Mechanical Systems and Signal Processing 124 (2019) 21–35

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

journal homepage: www.elsevier.com/locate/ymssp

An automated feature extraction method with application to empirical model development from machining power data

Dimitrios Pantazis *, Paul Goodall, Paul P. Conway, Andrew A. West

Wolfson School of Mechanical, Electrical and Manufacturing Engineering, Loughborough University, Loughborough LE11 3TU, UK

ARTICLE INFO

Article history:
Received 7 July 2018
Received in revised form 7 December 2018
Accepted 11 January 2019

Keywords:
Feature extraction
Machining
Hidden Markov model (HMM)
Response Surface Methodology (RSM)
Power data
Time-series segmentation
Energy decomposition
Hierarchical clustering
End-milling
Empirical model

ABSTRACT

Machining shop floor jobs are rarely optimised for minimisation of the energy consumption, as no clear guidelines exist in operating procedures and high production rates and finishing quality are requirements with higher priorities. However, there has been an increased interest recently in more energy-efficient process designs, due to new regulations and increases in energy charges. Response Surface Methodology (RSM) is a popular procedure using empirical models for optimising the energy consumption in cutting operations, but successful deployment requires good understanding of the methods employed and certain steps are time-consuming. In this work, a novel method that automates the feature extraction when applying RSM is presented. Central to the approach is a continuous Hidden Markov model, where the probability distribution of the observations at each state is represented by a mixture of Gaussian distributions. When applied to a case study, the automated extracted material cutting energies lay within 1.12% of measured values and the spindle acceleration energies within 3.33% of their actual values.

1. Introduction

The industrial sector accounts for more than 36% of the energy consumption of electricity in the European Union [17]. The European Union has set targets for improving energy efficiency by at least 20% before 2020 [18]. Manufacturing is one of the biggest industrial energy consumers and machining processes in particular can greatly benefit from the optimisation of cutting parameters and the tool path, potentially resulting in up to 40% savings of the total energy consumption [34]. As a result, there has been an increasing interest in the development of accurate statistical models that can be used for the estimation and optimisation of Key Energy Indicators (KEI), such as total energy consumption per operation, specific energy consumption, peak power demand and cycle duration, which can contribute to significant energy savings.

Lately, empirical models have been preferred over traditional optimisation techniques that are based on manufacturer recommendations and operator's knowledge that often do not account for energy consumption [48,12,4]. Theoretical models relying purely on mathematical equations and not on real data cannot easily account for the details and particularities of each machine or machining process, such as calibration adjustments, part wear inside the machine or defects that are difficult to integrate into formulas. On the other hand, empirical models can easily be adapted and applied to different machines without substantial effort and with minimal knowledge of the machine internals.

* Corresponding author.

E-mail addresses: D.Pantazis@lboro.ac.uk (D. Pantazis), P.A.Goodall@lboro.ac.uk (P. Goodall), P.P.Conway@lboro.ac.uk (P.P. Conway), A.A.West@lboro.ac.uk (A.A. West).

https://doi.org/10.1016/j.ymssp.2019.01.023
0888-3270/© 2019 The Authors. Published by Elsevier Ltd.

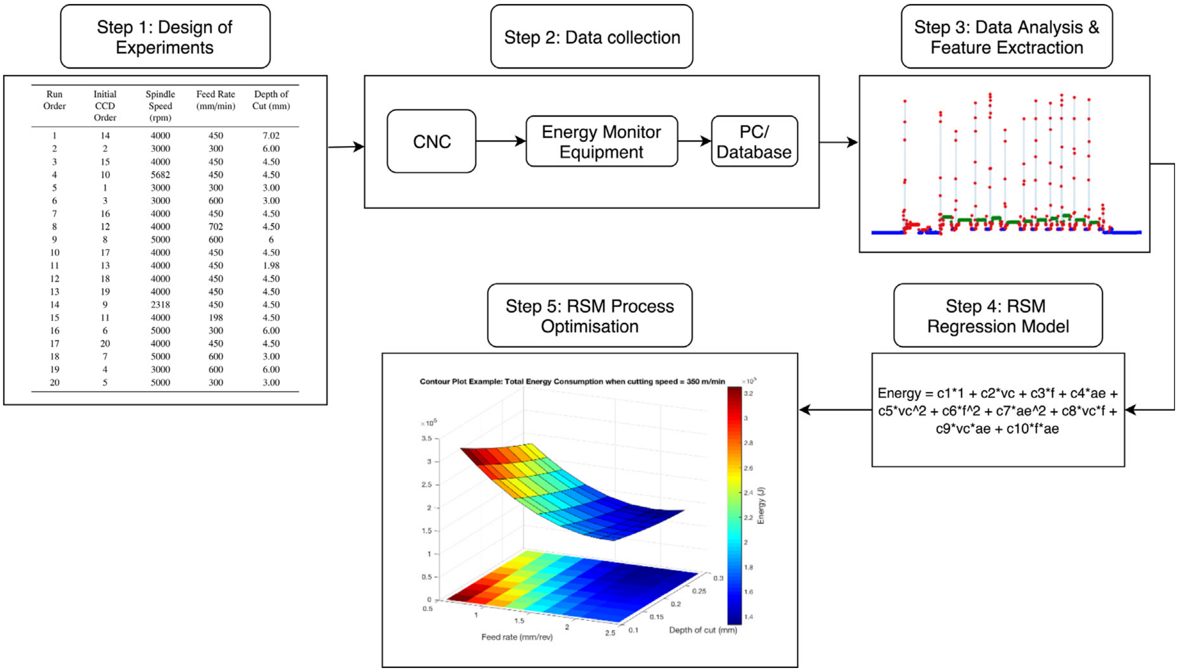
A variety of methods for the development of empirical models for CNC cutting operations have been suggested in the literature. Mukherjee and Ray [41] provide an extensive review of classification of modelling and optimisation techniques in metal cutting process problems, which have remained essentially unchanged since the paper was published. Taguchi's approach [45,47] and the Response Surface Methodology (RSM) [10] are usually the preferred methods due to their simplicity, flexibility, support in statistics packages and high reduction of the number of experimental runs, which nevertheless produce highly accurate results. Whilst the Taguchi method can offer a quick way of optimisation and describes the main factors' effects in the processes effectively, the RSM can identify the significance of squared terms and interactions between the cutting parameters more precisely [7,2]. Furthermore, the 3D surfaces of the RSM help in communicating the results to the user [44]. For this reason, the focus of this research is on RSM studies using Central Composite Design of experiments (CCD), but the proposed methodology can be used with orthogonal array designs or other types of design of experiments as well.

The RSM comprises mathematical and statistical techniques that aim at building a model that maps changes in a set of independent variables (input variables) to the response (output variables) [39]. RSM has been widely used for optimisation of the machining parameters in CNC processes, with regards to energy consumption minimisation as the primary aim and cutting quality and material removal rate (MRR) maximisation as secondaries. The authors Campatelli et al. [11] evaluate the effect of cutting speed, the axial and radial depth of cut, and the feed rate on the power consumption during the CNC milling process of a carbon steel block. Camposeco-Negrete [12] follows a similar RSM procedure but for a turning process and reports the feed rate and depth of cut as the most significant factors for optimising the energy consumption, while managing to achieve a 14.41% improvement compared to traditional optimisation methods. Bhushan [9] attempts to minimise power consumption while maximising tool life using desirability analysis with the RSM. Abhang and Hameedullah [1] built a predictive model for steel turning using RSM, proved that the second order model is more precise than the first order and concluded that the smaller the values of the studied variables (i.e. cutting speed, feed rate, depth of cut and tool nose radius), the lower the resulting power consumption.

The usual model development procedure for the RSM can be seen in Fig. 1 and consists of 5 steps:

1. Design of Experiments (DoE)
2. Data Collection
3. Data Analysis and Feature Extraction
4. Linear Regression Modelling
5. Process Optimisation

Steps (1), (4) and (5) are well understood and have been computer-aided to provide a quick and reliable solution to the optimisation problem. Both in academia and industry a variety of statistical software packages have been used to facilitate


Step 1: Design of Experiments
Step 2: Data collection
CNC
Energy Monitor Equipment
PC/ Database
Step 3: Data Analysis & Feature Exctraction
Step 4: RSM Regression Model
Energy = c1*1 + c2*vc + c3*f + c4*ae + c5*vc^2 + c6*f^2 + c7*ae^2 + c8*vc*f + c9*vc*ae + c10*f*ae
Step 5: RSM Process Optimisation
Contour Plot Example: Total Energy Consumption when cutting speed = 350 m/min
×10^5
3.5
3
2.5
2
1.5
1
0.5
0
0.5
1
1.5
2
2.5
Feed rate (mm/rev)
0.1
0.15
0.2
0.25
0.3
Depth of cut (mm)
×10^5
3.2
3
2.8
2.6
2.4
2.2
2
1.8
1.6
1.4
Energy (J)

Run Order	Initial CCD Order	Spindle Speed (rpm)	Feed Rate (mm/min)	Depth of Cut (mm)
1	14	4000	450	7.02
2	2	3000	300	6.00
3	15	4000	450	4.50
4	10	5682	450	4.50
5	1	3000	300	3.00
6	3	3000	600	3.00
7	16	4000	450	4.50
8	12	4000	702	4.50
9	8	5000	600	6
10	17	4000	450	4.50
11	13	4000	450	1.98
12	18	4000	450	4.50
13	19	4000	450	4.50
14	9	2318	450	4.50
15	11	4000	198	4.50
16	6	5000	300	6.00
17	20	4000	450	4.50
18	7	5000	600	3.00
19	4	3000	600	6.00
20	5	5000	300	3.00

Fig. 1. Response Surface Methodology.

this, such as Stat-Ease's Design Expert® [2] and Minitab [5]. The data collection and feature extraction steps however are still heavily dependent on the user's skills and experience, which not only adds subjectivity to the accuracy of the measurements, but also are very time-consuming and hence costly. While the data collection precision relies on the monitoring hardware equipment, the feature extraction takes place exclusively at a software level. In an industrial environment where strict requirements exist, users can benefit from the automation of the latter, which can also offer a degree of standardisation, reducing human error.

