A fuzzy logic map-based knock control for spark ignition engines

Benjamín Pla[a], Pau Bares[a], Irina Jimenez[a], Carlos Guardiola[b], Yahui Zhang[c], Tielong Shen[c]

[a] *CMT-Motores Termicos, Universitat Politecnica de Valencia, Camino de Vera s/n, E-46022 Valencia, Spain.*
[b] *Universitat Politecnica de Valencia, Departamento de Máquinas y Motores Térmicos Camino de Vera, s/n. 46022 Valencia, Spain.*
[c] *Department of Engineering and Applied Sciences, Sophia University, Chiyoda-ku Tokyo, 102-8554, Japan.*

Abstract

Knock control represents one of the most critical aspects to reach optimal thermal efficiency in spark ignition engines, and its research is crucially important because it determines thermal efficiency, engine durability, and power density, as well as noise and emission performance. In this paper, a spark advance control based on a map learning technique is combined with a knock estimator to maximize the engine efficiency while keeping the knock probability below a desired limit. The proposed controller is experimentally validated on a production spark ignition gasoline engine test bench, and compared with a conventional spark advance controller in both, steady and transient conditions. From experimental results, a benefit in terms of thermal efficiency, control stability and engine security are achieved. The results show that the proposed method is capable of regulating the knock probability to a target percentage with low spark advance and thermal efficiency dispersion than the conventional controller.

Keywords: combustion control, spark advance, knock, map learning

1. Introduction

Spark ignition engines are a widespread technology in the field of road transport, particularly in the segment of passenger cars [1]. Considering that road transport accounts for the largest proportion of energy consumption in the transport sector (73 % in 2017 according to European Environment Agency), the need of improving engine efficiency becomes evident. Methods to improve thermal efficiency include increasing compression ratios, reducing thermal losses or turbocharging, but all of them have the avoidance of abnormal combustion such as knock as a major challenge due to the associated increase in the combustion chamber temperature [2]. Knock is an abnormal combustion that occurs when a portion of the fuel-air mixture is auto ignited [3]. This rapid combustion produces shock waves which are propagated across the cylinder and heavily excite the cylinder resonant modes [4]. Knock is an undesirable phenomenon that reduces the engine efficiency, produces vibration noise, and heavy knock might cause damage to the engine [5]. Ignition timing is one of the most important parameters to optimize combustion in spark ignitionengines, because affects the combustion phase which is related with engine performance, emissions and fuel efficiency [6]. The spark advance (SA) is usually controlled to maximize efficiency. However, for medium and heavy load, the maximum of efficiency cannot be reached due to knock phenomenon [3].

In simpler SI engine control units (ECUs), SA is usually open-loop controlled by look up tables previously calibrated [7]. The problem of this method is that there is no feedback signal, and knock onset is affected by engine operating condition, fuel quality, and other factors, so a closed-loop knock control is usually required [8]. Due to its simplicity, vibration measurement is widely used in industry as closed-loop signal for knock detection. However, the quality of knock detection is often affected by noise and vibrations [3]. Cylinder pressure is directly affected by knock and provide reliable information, for that reason, in-cylinder pressure sensors are more accurate for knock detection and largely employed in laboratory tests [9]. One of the most common reference metrics for knock detection based on in-cylinder pressure is MAPO (Maximum Amplitude of Pressure Oscillations), which computes the maximum absolute value of the filtered pressure signal [10].

Regarding SA close-loop control strategies, they can be divided into two groups: stochastic and model-based controls [11]. Stochastic methods maintain the knock probability under a desired threshold by varying the SA while model-based methods aim to predict the timing at which knock occurs. The most widely used stochastic method referred to in literature as conventional strategy [12], consists of advancing the spark angle by an amount K_{adv} to improve engine efficiency, while retarding a higher amount K_{ret} when a knocking cycle is detected in order to avoid engine damage as:

$$SA^{i}_{conv} = \begin{cases} SA^{i-1}_{conv} - K_{ret} & if \quad knock \\ SA^{i-1}_{conv} + K_{adv} & otherwise \end{cases} \tag{1}$$

where i denotes the cycle number, and K_{adv}, K_{ret} are

controller gains. K_{ret} is much larger than K_{adv} so the spark slowly advances during non knocking periods, but it is rapidly retarded if a knock event occurs. These two gains are related by the knock probability, as follows:

$$K_{ret} = \frac{1 - p_{th}}{p_{th}} K_{adv} \tag{2}$$

The relation between K_{adv} and K_{ret} defines the percentage of knocking cycles while the value of K_{adv} characterizes the time response of the controller: on the one hand, high values might allow the controller to reach the optimal value faster when it is far from it, but on the other hand the variation of the SA will be higher and hence, higher knock intensities are also expected. Due to its easy implementation and tuning, the conventional knock controller is widely used in industrial applications, but the continuous evolution of the spark timing and overreactions to the knock event needed to preserve the knock probability result in a high variance in the spark timing [13] and therefore combustion phasing and associated performance.

