool energy consumption. *Journal of Cleaner Production*, 123125doi:10.1016/j.jclepro.2020.123125.
- Squeo, M., Frigerio, N., Matta, A., 2019. Multiple sleeping states for energy saving in cnc machining centers, in: *Procedia CIRP*, pp. 144–149. doi:10.1016/j.procir.2018.12.020.
- Wang, J., Fei, Z., Chang, Q., Li, S., 2019. Energy saving operation of manufacturing system based on dynamic adaptive fuzzy reasoning petri net. *Energies* 12, pp. 1–17. doi:10.3390/en12112216.
- Wang, J., Li, S., Liu, J., 2013. A multi-granularity model for energy consumption simulation and control of discrete manufacturing system, in: *19th International Conference on Industrial Engineering and Engineering Management: Assistive Technology of Industrial Engineering*, pp. 1055–1064. doi:10.1007/978-3-642-38391-5\_112.
- Yoon, H.S., Kim, E.S., Kim, M.S., Lee, J.Y., Lee, G.B., Ahn, S.H., 2015. Towards greener machine tools—a review on energy saving strategies and technologies. *Renewable and Sustainable Energy Reviews* 48, 870–891. doi:10.1016/j.rser.2015.03.100.

Zhou, L., Li, J., Li, F., Meng, Q., Li, J., Xu, X., 2016. Energy consumption model and energy efficiency of machine tools: a comprehensive literature review. *Journal of Cleaner Production* 112, 3721–3734. doi:10.1016/j.jclepro.2015.05.093.



Nicla Frigerio 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. She collaborated with Purdue University, Indiana, USA, and with Shanghai Jiao Tong University, Shanghai, P.R. China. In 2018, after being a post-doc research fellow, she became Assistant Professor (RTDa) at the Manufacturing and Production Systems Research Line of the Department of Mechanical Engineering at the Politecnico di Milano. She is a member of AITeM – Italian Association of Manufacturing Technologies and of IEEE - Institute of Electrical and Electronics Engineers.



Claudio FA Cornaggia received the M.Sc. degree in Mechanical Engineering from Politecnico di Milano, Milan, Italy, in 2020. 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 in internship in a consulting company.



Andrea Matta is Full Professor of Manufacturing and Production Systems at Department of Mechanical Engineering of Politecnico di Milano. 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 from 2014 to 2016. He has been visiting professor at Ecole Centrale Paris (France), University of California at Berkeley (USA), and Tongji University (China). 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 with a focus on quantitative methods. Since 2017 he is Editor-in-Chief of Flexible Services and Manufacturing Journal edited by Springer Nature.