In this paper a novel approach to automating step (3) is presented so that the extraction of the desired features from the energy data is performed without user intervention. To do so, the energy signature must be split into a number of sequences representing different features. In Section 2 a typical energy profile of a cutting operation is studied and how it can be decomposed in individual segments is detailed. In Section 3 a short introduction to segmentation techniques of time-series data encountered in the literature is presented, with a focus on Hidden Markov models (HMMs), which are the key component within the algorithm presented in this paper. In Section 4 the methods explaining analytically each of the separate steps involved in the analysis are presented. In Section 5 the validation study is presented along with discussion on the results. In Section 6 conclusions and further work are outlined.

2. Energy consumption profile analysis for feature selection

Analysing and understanding the energy consumption profile is crucial for extracting the necessary features and defining the separate components that contribute to energy consumption. In the literature, the relative terms can appear ambiguous depending on the scope of each research. For example, Camposeco-Negrete [12] considers a single energy consumption in her paper, which comprises the required energy for the material removal including all the participating components of the machine tool. However, Campatelli et al. [11] distinguish between two energies when presenting their results: the energy E which only considers the spindle and the axis consumption and the total energy E_{tot} which includes the total consumption related to the machine.

It is thus apparent that the energy consumption decomposition is not a straightforward process and several approaches are found in literature. Yoon et al. [49], Zhou et al. [52], Edem and Mativenga [16] and Zhao et al. [51] provide comprehensive and up-to-date reviews of different ways that a consumption profile can be split into smaller parts. The principle behind how each division occurs, changes according to the purpose of each study. As a result, the common point of reference may be the active subsystem, the consuming components, the operational status of the machine, the functional movement or even more generic, by categorising whether the consumption is constant or variable [52]. Depending on the choice, the result can be a vertical split of the energy signal (along the power axis), a horizontal split (along the time axis) or a combination of both.

A summary of the most important recent studies on the breakdown of E_{total} can be found in Table 1. In most of these works, the aim is to create a model which predicts the machine's consumption or to propose an optimisation method in terms of the cutting parameters, similar to that in Fig. 1.

Kordonowy [32] is considered one of the most fundamental studies when attempting to understand and describe power consumption in machining. An important contribution of this work is the separation of the consumption into constant and variable components. It has consolidated the basis for some pioneering work, such as Dahmus and Gutowski [13], which was one of the first to highlight key points such as that the actual cutting energy is relatively small when compared to the total energy, E_{total}, consumed by the process. A further subdivision of the constant consumptions into processes associated with the particular machine components is also included in Kordonowy's [32] work.

In Diaz et al. [15] and Kong et al. [31], the authors make use of and extend further Dahmus and Gutowski's [13] and Kordonowy's [32] energy breakdown approach. They separate the run-time energy into steady-state, where the spindle and axis drives speeds do not change, and transient where the acceleration/deceleration operations are represented. E_{cut} is the energy

Table 1
Machining cycle energy consumption breakdown. The contributing energy components differ between papers, although similarities can be observed, such as E_{basic} that usually represents the standby state consumption and $E_{spindle}$ that indicates the consumption of the spindle subsystem.

Author	E_{total}
Kordonowy [32]	$E_{total} = E_{constant\_startup} + E_{constant\_runtime} + E_{variable\_runtime}$
Diaz et al. [15]/Kong et al. [31]	$E_{total} = E_{const} + E_{run-time-transient} + E_{run-time-steady} + E_{cut}$
Mori et al. [40]	$E_{total} = P_1(T_1 + T_2) + P_2T_2 + P_3T_3$
Avram and Xirouchakis [8]	$E_{total} = E_{aS} + E_{dS} + E_{run} + E_{cut} + E_{aY} + E_{dY} + E_{sY} + E_{fixed}$
He et al. [24]	$E_{total} = E_{spindle} + E_{feed} + E_{tool} + E_{cool} + E_{fix}$
Hu et al. [26]	$E_{total} = E_{startup} + E_{idle} + E_{cutting}$
Yoon et al. [50]	$E_{total} = E_{basic} + E_{stage} + E_{spindle} + E_{machining}$
Aramcharoen and Mativenga [6]/Edem and Mativenga [16]	$E_{total} = E_{basic} + E_{tool} + E_{spindle} + E_{cutting} + E_{feed} + E_{cutting\_fluid}$
Liu et al. [36]	$E_{total} = \sum_{j=1}^{Q_s} E_{sj} + \sum_{j=1}^{Q_u} E_{uj} + \sum_{j=1}^{Q_c} E_{cj}$
Li et al. [34]	$E_{total} = E_{st-p} + E_{ac-p} + E_{c-p} + E_{ct-p}$ (horizontal)
	$E_{total} = E_{basic} + E_{idle} + E_{cut} + E_{ad} + E_{aux}$ (vertical)

consumed for the material removal process and E_{const} originates from components not directly related to the machining (e.g. computer fans, unloaded motors).

Mori et al. [40] consider three separate power consumptions related to the machine. P_1 is the constant power consumption regardless of the operational state, P_2 represents the cutting power consumption generated by the spindle and servo motors and depends on the cutting conditions and P_3 is the power required to move the worktable and accelerate or decelerate the spindle. Each of these is multiplied by the respective period to estimate the total energy.

As part of their work to develop a methodology and a model for the estimation of the energy requirements to machine a part, Avram and Xirouchakis [8] assume that the total energy required during the operation of a CNC machine can be decomposed into: 1. the spindle acceleration energy E_{aS}, 2. the spindle deceleration energy E_{dS}, 3. the constant-speed energy consumed by the spindle due to mechanical losses, when there is no material removal E_{run}, 4. the energy required to remove a specific amount of material E_{cut}, 5. the consumptions E_{aY}, E_{dY} and E_{sY}, which are caused by acceleration, deceleration and steady-state operation of the feed axis Y (but can be extended to include X axis as well) and 6. the energy E_{fixed} which accounts for the consumption of the auxiliary components.

Hu et al. [24] try to build a consumption model for each component of the machine, thus following a vertical split. E_{feed}, E_{tool}, E_{cool} and E_{fix} are the energy consumptions of axis feed, tool change system, coolant pump, and the fixed energy consumption (i.e. fan motors and servo systems), respectively. $E_{spindle}$ is the energy consumption related to the spindle, which is then subdivided further into E_m, the energy for enabling the spindle transmission module and E_c, the material removal energy.

Hu et al. [26] is another example of a study heavily inspired from Kordonowy's [32] work. The authors attempt to develop an on-line monitoring system to determine energy efficiency of machine tools. They consider the different operating states of the machine to define a horizontal breakdown of the total energy consumption, into the startup state, the idle state and the cutting state. To be in alignment with the published literature, Hu et al.'s [26] description has been translated into an equation as listed in Table 1. Furthermore, they relate the operating energies to the spindle power profile, where the idle state comprises the idle power P_u and the cutting state the cutting power P_c as well as the additional load loss power consumption P_a.

Yoon et al. [50] study the energy consumption and manufacturing cost modelling for the micro-drilling of printed circuit boards. The E_{basic} energy consists of the idle energy, the axis jog energy, the machine starting up consumption and some peripheral devices' consumption. The rest of the components of the equation relate to the energies consumed by the stage movements, the spindle and the material removal process.

Aramcharoen and Mativenga [6] use an energy decomposition equation which, from a high-level perspective, is almost identical with the one used for the model development in He et al. [24]. However, the developed model is claimed to be more precise by accounting for factors such as the workpiece machinability and multiple tool changes per operation. Edem and Mativenga [16] use the same energy model as those studies, but the modelling of each component is undertaken with higher granularity and complexity, by incorporating the contribution of factors such as the axes' and workpiece's weight and the energy intensity of toolpath interpolations in their calculations.

Similarly to Hu et al.'s [26] work, Liu et al. [36] first identify three distinct consumption classes within the machining process, the start-up, the idle (air-cutting) and the material cutting and then attempt to create a prediction model for each one of them.