Other stochastic methods, instead of only modify the spark timing when the difference between the observed and desired knock frequency exceeds a defined threshold. Many authors in recent years have published different stochastic control strategies, a stochastic algorithm based on cumulative summation (CumSum) combines fast corrective action with tight regulatory control about the knock limit was presented in [14]. Additional improvements of this method are presented in [15], where a threshold optimization technique is applied to optimize the feedback signal for the CumSum knock controller. Further work, use the likelihood ratio of observed events to scale the advance and retard gains [16]. Recently, a new dual-threshold knock controller was presented in [17] where the binary knocking and no-knocking classification used by most knock-event-based controllers was extended to a ternary knocking/not-knocking/indeterminate classification using two knock thresholds, this reduced the chance of misclassifying knocking cycles as non-knocking. Reducing the misclassification of knocking events allows to increase the controller gain, which leads to faster response [18]. A further method is presented in [19], where a beta distribution based stochastic knock control is proposed, knock probability estimation is performed by a Bayes rule and beta distribution and likelihood ratio test is used for control decision. Stochastic methods shows good results [20], but have a delayed response in transient operation, since the statistical knock properties must be estimated in real time [21].

Model-based methodologies aim to predict the timing of the knock occurrence. Some methods use physical models such as auto ignition prediction by using the Arrhenius equation, for example Livengood-Wu integral method presented in [22]. Other methods are map-based models, that have the advantage of being easily calibrated, but the main problem by using this look-up tables is that variables change during operation and there is a risk of exploring dangerous knocking conditions [23]. To avoid this, different on-board map learning techniques for knock control were published in recent years. For example an extremum seeking and a 2D feed forward map learning algorithm is presented in [24]. In order to control the SA with constrained knock probability other method is shown in [25], where a knock probability constrained SA control framework based on the on-board map adaptation of knock event distribution is presented. In this approach, two maps, efficiency map and knock probability map, are updated in real time according to the measured signals, knock event signal, crank angle position where the 50% of the heat is released (CA50), intake manifold pressure and efficiency. The problem with these control strategies is that need a previous calibration in order to obtain initial knock probability maps and tend to control knock with a probability thresholds around 1%. This is a problem in transient conditions since a large number of cycles is necessary to obtain a 1 % of knock probability with high enough resolution. As demonstrated in [26], working at high knock rates allows to shorten the time required to identify the stochastic properties of knock and, therefore, update the knock models, and the performance of the controllers is very dependent on the thresholds that define the knock-event feedback signal.

In order to cope with the high variance associated to stochastic methods and the large number of cycles required by map-based adaptive methods, this article presents a fuzzy logic map based knock controller for SI engines, applying a detection method able to discern low intensity knocking cycles and to update a knock event probability map. The map is learned with the intake manifold pressure and spark advance as inputs, and the spark advance is restricted by the probability threshold that is subsequently tracked by a hypothesis test based feedback controller. If the model probability differs greatly from the real knock probability, a conventional knock controller is used; however, if the model probability is similar to the real one, the SA value obtained from the established probability limit is used to control. If the engine system behavior changes, for example, due to aging effects or varying external gas re circulation (EGR), the maps can be updated using the on-board learning algorithm. This method allows to avoid dangerous work areas operating with the optimum SA of the engine possible, but also to adapt to the aging of the engine and unfamiliar situation. In particular, due to the use of a learning algorithm, one of the most remarkable advantages of this method is that a previous calibration of the model is not necessary. Experimental validations have been carried out on a four cylinder gasoline SI engine which is equipped with in-cylinder pressure sensors.

This article is organized as follows: first, the on-board map learning based combustion control scheme is presented, where the map learning knock event algorithm is intro-

duced and the control decision layer is described. Then, the experimental set-up and validation test are presented, next section shows the experimental validation and discussion for steady and transient operation conditions. Finally, details on the high sensitivity knock detection algorithm are provided as an appendix.

2. Fuzzy logic map-based knock controller

This paper proposes an SA control constraining knock probability based on a map learning algorithm combined with a conventional controller for SA control. The structure of the control algorithm is shown in Figure 1.

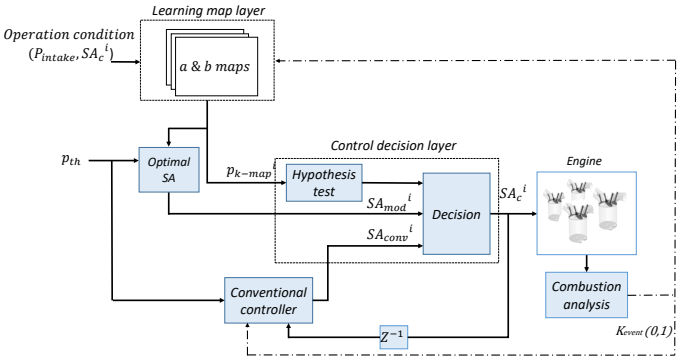

Figure 1: Fuzzy logic map-based knock controller proposed.

The proposed control framework has two main components, the learning map layer and the control decision layer. The learning map layer consists of two probability distribution maps, a and b, where the inputs are the intake manifold pressure (P_{intake}) and the SA. The output of the learning map layer is the expected knock probability at each operating condition, which is updated in real time according to the measured input signals. The model can be used to estimate the optimal SA (SA_{mod}) for a given knock probability threshold (p_{th}).

To be able to evaluate if the model is working correctly, a control decision layer is added, which consist of a hypothesis test and a decision block. The hypothesis test evaluates the probability of the most recent series of knock events, compared with the probability obtained through the model (p_{k-map}). A subsequent decision block determines the SA controller output: as long as the probability of the model approaches the real one, the controller will work with the model output (SA_{mod}), but if there is a mismatch between the expected and the observed probability, a conventional controller is used SA_{conv}. In between a combination of both, the model and a conventional controller, is used. In next sections, the learning algorithm and the hypothesis test controller are introduced in detail.

2.1. Learning map layer

On the basis that knock events are binomially distributed regardless of the probability density function of knock intensities [27], the binomial signal is obtained by comparing knock intensity metric I_r value calculated from in-cylinder pressure with a threshold value I_{th}. The methods for I_r and I_{th} calculation were presented in [28] and [29] respectively, while are briefly described in the appendix section at the end of the document. In the present work, the knock probability distribution is learned in the perspective of Bayesian learning as is shown in [25], where the conjugate prior to binomial distribution adopts beta distribution, $Beta(p \mid a, b)$, where a and b are the two parameters of the distribution. The parameter a represents knocking cycles and b no-knocking cycles, the probability that a certain cycles is a knocking cycle ($Pr\{knock = 1\}$) is denoted by p which can be considered as a stochastic variable.

The Bayesian learning process is applied to learn the knock event distribution, figure 2 (left plot) shows the calculation process, where the cross represents a knocking cycle and filled circles represents no-knocking. Consider a given time step (k) in the engine operating sequence and a receding window of 10 cycles. Point k, at the left side plot of figure 2 shows this situation when 3 cycles of the receding window have shown knock and 7 of them do not, so $a_k = 3$ and $b_k = 7$. In the right plot the corresponding beta distribution can be observed. Consider then that cycle $k+1$ shows knock, then a and b parameters are shifted to $a_{k+1} = 4$ and $b_{k+1} = 7$, which leads to a modification in the beta distribution as shown in the right plot. If cycle $k+2$ is a no-knocking cycle, then parameter $a_{k+2} = a_{k+1}$ and $b_{k+2} = 8$. In the right plot the corresponding beta distribution can be observed. The Bayesian learning process is applied to learn the distribution of the knock event map.

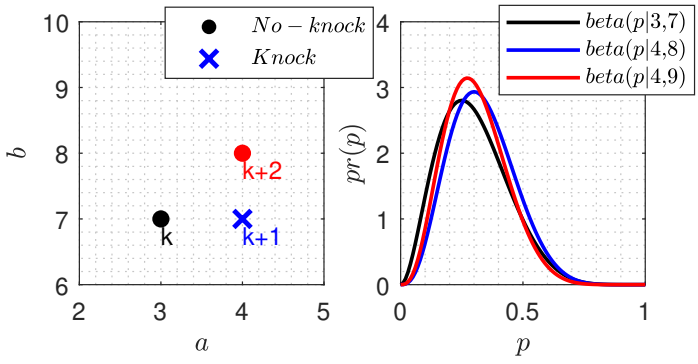

Figure 2: Left plot: a and b parameters evolution during learning process. Right plot: Beta distribution for left plot cases.

The mean and variance of the stochastic variable p can

be calculated as:

$$E[p] = \frac{a}{a+b}$$
$$Var[p] = \frac{ab}{(a+b)^2(a+b+1)} \quad (3)$$

At a constant speed, the engine operating range is discretized into two dimensional (2D) grids: pressure of the intake manifold (x_1) and spark advance (x_2), as it is shown in figure 3 left plot black points. The blue point in figure 3 left plot represents an operating condition $[x_{1,op}, x_{2,op}]$, the engine not always operates precisely on the grid points so the data of the adjacent grids must be updated following some method.

There are many adaptation methods that can be found in the literature on map lookup table estimation, for example, Bilinear interpolation, Kalman filter [30], Gaussian filter [31], among others. These last two algorithms update a wider area or even the complete map. In the present work, knock probability map is update by a Gaussian function, as:

$$\phi = \frac{1}{\sqrt{2\pi}\sigma_{x_1}} e^{-\frac{1}{2}\frac{(x_1 - x_{1,op})^2}{\sigma_{x_1}^2}} + \frac{1}{\sqrt{2\pi}\sigma_{x_2}} e^{-\frac{1}{2}\frac{(x_2 - x_{2,op})^2}{\sigma_{x_2}^2}} \quad (4)$$

where ϕ is the weight for each grid point ($X : [x_{1,op}, x_{2,op}]$), and σ_{x_1} and σ_{x_2} are parameters that determine the smoothness of map. This effect is shown in figure 3 (left plot), while the red surface represents the map for $\sigma_{x_1,x_2} = 2$ and in blue for $\sigma_{x_1,x_2} = 1$.