Li et al. [34] follow a more thorough breakdown of the different energies encountered. There is a horizontal separation, where four processing periods comprise the total energy associated with the machining operation. E_{st-p} is the stand-by period consumption, E_{ac-p} corresponds to the air-cutting period energy draw, E_{c-p} to the cutting period and E_{ct-p} to the tool-change. Regarding the vertical decomposition, different kinds of energy consumption are stacked to form the four processing periods. In this case, E_{basic} is the minimum consumption during the standby state, E_{idle} is the additional energy waste during the air-cutting which together with E_{basic} forms the E_{ac-p} (i.e. $E_{ac-p} = E_{basic} + E_{idle}$), E_{cut} is the cutting energy, E_{ad} is the additional load loss and E_{aux} represents the waste occurring from the machining auxiliary system.

In this research, a horizontal decomposition of the power signal is undertaken, in order to extract the energy segments associated with particular operational states. Based upon the previous works, the total consumption of a CNC machine E_T is defined as:

$$E_T = E_{sb} + E_{su} + E_{ac} + E_c + E_r + E_{tc} \tag{1}$$

where E_{su} is the energy which corresponds to the spindle acceleration action, E_{ac} is the energy consumed during the air-cutting period, E_c is the energy during the actual material cutting, E_r is the consumption during any rapid axis feed, E_{tc} is the tool-change consumption and E_{sb} is the minimum standby consumption when the motor is not spinning and the axis is not moving and is caused by the auxiliary subsystem.

It is important to notice here that E_{sb} is not always constant but can vary according to which subsystems are activated. For example, when a CNC program is running and the cutting fluid and the lights are activated, E_{sb} is slightly higher compared with when no program runs. This increased consumption can be observed in Fig. 2 as the consumption between the two machining cycles.

In this research, an attempt has been made to isolate the energies E_{su} and E_c, which are the main variables of interest in most of the optimisation studies. These two energies will be automatically extracted from the experimental design of Section 5, as if they were to be used in an RSM study. The reason that these two energies are of such high interest in literature

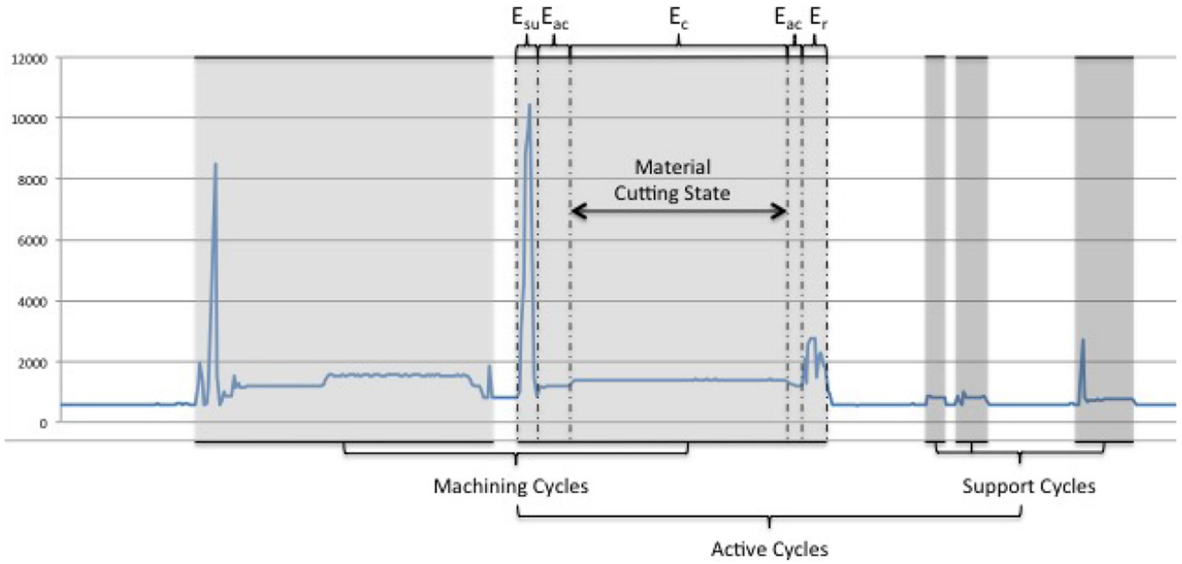


Fig. 2. Typical power consumption signal during consecutive linear end-milling cuts. E_{su} is energy consumption for the spindle acceleration phase, E_{ac} is the energy consumption during the air-cutting, E_c corresponds to the energy consumption for the material cutting period and E_r is the energy consumption during rapid axis feed operations.

is that they can be modelled as a direct function of the cutting parameters, as opposed to the rest of the factors of Eq. (1), which might be more difficult to optimise without significant changes in the machine operation.

Under the assumption that the power consumption in-between two data samples remains constant, each of the previous energies can be calculated as:

$$E = t_s \sum_{k=1}^{K} P_k \tag{2}$$

where $t_s = \frac{1}{f}$ represents the sampling interval in seconds and f is the sampling frequency, K equals the number of samples in the specific period and P_k is the measured active power consumption at the data point k.

3. Feature extraction through time-series segmentation

The sequences of power data points along the time axis mean that the extraction of the required features can be reduced into a segmentation problem, where only segments of interest are retained. Time-series segmentation is a problem that has become significant over the recent years, particularly with the appearance of the Internet of Things and the necessity to process large amounts of data in order to generate knowledge in real-time. In the literature, authors commonly refer to this as the time-series partitioning or the change-point detection problem, but all the algorithms under these names seek the same solution. It is often one of the preprocessing stages of time-series streams which facilitates smart feature extraction, modelling, filtering and sometimes reduction of transmitted data [21,37,25].

Traditional clustering algorithms like k-means or Gaussian Mixture Models (GMM) are not appropriate without modifications for this task, as they do not model the temporal dynamics of the power signal. Keogh et al. [30] reports that typical solutions include the sliding-window algorithm, which is a simple and fast solution that can be applied in real-time, but performs with poor accuracy in many cases or dynamic programming (DP) approaches such as the top-down algorithm, which is a heuristic algorithm and the bottom-up algorithm, which is the complement to the former. Use of more advanced, modified DP algorithms can be found, for example, in hydrometeorological time-series segmentation studies [22,23]. Fu [21] provides a very good review of the most common time-series partitioning techniques.

Hidden Markov model (HMM) based techniques have been used repeatedly for this purpose, due to their suitability to model sequential data [46]. HMMs are not a new concept and variations have been extensively used in areas such as speech recognition applications [42] and genome analysis [33]. For example, in Kehagias [28] and Kehagias and Fortin [29] the authors use HMM-based algorithms to segment environmental data.

A caveat with HMM-based algorithms is that there can exist cases where a HMM may fail to model adequately the temporal persistence of the states, if the signal contains a lot of noise. This is a known limitation of HMMs which sometimes leads to undesired rapid state switching and modifications accounting for this issue have been proposed in literature. One solution that provides improved results is the sticky HDP-HMM, which introduces a state self-transition bias [20]. A different approach is to extend a finite or an HDP-HMM by considering explicit-duration semi-Markov modelling [27]. This solution overcomes the geometrically distributed state duration restriction and may prevent errors from appearing when the experiments include more complex operations as part of the machining cycles.

However, because HMM-based segmentation techniques have not been applied to applications concerned with machining power data before, the simplest form of HMMs will be used in this research, rather than an HDP-HMM or a semi-Markov model. If one of the aforementioned problems is encountered, one of those more advanced HMM-based algorithm can be implemented as part of further work.

3.1. A Hidden Markov model with Gaussian Mixture emissions

The generic Hidden Markov Model is a type of a probabilistic graphical model where a system is modelled as a Markov chain consisting of latent states (s) described by a stochastic process. These states can only be observed through another stochastic process that creates a sequence of observations (y) [42]. A visual representation of this can be seen in Fig. 3.

There are many modifications and extensions of an HMM. In its simplest form, a general, homogeneous HMM can be described by 3 elements: the initial probability distribution (π), the transition matrix (a) and the emission probability distribution (b). The transition matrix incorporates the probabilities that control the transition $s_t \rightarrow s_{t+1}$, while the emission probabilities describe how the observable variable y behaves when the system is in a certain state S_k at time t.

In the most simple case, the emissions distribution is described by a discrete probability distribution. Therefore, if N is the number of latent states, the discrete Hidden Markov model can be fully characterised as:

$$\begin{aligned}&\pi_i = P(s_1 = S_i), 1 \leqslant i \leqslant N\\&a_{ik} = P(s_{t+1} = S_k | s_t = S_i), 1 \leqslant i,k \leqslant N\\&b_k(y) = P(y|s_t = S_k), 1 \leqslant k \leqslant N\end{aligned} \tag{3}$$

If a continuous probability distribution is used to model the emissions distribution associated with each hidden state, the HMM can be used to model continuous signals, without having to discretise them first. In a typical scenario, the Gaussian distribution ($\mathcal{N}$) is commonly preferred, so $b_k(y)$ can be described as:

$$b_k(y) = P(y|s_t = S_k) = \mathcal{N}(y; \mu^k, \sigma^{2\ k}) \tag{4}$$

where μ^k is the mean and $\sigma^{2\ k}$ is the variance of the Gaussian distribution associated with state S_k.