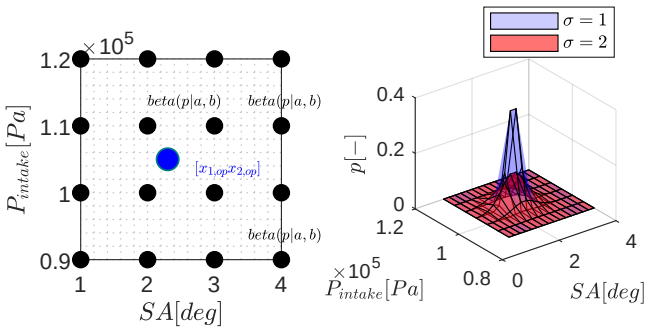

Figure 3: Proposed probability map learning algorithm for spark advance control.

The two parameters a and b of beta distribution are updated cycle-by-cycle as:

$$a_{i+1} = a_i + \phi \qquad if \quad knock$$
$$b_{i+1} = b_i + \phi \qquad otherwise \quad (5)$$

As parameters a and b grow, the difference between the anterior and posterior distributions decreases, so it is necessary to normalize these parameters as follows [25]:

$$a + b = N \quad (6)$$

where N is the normalization size, if N is small, the effect of new data on learning is fast. Otherwise, the effect of new data is slow. In this study a value of 200 is used as normalization size.

Then, once the parameters a and b have been obtained, the knock probability on a grid point and its variation can be calculated by equation (4). To avoid the model over estimates the probability at points near to the operating one, the σ value in each of the axes should not be very high, for this work the value is $\sigma_{x1,x2} = ([2, 1])$.

Another aspect to take into account is the fact that the probability at the same operation point for increasing values of SA is never decreasing [32] since increasing SA cannot lead to lower knock probabilities. This is taken into account to obtain the limit value of SA for the corresponding point of operation, but not to update a and b.

2.2. Control decision layer

As shown in the figure 1, the decision layer is composed of a hypothesis test, which compare the real knock probability with the model knock probability, and a decision block, which determines the SA used as controller output, based on the combination of the model SA and conventional SA controllers previously explained.

2.2.1. Hypothesis test

The statistical controller based on hypothesis tests is used to evaluate how the probability of the model adjusts to real knock probability. As the knock process can be assumed as a binomial distribution, for a given knock probability (p_k), the probability of observing k knock events in n cycles can be calculated as [33]:

$$P_n^k = \binom{a}{b} p_k^k (1 - p_k)^{(n-k)} \quad (7)$$

The likelihood L_h ratio is widely used in knock control [33, 34], to evaluate the ratio between the expected knock probability (p_0), in the case at hand obtained from the knock probability learning algorithm, and the observed knock probability (p_{obs}) obtained from the knock rate ($p_{obs} = k/n$), which maximizes the Likelihood. L_h can be expressed as:

$$L_h = \frac{L(p_0 \mid N_k)}{L(p_{obs} \mid N_k)} = \frac{p_0^k (1 - p_0)^{(n-k)}}{p_{obs}^k (1 - p_{obs})^{(n-k)}} \quad (8)$$

where N_k is the observed knock event sequence, k is the number of knock events in N_k, n is the number of engine

cycles. In figure 4 a plot of the likelihood ratio as function of the cycle number is shown for a knock rate $p_0 = 0.1$.

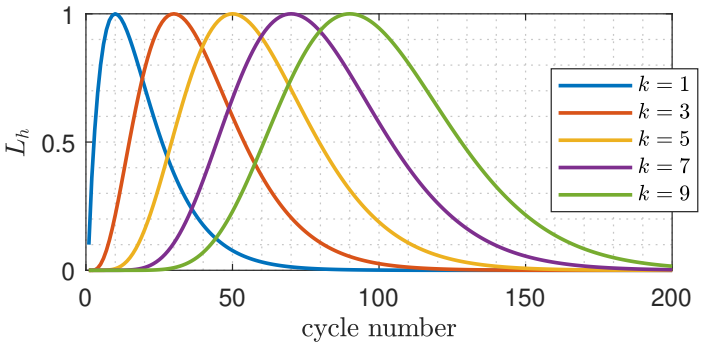

Figure 4: Likelihood ratio for different number of knock events ($p_0 = 0.1$).

As expected, the likelihood ratio peaks when the observations match the assumed knock rate p_0, i.e, at $n = 50$ for one knock and five knock events $k = 5$. A likelihood ratio of 1, indicates that p_0 perfectly match the knock rate p_{obs}, while likelihood ratios near zero suggest a mismatch.

To be able to calculated the knock probability for the most recent series of knock events, Last In First Out (LIFO) buffers were used to store the knock event data in a fixed size array. For this controller, two buffers were employed, one of 100 cycles representative of the operation point, and an emergency buffer of 5 cycles in order to detect consecutive knocking cycles. The probability of each buffer can be calculated as:

$$p_{buffer} = \frac{\sum_{j=1}^{N_{buffer}} k_j}{N_{buffer}} \tag{9}$$

where k_j is the boolean knock signal, 1 for knocking cycles and 0 for a no-knocking cycle, and N_{buffer} is the length of the buffer.

Accordingly, the likelihood ratio defined in (8) is computed for both buffers as:

$$L_h = \frac{p_{k-map}^k (1 - p_{k-map})^{(n-k)}}{p_{buffer}^k (1 - p_{buffer})^{(n-k)}} \tag{10}$$

where p_{k-map} represents the probability obtain in the map and p_{buffer} is the estimate real probability over the buffers.

2.2.2. Decision block

In this block a combination between the SA of the model and a conventional controller is performed: if the probability of the maps p_{k-map} match with the probability of the buffer p_{buffer}, the SA of the controller it will be the one obtain in the maps, otherwise, the SA it will be controlled by a conventional controller. Transition between model and conventional SA is managed as follows:

$$SA_c^i = (1 - \alpha_c) SA_{conv} + \alpha_c SA_{mod} \tag{11}$$

where α_c will be 0 when the model mismatch with the estimated knock probability and 1 when the model matches.

From the hypothesis test two likelihood ratios are obtained, one from the 100 cycles buffer L_{hL} and the other one from the short buffer L_{hS}, which will be used to determine α_c as is shown in figure 5.

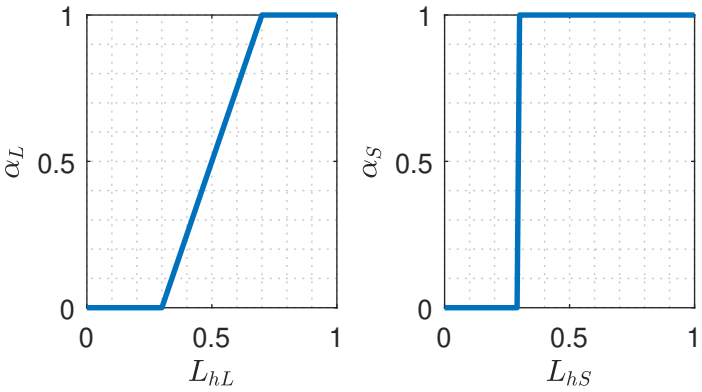

Figure 5: Alpha controller for different likelihood ratio values.

The value α_c set in the controller will be the product between α_L and α_S. Note that a value of L_{hL} or L_{hS} of 0.3 leads to an alpha value of 0 which means a complete use of the conventional controller, while a Ł$_{hL}$ above 0.7 means the a complete use of the model.

3. Experimental set-up

A four-stroke SI engine was used to validate the proposed knock controller. The specifications of the engine are shown in Table 1.

Table 1: Engine specifications

Displaced volume	1300 cc
Stroke	81.2 mm
Bore	72 mm
Compression ratio	10.6:1
Combustion	SI

The engine was coupled with a dynamometer, and in-cylinder, intake manifold and exhaust manifold pressure sensors were installed. The control system of the engine includes an electronic control unit (ECU), which has been bypassed with an ETAS ES910 system for modifying the standard calibration. A prototyping system from National Instruments TM was used for acquisition, control, and diagnosis purposes.

The system consists of:

- A PXI 6123 and PXI-6251 acquisition modules were used to acquire information from sensors with high sampling rate and 16 bit resolution, e.g. in-cylinder pressure sensor.
- A NI-9759 module was used with a PXI 7813R for programming in FPGA the full-pass control.
- A CAN interface module, PXI-8513, was used to communicate with the ES910 system.
- A PXI 8110 controller was used to compute the control strategy and to process, store and analyze all the information from the previous modules.

The variable geometry turbine (VGT) position was directly controlled with the FPGA by full-pass while the throttle valve position, the VVT, the spark advance, and the injected fuel were controlled through ECU by-pass. An optical encoder was used to provide the crank angle reference for the in-cylinder pressure collection with a sampling accuracy of 0.2 crank angle degree.

During the experiments, the engine was tested at several operating conditions by keeping the speed constant, and two kind of tests for validation proposes were performed:

- Steady test: 7 steady points were analyzed at 2000 RPM, 90000, 95000 , 100000, 105000, 110000, 115000 and 120000 Pa of intake pressure.
- Transient tests: The controller was also validated with step throttle angle command. In figure 6 the throttle angle command and the corresponding manifold pressure are shown.

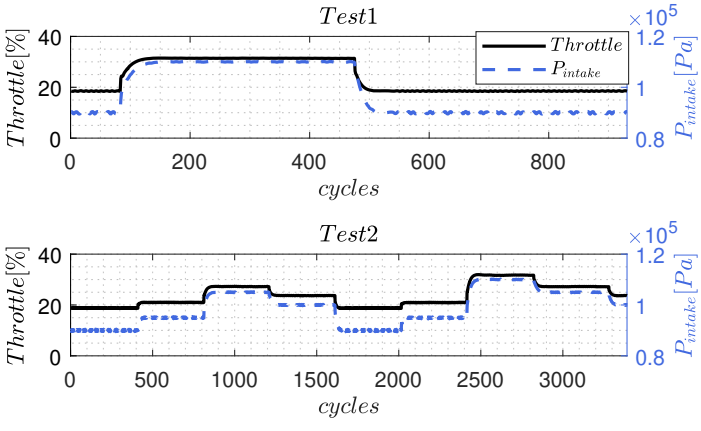

Figure 6: Experimental tests: Transient operation conditions.