In this scenario however, the assumption that the observations of each state follow a normal distribution is made, which is generally not true. A mixture of Gaussian distributions is a better choice, as it can be used to approximate any other finite and continuous probability distribution that might appear in the observation sequence, given enough mixture components, as proven in Li and Barron [35]. In two of the steps of the methodology proposed in Section 4, a variant of the basic HMM is used, where the power signal observations are treated as a set of Gaussian Mixture models (GMM). This is called a Gaussian Mixture Hidden Markov model (GMHMM) and has been used repeatedly in different applications such as time-series classification [38], but not for the purpose of partitioning time-series energy data in manufacturing. The emissions of a GMHMM can then be described in the following form:

$$b_k(y) = P(y|s_t = S_k) = \sum_{m=1}^{M} c_{km}\mathcal{N}(y; \mu^{km}, \sigma^{2\ km}) \tag{5}$$

where c_{km} are the mixture weights, μ^{km} is the mean and $\sigma^{2\ km}$ is the variance for the mth mixture component in state S_k. The main limitation of a GMHMM is the fine-tuning of the hyper-parameters, such as the number of hidden states N and the number of mixture components M, although there are solutions that enable the best-fit model to be selected automatically,

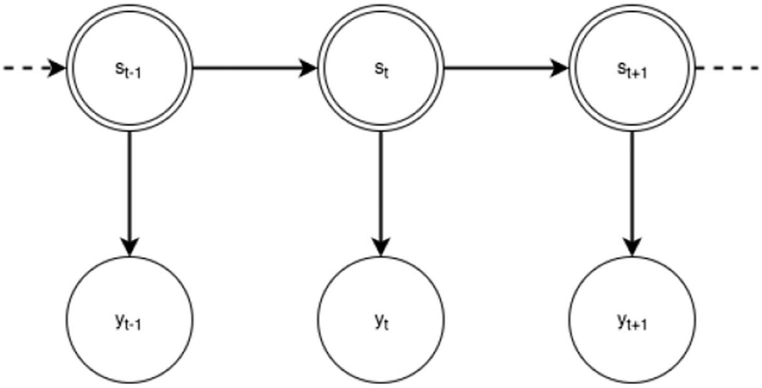


Fig. 3. The basic Hidden Markov model structure. s_t is the latent state at time t and y_t is the observation dependent on s_t at time t. (Figure generated from Rabiner [42]).

such as the Bayesian Information Criterion (BIC) [43]. However, in this research the energy signal shape is known a priori, so these hyper-parameters will be selected manually and need not be automatically adjusted.

The standard algorithm to train any HMM from either single or multiple observations sequences is the Baum-Welch algorithm, which is based on the Expectation–Maximisation algorithm [14]. The general HMM Baum-Welch model parameter update step can be adjusted as per [42] to account for the Gaussian mixtures modification of the emission probability density function.

By knowing the model parameters, the Viterbi algorithm [19] can be applied to find the single most probable path of hidden states, considering the whole observation signal. In this case, the hidden states represent separate machine states, or a group of machine states, each tied to a certain energy consumption distribution, where the collected data samples are assumed to have been generated from.

4. Proposed method for feature extraction

Fig. 4 outlines the method proposed in this paper, which relies on consecutive splitting and merging of segments to form and extract the desired energy state sequences. The methodology followed to develop this technique is based on the concept that by proper selection of the hyper-parameters, a set of GMHMM training and decoding passes (Section 4.1 and 4.2) can split the signal at the correct points. However, there are also supporting cycles (as seen in Fig. 2) that need to be discarded as they do not contribute to the results. The algorithm to implement this is described in Section 4.2 and its accuracy is enhanced by the user's input to correct any misidentified cycles in Section 4.3.

4.1. Extraction of cutting energies E_c

The extraction of the cutting energies E_c is a straightforward procedure. A GMHMM is trained on the sampled power signal and then the generated model is used to decode the underlying behaviour by applying the Viterbi algorithm, as described in Section 3.1. A prior selection of $N = 3$ hidden states leads to the following clustering of the time-series data-points:

1. The standby states
2. The material cutting states
3. Anything that does not fit into one of the previous categories

Having isolated each material cutting state, E_c can be readily calculated for each machining cycle from Eq. 2.

The reason this approach works well is because of the very different dynamics of each category. The standby and the cutting state data points originate from distributions with small variances. The third state includes mainly rapidly changing signals (such as the spindle acceleration or rapid movements from the feed motors) or signals that do not follow a specific pattern (setup or calibration procedures and other supporting cycles, some of which are run manually by the machine operator; examples can be seen in Fig. 2).

The second hyper-parameter that is required to be set is the number of Gaussian distributions. By trying different numbers of mixture components ($M \in \{8, 16, 32, 64\}$), it was observed that there was no practical effect on the algorithm's clustering accuracy, as long as $M \geqslant 8$. No over-segmentation was observed due to over-fitting.

4.2. Filtering out the support cycles

At this point, all the E_{su} consumptions are included in the third cluster group but cannot be extracted yet, as in addition to those, this category comprises a variety of signals as mentioned, such as air-cutting states, rapid feeds and support cycles. All of these signals need to be filtered out first, as some of them include shapes and frequency components similar to the spindle acceleration operation. However, this is not trivial by working only with the third cluster of data points, as there are no clear criteria to distinguish the spindle acceleration state from all the rest.

To facilitate this task, the process is decomposed into several steps. The active cycles (which are formed by excluding the standby states from the initial power signal) need to be separated into machining cycles and support cycles (see Figs. 2 and 4) and retain only the former. The expected shape of a machining cycle is known in advance, so it will be easier to identify later the spindle acceleration states inside these segments. In addition, since the machining cycle shapes are also almost identical to each other irrespective of the cutting parameters, any support cycles with different shapes can be clustered out of the active cycles.

To separate the active cycles as described, a method of time-series clustering is needed. The agglomerative (or bottom-up) hierarchical clustering combined with the Dynamic Time Warping (DTW) algorithm have been used in the past successfully for clustering of time-series data (see Aghabozorgi et al. [3] for examples). The bottom-up linkage algorithm works as follow: Beginning with the maximum number of clusters (i.e. each segment constitutes a cluster on its own), the algorithm merges at each step the two clusters with the shortest distance apart. This procedure is repeated until the terminating condition is reached, i.e. when only two clusters exist (for this case). At this point, all the active cycle segments have been allocated to one of the two final clusters, either machining or support.

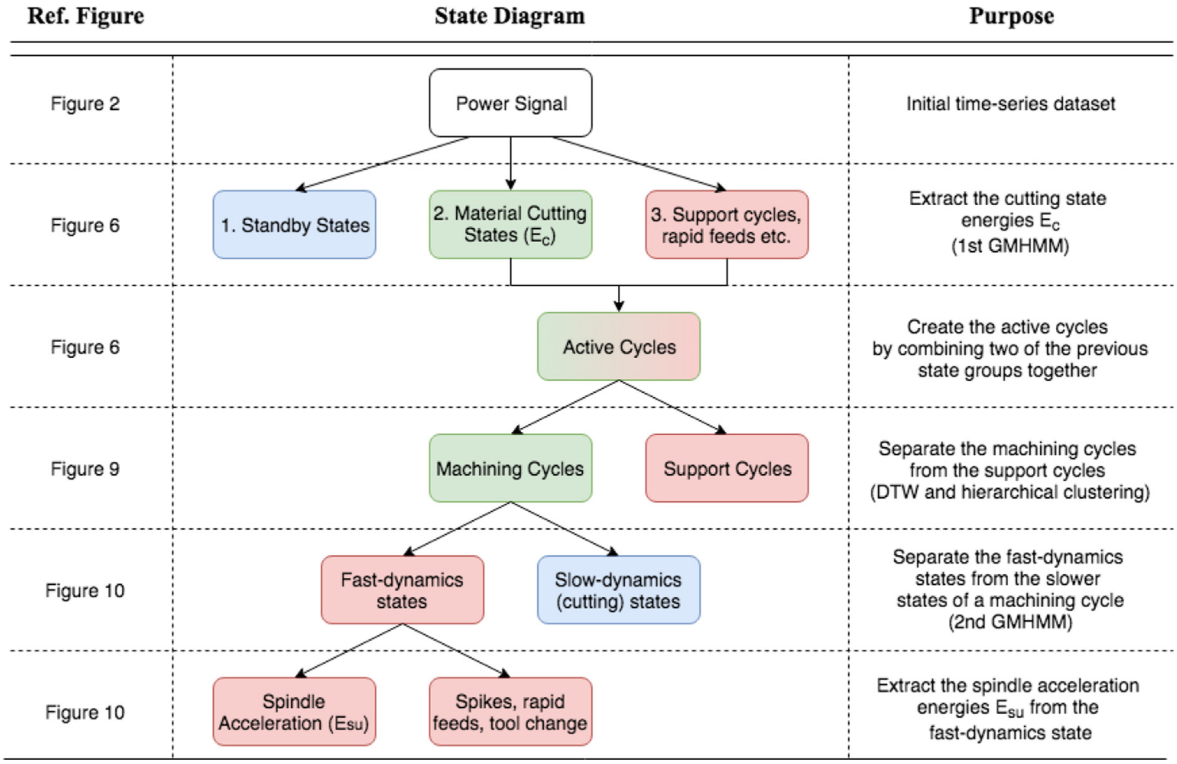

Ref. Figure
State Diagram
Purpose
Figure 2
Power Signal
Initial time-series dataset
Figure 6
1. Standby States
2. Material Cutting States (E_c)
3. Support cycles, rapid feeds etc.
Extract the cutting state energies E_c (1st GMHMM)
Figure 6
Active Cycles
Create the active cycles by combining two of the previous state groups together
Figure 9
Machining Cycles
Support Cycles
Separate the machining cycles from the support cycles (DTW and hierarchical clustering)
Figure 10
Fast-dynamics states
Slow-dynamics (cutting) states
Separate the fast-dynamics states from the slower states of a machining cycle (2nd GMHMM)
Figure 10
Spindle Acceleration (Esu)
Spikes, rapid feeds, tool change
Extract the spindle acceleration energies E_{su} from the fast-dynamics state