The controller performance is compared with a conventional SA controller in both, steady and transient test.

4. Validation results and discussion

4.1. Learning

The SA output controller is defined by equation (11) , during the learning process $\alpha_c = 0$, so the controller will be totally conventional. When the knock probability map is updated, the value of α_c is defined as function of the likelihood ratio from equation (10).

Different variables are plotted during the learning process in figure 7, in the top plot, the SA controller output is shown, in the central plot the probability of the map evolution and the probability threshold are plotted, and in the bottom plot the value of α_c is shown. The knock probability threshold set in both controllers is 10% according to knock definition described in [29], which is equivalent to a 2 % of MAPO threshold at 40000 Pa.

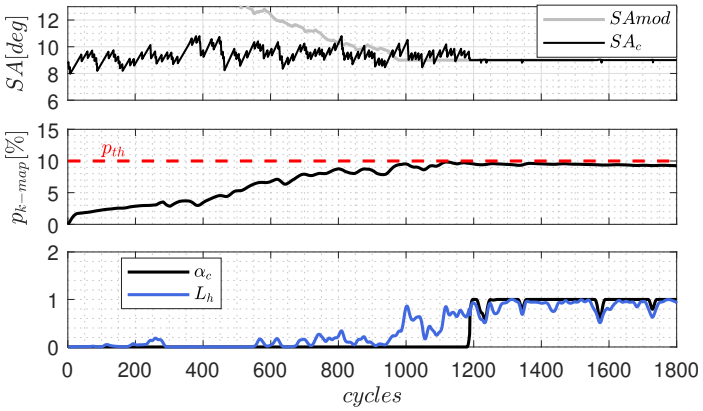

Figure 7: Spark advance controller output during learning process. Top plot: Spark advance output controller. Middle plot: knock probability evolution of the model. Bottom plot: Likelihood ratio (100 cycles buffer) and alpha controller value.

Note that while the map is being updated, the model knock probability at the operation point is reaching the control value set p_{th}. Once the probability map is updated, and the model sufficiently describes the knock probability function, the SA output controller responds to a combination between the one obtained on the map and the conventional controller according α_c value.

As can be seen in figure 7 bottom plot, the number of cycles required to filled the model is 1200 at 2000 RPM, which represents around 80 seconds.

4.2. Steady state results

In order to compare the performance of the controller in steady conditions, 300 cycles of two different operation points are analyzed once the maps are updated. Figure 9 shows the response of the controller proposed and conventional SA controller for 2000 RPM at 105000 Pa and 90000 Pa of intake manifold pressure with a K_{ret} of 0.5 crank angle degree (CAD) for the conventional controller.

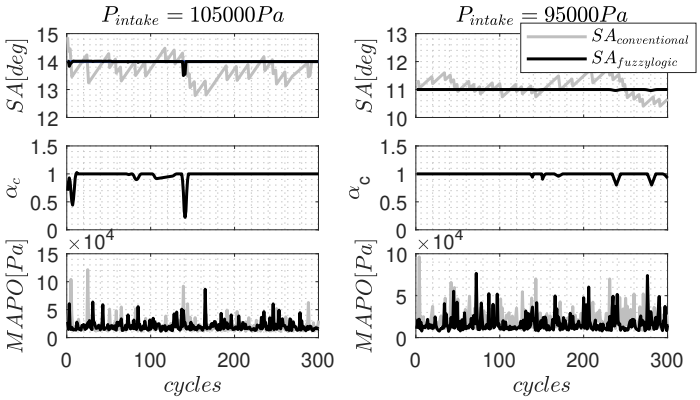

Figure 8: Control performance at different intake pressure. Left plot: 105000 Pa. Right plot 95000 Pa. Top plot: Spark advance command. Middle plot α_c value. Bottom plot MAPO value.

Note that in left plot of figure 8 around cycle 140, the SA out of the controller proposed is delayed 0.5 CAD, this is because consecutive knock cycles have been observed making α_c bellow one.

For each test the mean SA ($\overline{SA}$), its standard deviation, the mean indicated mean effective pressure ($\overline{IMEP}$), its standard deviation, maximum MAPO ($M\hat{A}PO$), and knock probability for MAPO (p_{k-M}) and the low knocking cycles indicator (p_{k-I}) over 300 cycles were calculated. All the values are shown in Table 2.

Table 2: Control performance at different intake pressure: 105000 Pa and 95000 Pa

	Conventional		Fuzzy logic	
$P_{int}[Pa]$	105000	95000	105000	95000
$K_{ret}[deg]$	0.5	0.5	0.5	0.5
σ_{SA}	0.34	0.31	0.1	0.006
$\overline{SA}[deg]$	13.77	10.86	13.86	11.07
σ_{IMEP}	0.13	0.12	0.12	0.09
$\overline{IMEP}[Pa]$	9180000	1040000	940000	1066000
$M\hat{A}PO[Pa]$	120000	96000	86000	70000
$p_{k-M}[\%]$	3.98	2.72	1.99	1.7
$p_{k-I}[\%]$	10.6	10.4	10.1	9.97

Analyzing table 2, when comparing both controllers at the same intake pressure, for 105000 Pa a similar average value of SA is found, which is consistent with the fact that the final knocking events are quite similar. However, the variability of the SA is more than the double for the conventional knock controller. That higher variability makes the knock controller modify the SA to more advanced values leading to dangerous knocking events. In 95000 Pa case, the average value of SA for the controller proposed is higher than for a conventional controller, and the variability of the SA is more than three times larger what is reflected in the IMEP standard deviation.

A summary of seven points steady test performed is shown in figure 9, where empty circles represents test performed with a conventional knock controller and filled circles with the proposed controller, while the color represents the operating point. Both controllers were set at 10% knock probability threshold, based on knock detection presented in [29], and K_{ret} of 0.5 CAD. The maximum MAPO over 500 cycles is represented in function of the thermal efficiency η_{th} computed as:

$$\eta_{th} = \frac{\oint p\, dV}{m_f q_{LHV}} \tag{12}$$

m_f represents the fuel amount injected in the cylinder during one combustion cycle, q_{LHV} is the lower heating power of the fuel (45 MJ/kg for gasoline), p is the in-cylinder pressure, dV is the rate of cylinder volume change.

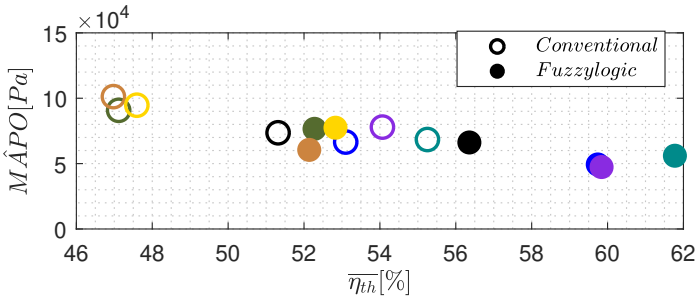

Figure 9: MAPO comparison for 7 operating points at steady state condition.

As can be seen in figure 9, while comparing the same color dots, the proposed controller shows an improvement in steady state due to a higher average of η_{th} and a lower maximum MAPO, thus allowing to obtain higher efficiency avoiding high knocking cycles.

4.3. Transient state results

The controller was also validated in transient operation condition, as is shown in figure 6. Test 1 results are plotted in figure 10, in the first plot starting from above, the SA controller output is shown with a knock probability threshold of 10% for both controllers, in the second plot the IMEP evolution is plotted, and in the third plot MAPO is shown.

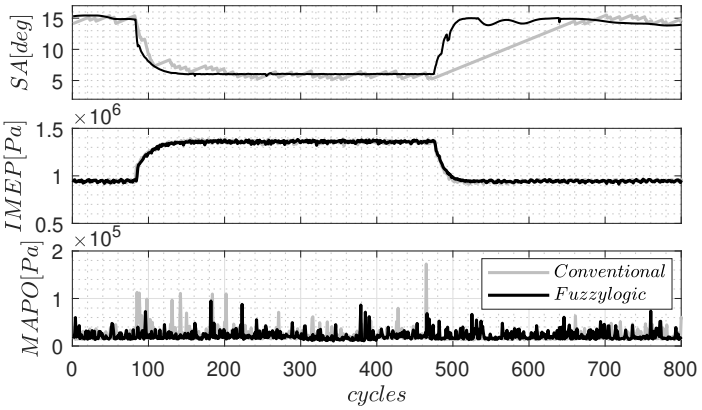

Figure 10: Control performance of Test 1. Top plot: Spark advance. Middle plot: IMEP. Bottom plot: MAPO

As can be seen in figure 10 for the conventional controller for the first throttle step, around cycle 100, several cycles with MAPO above 100000 Pa can be found, and this is because the controller delays the spark more progressively. While in the second throttle step, around cycle 500, the spark takes longer to reach the optimum point, which suggests a loss of performance at that point. Traditional knock controller is generally applied with a lookup table, which would avoid this problem, but these open loop tables are off line pre calibrated for a given boundary condition, which can be modified through the real operation condition.

Test 2 results are summarized in table 3, where it shows the mean SA, mean IMEP, maximum MAPO, percentage of cycles with MAPO over 40000 Pa (p_{k-M}), percentage of low knocking cycles (p_{k-I}), and the thermal efficiency for both controllers.

Table 3: Control performance: Test 1 & 2.