Fig. 4. Proposed method flowchart, indicating state splitting/merging steps. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

After the first step of this procedure, each cluster comprises multiple time-series segments, so different methods to calculate the cluster proximity for the merging step exist. For example, the mean distance between elements of each cluster or the shortest distance can be used. In this case, single-linkage, complete-linkage, unweighed and weighted average-linkage methods were compared with all yielding similar results on our dataset. Further experimentation is needed to conclude which works best in the general case.

To apply the hierarchical clustering algorithm, the distance between any two time-series segments has to be calculated. The definition of the distance between two temporal sequences can be ambiguous, as opposed to distances between two points. The DTW algorithm was used for the pairwise comparison of the active cycles and the formation of a distance matrix, which can describe their dissimilarity.

The DTW basically follows a dynamic programming approach to perform a non-linear time-stretching operation in order to compare two vectors. This technique can provide much more accurate results than a Manhattan point-to-point distance, mainly when one of the sequences is a time-shifted or time-warped version of the other and allows comparison of series of different lengths. Each pair of machining cycles is expected to have a very small DTW distance, because they originate from the same operation, though their durations and thus sequence lengths can differ.

4.3. The user-revision step

After the clustering algorithm is finished, the user can revise the results by using the GUI of Fig. 9. This step is aimed at correcting any clustering errors that have occurred from the agglomerative clustering algorithm. The dropdown menu with the first button allows for reallocation of segments that have been misidentified and the second button saves the results. An error will be generated if the user chooses to update the results and proceed while there is a mismatch between the identified number of machining cycles and the number expected from the DoE of the RSM.

4.4. Extraction of E_{su} energies

After the machining cycles have been separated from the support cycles, the extraction of the E_{su} energies can proceed. A new GMHMM needs to be trained using all the machining cycle sequences to generate a model that will be able to identify the hidden state representing the spindle up process. The GMHMM hyper-parameters are now set to $N = 2$ and $M = 1$, i.e. two hidden states are used and only a single Gaussian distribution to model the observations of each state, as the complexity needed is lower than previously. The emission probability distribution is simplified to Eq. (4).

Once the GMHMM has been generated, each machining cycle is decoded separately using the Viterbi algorithm again. The result is two hidden states; the first one tends to describe the fast dynamics of the spindle acceleration, while the second one models slower behaviour, comprising the air-cutting period and the material cutting period. The fast-dynamics state may pick up rapid feeds or other spikes as well, if they exist.

To understand this better, a real example application from a machining case study can be seen in Fig. 10. In most of the cycles, the spindle acceleration can be extracted readily, but some of these include additional noise components and the last sequence includes an additional rapid tool retract state, which needs to be filtered out. At this point, it is known that the spindle acceleration duration will always be longer than any spike or rapid tool movement, so the correct segment can be extracted by detecting the longer subsequence of the fast-dynamics state. E_{su} energies are then calculated using Eq. (2).

5. Case study validation

5.1. Testing methodology

A Central Composite Design (CCD) of experiments [10] was used to test and validate the proposed feature extraction techniques. The CCD is the most popular DoE method for the RSM. It has been used repeatedly to reduce the number of test runs while still exploring the experiment space at a good level, as for example in [11] where the authors reduce the number of runs from 625 to 31 for a 4-factor design. A set of 20 runs was designed, which can be used to study the effect that cutting speed, feed rate and depth of cut have in the spindle acceleration and the material cutting energy consumptions during a linear end-milling process, using an RSM. The CCD was structured as a circumscribed design with 6 central points, 5 different levels for each factor and $a = 1.682$. The minimum and maximum levels of interest of the optimisation factors are outlined in Table 2. It is worth noting that some of the CCD runs include extreme low and high values outside this range (but still within the region of operability). This is expected in a circumscribed CCD as it helps with better estimation of the curvature. The run order was randomised as seen in Table 3. All the spindle speed and feed rate values were rounded to the closest integer, while the depth of cut values to the nearest second decimal digit.

For the experimental procedure, a three-phase Hurco VM1 CNC machine was used. The machine has a maximum spindle speed of 8000 rpm achieved through an 11 kW motor and a maximum feed rate of 7.62 m/min. All the experiments were conducted on the same day using an 8 mm 4-flute High-Speed Steel (HSS) end mill tool and both sides of an aluminium alloy 6082 T651 block with dimensions 320 mm × 100 mm × 30 mm (Note: a single block of twice the size would not fit into the machine worktable and for this reason, the experiments were carried out in two stages). The machined workpiece can be seen in Fig. 5. The RMS data were captured at a sample rate of $f = 5$ Hz using a National Instruments^{TM} CompactDAQ chassis with the NI-9247 and NI-9242 modules and LabVIEW software.

The CNC G-Code program was written so that there is a 5 s delay between consecutive runs, during which the spindle speed is reduced to zero and the tool stays still. The delay is placed immediately after the end of the material cutting phase, as seen in Fig. 6. This convention helps at the first step of the segmentation procedure by inducing a standby phase between two cuts, so that two machine cycles can be separated from each other.

5.2. Results

The output from training and decoding using the first GMHMM model can be seen in Fig. 6. The standby state is represented by the blue line, the material cutting state is in green and everything else in red, as presented in Fig. 4. It can be seen clearly that the experiments were carried out in two stages.

The cutting energies E_c were automatically calculated from the green segments and compared with the real consumptions, which were calculated manually by an expert. For the manual procedure, the material cutting cycle was considered to begin at the point of the rising edge where the First-Order Finite Central Difference (FOFCD, see: Eq. (6)) of the discrete-time function of the real power consumption P was maximum. An example can be seen in Fig. 8. Similarly, the end of the material cutting cycle was assumed to coincide with the minimisation of the same quantity, along the points of the falling edge of the power signal.

$$FOFCD(t) = \frac{P(t+h) - P(t-h)}{2h} \tag{6}$$

Table 2
Study range of the optimisation parameters.

Studied Factor	Minimum	Maximum
Spindle Speed (rpm)	3000	5000
Feed Rate (mm/min)	300	600
Depth of Cut (mm)	3	6

Table 3
Central Composite Design of Experiments.

Run Order	Initial CCD Order	Spindle Speed (rpm)	Feed Rate (mm/min)	Depth of Cut (mm)
1	14	4000	450	7.02
2	2	3000	300	6.00
3	15	4000	450	4.50
4	10	5682	450	4.50
5	1	3000	300	3.00
6	3	3000	600	3.00
7	16	4000	450	4.50
8	12	4000	702	4.50
9	8	5000	600	6
10	17	4000	450	4.50
11	13	4000	450	1.98
12	18	4000	450	4.50
13	19	4000	450	4.50
14	9	2318	450	4.50
15	11	4000	198	4.50
16	6	5000	300	6.00
17	20	4000	450	4.50
18	7	5000	600	3.00
19	4	3000	600	6.00
20	5	5000	300	3.00

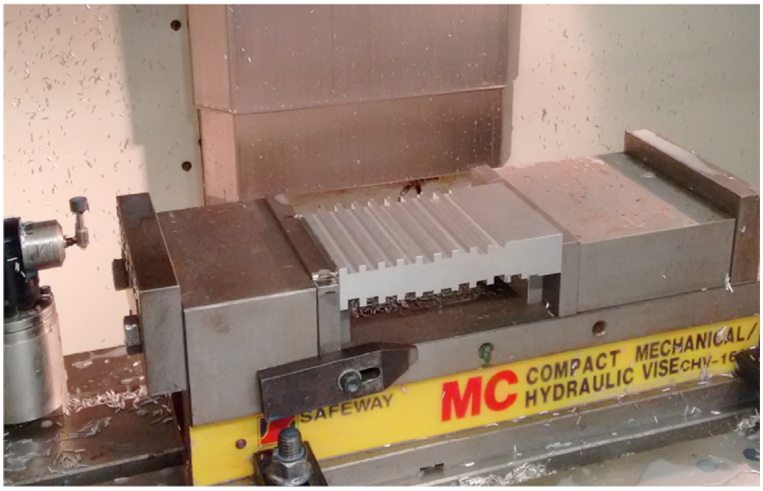


Fig. 5. Workpiece at the end of experiments.

where $h = 1$ in this case.

The results of the E_c energies comparison can be seen in Table 4. The Relative Percentage Error (RPE) for each run i is defined by Eq. (7):

$$RPE_{(i)} = \frac{E_{true(i)} - E_{extracted(i)}}{E_{true(i)}} \cdot 100\% \tag{7}$$

The Mean Absolute Percentage Error (MAPE) has been used as an indicator of the average performance of the algorithm on the entire dataset, as defined by Eq. (8):

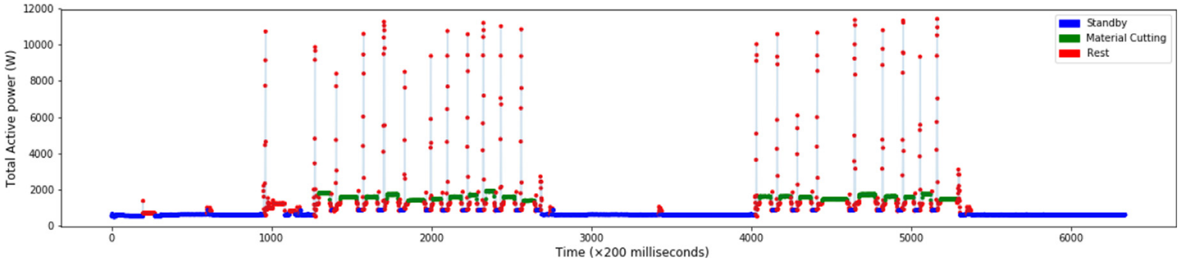


Fig. 6. Energy profile segmentation for the CCD experiment: Extraction of E_c energies.

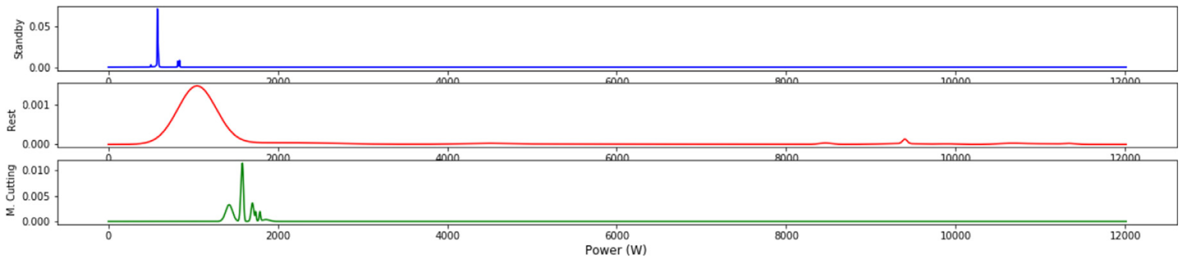


Fig. 7. Emission probability distributions for the hidden states of the first GMHMM. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

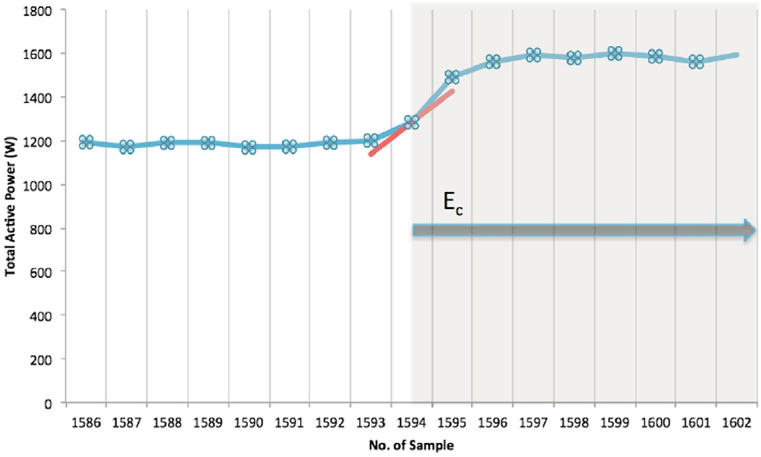


Fig. 8. Example of manual identification of material cutting cycle start.

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{E_{true(i)} - E_{extracted(i)}}{E_{true(i)}}\right| \cdot 100\% \tag{8}$$

where $N = 20$ in this case. A $MAPE = 1.12\%$ was reported during the extraction of the cutting energies E_c.

The probability distribution of each GMM can be seen in Fig. 7. The first (blue) distribution is very narrow and describes the standby state. The second (red) distribution is approximated almost entirely by a single Gaussian distribution and is

Table 4
Material cutting energies (E_c).

Run Order	Manually Extracted E_c (Joules)	Automatically Extracted E_c (Joules)	Relative Percentage Error
1	24544.5	24239.3	1.24%
2	32149	31886.7	0.82%
3	21690.4	21433.7	1.18%
4	23692.6	23430.7	1.11%
5	28305.1	27268.9	3.66%
6	15209.7	14959.3	1.65%
7	21443.6	21443.6	0%
8	14828.3	14828.3	0%
9	19429.2	19429.2	0%
10	21762.5	21500.5	1.2%
11	18830.1	18051.9	4.13%
12	21610.9	21610.9	0%
13	21878	21314.7	2.57%
14	21413.2	21154.9	1.21%
15	44897.4	44716	0.4%
16	34852.1	34583.2	0.77%
17	21896.6	21896.6	0%
18	16269.5	16007.6	1.61%
19	17968	17968	0%
20	29945.4	29679.2	0.89%
Mean Absolute Percentage Error (MAPE):			**1.12%**

more sparse because it includes various signals as explained in Section 4.2. The third distribution (green) needs more Gaussian components to be approximated because each cutting state has a different average level and thus results in multiple peaks.

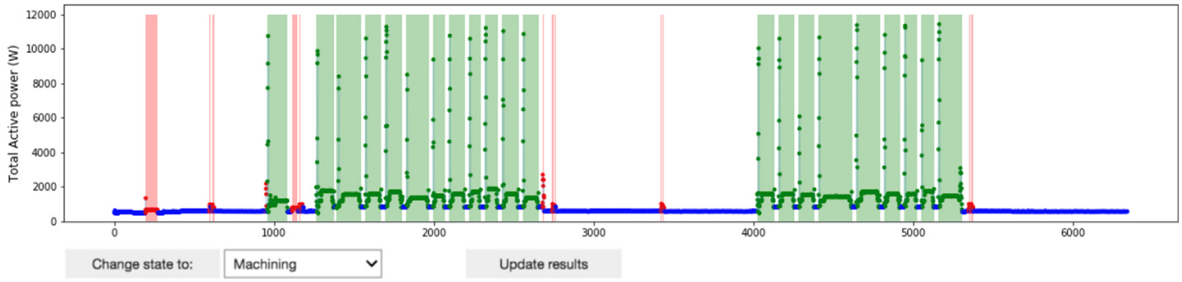


Fig. 9. Hierarchical clustering for separation between DoE runs and supportive operations. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

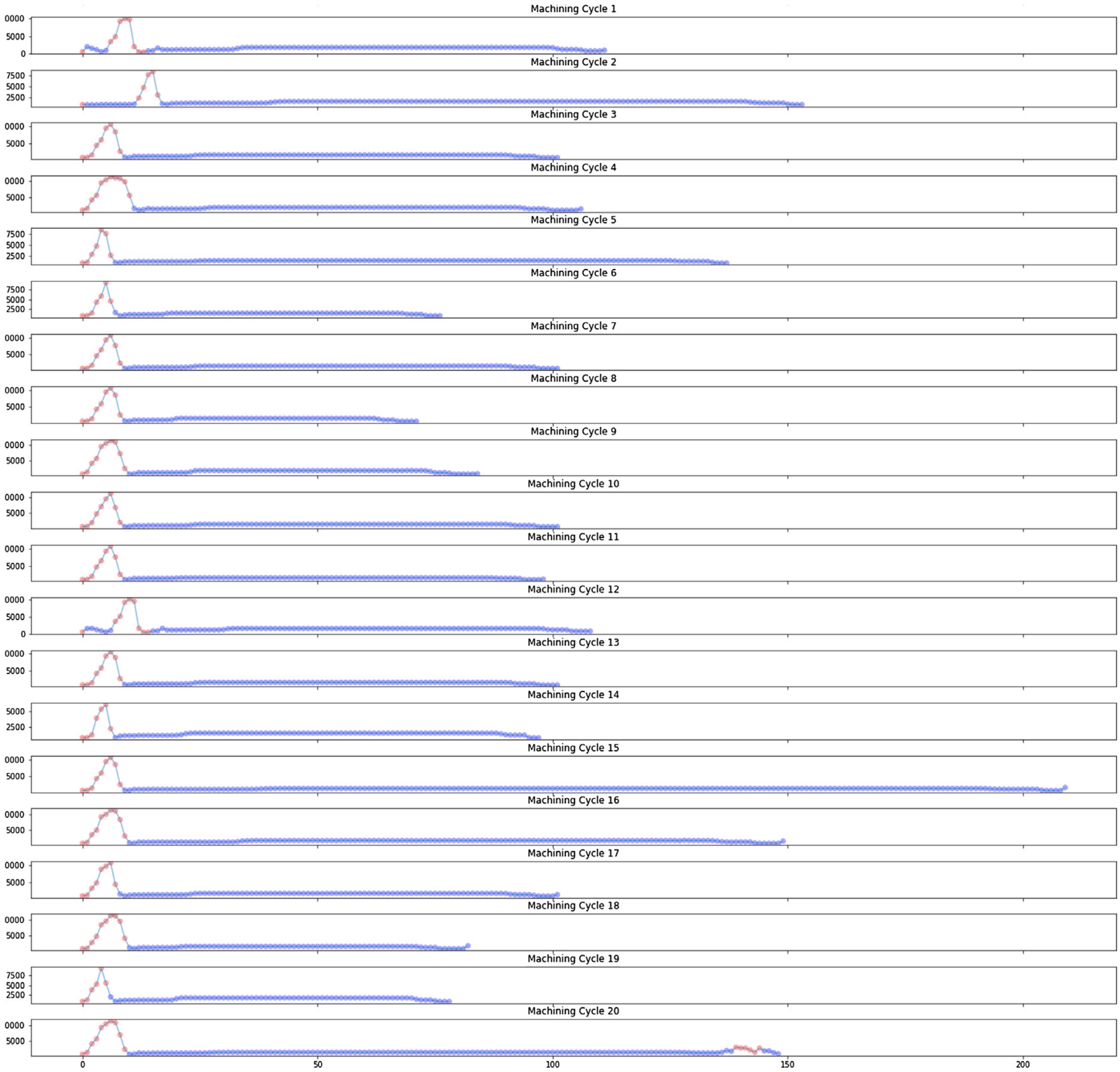


Fig. 10. Energy profile segmentation for the CCD experiment: Extraction of E_{su} energies.

Table 5
Spindle acceleration energies (E_{su}).

Run Order	Manually Extracted E_{su} (Joules)	Automatically Extracted E_{su} (Joules)	Relative Percentage Error
1	7793.9	7988.5	−2.5%
2	5639.3	5246.6	6.96%
3	8803.8	8963.3	−1.81%
4	16815	16045.1	4.58%
5	5818.5	5617.7	3.45%
6	5626.1	5468.9	2.79%
7	8619.3	8961.5	−3.97%
8	8621.2	8960.5	−3.94%
9	12567.9	12736.1	−1.34%
10	8636.6	8987.8	−4.07%
11	8628.2	8973	−4%
12	7990.9	7985.7	0.06%
13	8630.3	8970.4	−3.94%
14	3789.4	4127.6	−8.93%
15	8632.9	8975	−3.96%
16	12766.5	12746.2	0.16%
17	8817	8707.6	1.24%
18	12785.2	12722.6	0.49%
19	5606.4	5213	7.02%
20	12620.2	12789.7	−1.34%
Mean Absolute Percentage Error (MAPE):			**3.33**%

It is worth noting that when grouping the non-standby states together to form the active cycle, the GMHMM algorithm achieves a 99.12% accuracy in distinguishing between the active and the standby cycles, so this method can also be readily used for the calculation of the duty cycle of a CNC machine.

The next step is the separation of the supporting cycles from the machining operations. The hierarchical clustering results can be seen in the GUI of Fig. 9. This is before any user input. The green segments in this figure represent the DoE runs and the red segments identify the supporting operations. All of the machining cycles are detected apart from the rapid feed at the end of the first stage of experiments, due to a small standby period existing between this and the preceding material cutting. The only falsely identified machining cycle is the test cut at [957,1084] (first green segment) because of its shape which is similar to the DoE cycles. These errors are the reason the hierarchical clustering reports 93.60% accuracy and both can be easily corrected manually by the user as indicated in Section 4.3.

The final step comprises the extraction of the E_{su} energies. The second GMHMM is trained using all the machining cycle sequences and then each cycle is decoded separately. The results are seen in Fig. 10. Noise is picked up at the beginning of some cycles and the last run includes a tool retracting operation, which is also identified as belonging to the fast-dynamics state, but both will be automatically excluded from the E_{su} calculations as described in Section 4.4. The longest sequence of red points in each machining cycle represents the power consumed during the acceleration of the spindle motor.

The comparison between the automatically extracted and the manually calculated E_{su} energies can be found in Table 5. The *RPE* from Eq. (7) was used again to measure the performance of the algorithm at each run and a $MAPE = 3.33\%$ was reported after the extraction of all the E_{su} energies.

It should be noted that the main reason for the large variability of the E_{su} *RPE* (see runs 2, 14 and 19 in Table 5) is the fast dynamics of the spindle acceleration state combined with the short duration (1–2 s) of each spindle acceleration operation and the low sampling rate used for the data collection (5 Hz). Due to the 5 Hz sample frequency, only 5–10 datapoints are collected during each spindle acceleration operation. This means that if the proposed method fails to identify even one point, this results into 5–10% less extracted spindle acceleration points. However, due to the bell-curved shape of the spindle acceleration state, any missed points are either at the beginning or the end of that, so they contribute less to the E_{su} energy, as opposed to points in the middle.

The *MAPE* errors for the two features indicate that the proposed method is of significant value in automatically determining the energy features studied in this research.

6. Conclusions & future work

In this work, a method which allows key energy-related features to be extracted from time-series power data through sequential segmentation, clustering and simple filtering of the power signal has been presented. To be effective, the DoE must be known in advance. The user's input is minimal and only necessary to improve the accuracy when unexpected features are present in the data. A central composite design of experiments was used as a case study to test the proposed algorithms as a way of accelerating the response surface methodology procedures and follow a faster and more standardised route during energy waste optimisation studies, where results will also rely less on human errors.

Two features have been selected to be used for the validation of the algorithms. First and foremost, the material cutting energies E_c, which constitute the key efficiency indicator for a machining process, were detected and extracted with a mean absolute percentage error of 1.12%. Six out of 20 material cutting energies were extracted precisely, with no error at all. From the results of Table 4 it can be observed that the errors were a result of the algorithm underestimating the duration of the material cutting process. On the other hand, the E_{su} energy extraction method was less accurate, reporting 3.33% mean absolute percentage error.

No over-segmentation or rapid state switching was observed when applying the GMHMM algorithm to the data recorded in this research. However, if this is observed when applying the proposed solution to different datasets, there have been suggested modifications of HMMs in literature that can overcome this issue [20,27].

Acknowledgements

The authors would like to thank Mr. Rob Hunter, Engineering Applications Workshop Technician for assistance with operation of the CNC equipment. This project was supported by the EPSRC through the project Adaptive Informatics for Intelligent Manufacturing (EP/K014137/1) and the Centre for Doctoral Training in Embedded Intelligence (EP/L014998/1).

References

[1] L.B. Abhang, M. Hameedullah, Power prediction model for turning EN-31 steel using response surface methodology, J. Eng. Sci. Technol. Rev. 3 (1) (2010) 116–122.
[2] A. Aggarwal, H. Singh, P. Kumar, M. Singh, Optimizing power consumption for CNC turned parts using response surface methodology and Taguchi's technique-A comparative analysis, J. Mater. Process. Technol. 200 (1–3) (2008) 373–384.
[3] S. Aghabozorgi, A. Seyed Shirkhorshidi, T. Ying Wah, Time-series clustering – a decade review, Inf. Syst. 53 (2015) 16–38, https://doi.org/10.1016/j.is.2015.04.007.
[4] C. Ahilan, S. Kumanan, N. Sivakumaran, J. Edwin Raja Dhas, Modeling and prediction of machining quality in CNC turning process using intelligent hybrid decision making tools, Appl. Soft Comput. J. 13 (3) (2013) 1543–1551, https://doi.org/10.1016/j.asoc.2012.03.071.
[5] R.S. Altintaş, M. Kahya, H.Ö. Ünver, Modelling and optimization of energy consumption for feature based milling, Int. J. Adv. Manuf. Technol. 86 (9–12) (2016) 3345–3363, URL http://link.springer.com/10.1007/s00170-016-8441-7.
[6] A. Aramcharoen, P.T. Mativenga, Critical factors in energy demand modelling for CNC milling and impact of toolpath strategy, J. Clean. Prod. 78 (2014) 63–74, https://doi.org/10.1016/j.jclepro.2014.04.065.
[7] I. Asiltürk, S. Neşeli, Multi response optimisation of CNC turning parameters via Taguchi method-based response surface analysis, Meas.: J. Int. Meas. Confed. 45 (4) (2012) 785–794.
[8] O.I. Avram, P. Xirouchakis, Evaluating the use phase energy requirements of a machine tool system, J. Clean. Prod. 19 (6–7) (2011) 699–711, https://doi.org/10.1016/j.jclepro.2010.10.010.
[9] R.K. Bhushan, Optimization of cutting parameters for minimizing power consumption and maximizing tool life during machining of Al alloy SiC particle composites, J. Clean. Prod. 39 (2013) 242–254, https://doi.org/10.1016/j.jclepro.2012.08.008.
[10] G. Box, K. Wilson, On the experimental attainment of optimum conditions, J. Roy. Stat. Soc. 13 (1) (1951) 1–45, URL http://www.jstor.org/stable/2983966.
[11] G. Campatelli, L. Lorenzini, A. Scippa, Optimization of process parameters using a Response Surface Method for minimizing power consumption in the milling of carbon steel, J. Clean. Prod. 66 (2014) 309–316, https://doi.org/10.1016/j.jclepro.2013.10.025.
[12] C. Camposeco-Negrete, Optimization of cutting parameters using Response Surface Method for minimizing energy consumption and maximizing cutting quality in turning of AISI 6061 T6 aluminum, J. Clean. Prod. 91 (2015) 109–117, https://doi.org/10.1016/j.jclepro.2014.12.017.
[13] J.B. Dahmus, T.G. Gutowski, An environmental analysis of machining, ASME – Int. Mech. Eng. Congr. RD&D Expo (2004) 1–10.
[14] A. Dempster, N. Laird, D. Rubin, A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. R. Stat. Soc. Ser. B (Methodol.) 39 (1) (1977) 1–38.
[15] N. Diaz, S. Choi, M. Helu, Y. Chen, S. Jayanathan, Y. Yasui, D. Kong, S. Pavanaskar, D. Dornfeld, Machine tool design and operation strategies for green manufacturing, Proceedings of 4th CIRP International Conference on High Performance Cutting, 2010.
[16] I.F. Edem, P.T. Mativenga, Modelling of energy demand from computer numerical control (CNC) toolpaths, J. Clean. Prod. 157 (2017) 310–321, https://doi.org/10.1016/j.jclepro.2017.04.096.
[17] EEA, 2017. Final energy consumption by sector and fuel. URL https://www.eea.europa.eu/data-and-maps/indicators/final-energy-consumption-by-sector-9/assessment-1.
[18] European Commission, 2015. The 2020 climate and energy packages. URL http://ec.europa.eu/clima/policies/package/index_en.htm.
[19] G.D.J. Forney, The Viterbi algorithm, Prof. IEEE 61 (3) (1973) 268–278.
[20] E.B. Fox, E.B. Sudderth, M.I. Jordan, A.S. Willsky, An HDP-HMM for systems with state persistence, in: Proceedings of the 25th international conference on Machine learning, 2008, pp. 312–319.
[21] T.C. Fu, A review on time series data mining, Eng. Appl. Artif. Intell. 24 (1) (2011) 164–181, https://doi.org/10.1016/j.engappai.2010.09.007.
[22] A. Gedikli, H. Aksoy, N.E. Unal, A. Kehagias, Modified dynamic programming approach for offline segmentation of long hydrometeorological time series, Stoch. Env. Res. Risk Assess. 24 (5) (2010) 547–557.
[23] H. Guo, X. Liu, L. Song, Dynamic programming approach for segmentation of multivariate time series, Stoch. Env. Res. Risk Assess. 29 (1) (2014) 265–273.
[24] Y. He, F. Liu, T. Wu, F.-P. Zhong, B. Peng, Analysis and estimation of energy consumption for numerical control machining, Proc. Inst. Mech. Eng., Part B: J. Eng. Manuf. 226 (2) (2012) 255–266.
[25] J. Himberg, K. Korpiaho, H. Mannila, J. Tikanmaki, H. Toivonen, Time series segmentation for context recognition in mobile devices, in: Proceedings 2001 IEEE International Conference on Data Mining c, 2001, pp. 203–210.
[26] S. Hu, F. Liu, Y. He, T. Hu, An on-line approach for energy efficiency monitoring of machine tools, J. Clean. Prod. 27 (2012) 133–140, https://doi.org/10.1016/j.jclepro.2012.01.013.
[27] M.J. Johnson, A.S. Willsky, 2013. Bayesian Nonparametric Hidden Semi-Markov Models. arXiv preprint arXiv:1203.1365 14, 673–701. URL http://arxiv.org/abs/1203.1365.
[28] A. Kehagias, A Hidden Markov Model Segmentation Procedure for Hydrological and Enviromental Time Series, 2003.
[29] A. Kehagias, V. Fortin, Time series segmentation with shifting means hidden markov models, Nonlinear Process. Geophys. 13 (3) (2006) 339–352.
[30] E. Keogh, S. Chu, D. Hart, M. Pazzani, An online algorithm for segmenting time series, in: Proceedings 2001 IEEE International Conference on Data Mining, 2001, pp. 289–296.
[31] D. Kong, S. Choi, Y. Yasui, S. Pavanaskar, D. Dornfeld, P. Wright, Software-based tool path evaluation for environmental sustainability, J. Manuf. Syst. 30 (4) (2011) 241–247, https://doi.org/10.1016/j.jmsy.2011.08.005.
[32] D.N.D.N. Kordonowy, 2002. A Power Assessment of Machining Tools. Ph.D. thesis.
[33] A. Krogh, B. Larsson, G. von Heijne, E. Sonnhammer, Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes, J. Mol. Biol. 305 (3) (2001) 567–580.

[34] C. Li, Q. Xiao, Y. Tang, L. Li, A method integrating Taguchi, RSM and MOPSO to CNC machining parameters optimization for energy saving, J. Clean. Prod. 135 (2016) 263–275.

[35] Li, J.Q., Barron, A.R., 1999. Mixture density estimation. In: Advances in Neural Information.... pp. 279–285. URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.33.7837%5Cnpapers3://publication/uuid/97445B8B-4AD2-4EC9-BC97-65F6C500796E.

[36] F. Liu, J. Xie, S. Liu, A method for predicting the energy consumption of the main driving system of a machine tool in a machining process, J. Clean. Prod. 105 (2015) 171–177, https://doi.org/10.1016/j.jclepro.2014.09.058.

[37] X. Liu, Z. Lin, H. Wang, Novel online methods for time series segmentation, IEEE Trans. Knowl. Data Eng. 20 (12) (2008) 1616–1626.

[38] A. Mannini, A.M. Sabatini, Machine learning methods for classifying human physical activity from on-body accelerometers, Sensors 10 (2) (2010) 1154–1175.

[39] D.C. Montgomery, Design and Analysis of Experiments, John Wiley & Sons, 2006.

[40] M. Mori, M. Fujishima, Y. Inamasu, Y. Oda, A study on energy efficiency improvement for machine tools, CIRP Ann. – Manuf. Technol. 60 (1) (2011) 145–148.

[41] I. Mukherjee, P.K. Ray, A review of optimization techniques in metal cutting processes, Comput. Ind. Eng. 50 (1–2) (2006) 15–34.

[42] L.R. Rabiner, A tutorial on hidden markov models and selected applications in speech recognition, Proc. IEEE 77 (2) (1989) 257–286.

[43] G. Schwarz, Estimating the dimension of a model, Ann. Stat. 6 (2) (1978) 461–464, URL http://projecteuclid.org/euclid.aos/1176344136.

[44] S. Subramonian, S. Abdul Rahim, Comparison between Taguchi Method and Response Surface Methodology (RSM) in modelling CO2 laser machining, Jordan J. Mech. Ind. Eng. 8 (1) (2014) 35–42.

[45] G. Taguchi, S.S. Konishi, Orthogonal arrays and linear graphs: tools for quality engineering, Am. Suppl. Inst. (1987) 72.

[46] D. Trabelsi, S. Mohammed, F. Chamroukhi, L. Oukhellou, Y. Amirat, An unsupervised approach for automatic activity recognition based on Hidden Markov Model regression, IEEE Trans. Autom. Sci. Eng. 10 (3) (2013) 829–835.

[47] R. Unal, E.B. Dean, Taguchi approach to design optimization for quality and cost: an overview, Annu. Conf. Int. Soc. Param. Anal. (1991) 1–10.

[48] J. Yan, L. Li, Multi-objective optimization of milling parameters-the trade-offs between energy, production rate and cutting quality, J. Clean. Prod. 52 (2013) 462–471, https://doi.org/10.1016/j.jclepro.2013.02.030.

[49] H.S. Yoon, E.S. Kim, M.S. Kim, J.Y. Lee, G.B. Lee, S.H. Ahn, Towards greener machine tools – a review on energy saving strategies and technologies, Renew. Sustain. Energy Rev. 48 (2015) 870–891, https://doi.org/10.1016/j.rser.2015.03.100.

[50] H.S. Yoon, J.S. Moon, M.Q. Pham, G.B. Lee, S.H. Ahn, Control of machining parameters for energy and cost savings in micro-scale drilling of PCBs, J. Clean. Prod. 54 (2013) 41–48, https://doi.org/10.1016/j.jclepro.2013.04.028.

[51] G.Y. Zhao, Z.Y. Liu, Y. He, H.J. Cao, Y.B. Guo, Energy consumption in machining: classification, prediction, and reduction strategy, Energy 133 (2017) 142–157.

[52] L. Zhou, J. Li, F. Li, Q. Meng, J. Li, X. Xu, Energy consumption model and energy efficiency of machine tools: a comprehensive literature review, J. Clean. Prod. 112 (2016) 3721–3734, https://doi.org/10.1016/j.jclepro.2015.05.093.