	Conventional		Fuzzy logic	
$Test$	1	2	1	2
$\overline{SA}$	10.6	9.9	10.95	10.53
$\overline{IMEP}$	1080000	1110000	1118000	1140000
$M\hat{A}PO$	172000	14000	86000	90000
$p_{k-M}[\%]$	6	4	2.4	2
$p_{k-I}[\%]$	17.8	15.6	10.3	10.2
$\eta_{th}[\%]$	38.98	38.7	40.69	40.4

As can be seen in the table 3, the proposed controller reaches a higher average IMEP value with a MAPO max value lower than the conventional controller, this is reflected in an improvement of almost 2 % of the thermal efficiency for both transient tests.

Comparing with recent adaptive strategies for knock control [16], the main contribution of the proposed method is that the use of an indicator able to detect low-knocking cycles provides more information about the knock nature and also allows to use higher knocking rates, then reducing the number of cycles needed to estimate a given knock probability. Compared to recent methods of model-based controller such as [25], this method does not require a previous calibration of the maps, being able to adapt to the operating and contour conditions in a more realistic way, and being able to adapt the model 10 times faster.

5. Conclusions

This article presents a knock probability constrained spark advance control based on the on-board map adaptation of knock event distribution, applying a low-knocking cycles detection method to update the model. The proposed control framework is experimentally validated in both steady and transient state experiments, and compared with a conventional controller. The following conclusions can be drawn from this study:

- The proposed controller achieved an improvement between 5-7% of the thermal efficiency in steady state conditions, and maintain MAPO at safe levels of operation, compared with a conventional knock controller.
- This method allows an on-board calibration no need for prior system calibration, which is a great advantage for different boundary operating conditions.
- Controlling the ignition timing with the method presented offers less spark advance variability, making it more stable.
- Applying a low-knocking cycles allows updating the model faster, obtaining more information about the system to be controlled, and quickly adapt the model to changes in the operation.

6. Appendix: Knock detection method

Working at high knock rates allows to shorten the time required to identify the stochastic properties of knock and, therefore, to update the detonation models. Hence, a low-knocking cycle detection method is used in this work.

A knock indicator which take into account the intensity of the oscillations of the in cylinder and its angular evolution is used. The minimum oscillation required for the end gas auto ignition detection is based on the constant volume combustion of the fuel mass that is not burnt by the SI flame development.
The resonance indicator presented in [28] is used.

$$I_r(\theta) = \sum_{\theta=\theta_1}^{\theta=\theta_2} w(\theta-\theta_1) p_{hp}(\theta) e^{-2\pi \sum_{\psi=0}^{\psi=\theta} \frac{B\sqrt{\gamma(\psi)p(\psi)V(\psi)}}{\pi D\sqrt{m}} T(\theta)} \tag{13}$$

where θ_1 and θ_2 define the interval where the resonance analysis is performed, w is a window function of $\theta_1 - \theta_2$ length, and $T(\theta)$ is the instantaneous sampling rate, which is constant only in time-based acquisition or if the instantaneous engine speed fluctuations are negligible, B is the Bessel constant, D is the bore of the cylinder, V the chamber volume, m the trapped mass and p the in-cylinder pressure.

The minimum oscillation required for the end-gas to auto ignite is calculated as [29]:

$$I_{rmin} = G\frac{(\gamma - 1)}{V} m_f q_{LHV}(1 - MFB) \qquad (14)$$

where G is the transference constant of excitation of the resonance modes due to the rise of pressure because of the auto ignition, MFB the mass fraction burn, m_f the fuel mass, where q_{LHV} is the lower heating value of the fuel (45 MJ/kg for gasoline).

If the maximum resonance indicator I_r is greater than the expected resonance at the angle where the indicator is maximum knock is detected:

$$if max(I_r) >= I_{rmin}(\theta_{Irmax}) knock \qquad (15)$$

In [29] was demonstrated that a 10 % of this definition it is equivalent to a 2 % according to MAPO definition with a threshold at 40000 Pa.

7. Acknowledgments

Irina A. Jimenez received a funding through the grant 132GRISOLIAP/2018/132 from the Generalitat Valenciana and the European Social Fund.

References

[1] H. Liu, C. Wang, Y. Yu, H. Xu, X. Ma, An experimental study on particle evolution in the exhaust gas of a direct injection si engine, Applied Energy 260 (2020) 114220.

[2] C. Chen, P. Pal, M. Ameen, D. Feng, H. Wei, Large-eddy simulation study on cycle-to-cycle variation of knocking combustion in a spark-ignition engine, Applied Energy 261 (2020) 114447.

[3] X. Zhen, Y. Wang, S. Xu, Y. Zhu, C. Tao, T. Xu, M. Song, The engine knock analysis–an overview, Applied Energy 92 (2012) 628–636.

[4] J. B. Heywood, Combustion engine fundamentals, 1ª Edição. Estados Unidos (1988).

[5] X. Shen, T. Shen, Real-time statistical learning-based stochastic knock limit control for spark-ignition engines, Applied Thermal Engineering 127 (2017) 1518–1529.

[6] E. Pipitone, A comparison between combustion phase indicators for optimal spark timing, Journal of Engineering for Gas Turbines and Power 130 (5) (2008) 052808.

[7] E. Corti, C. Forte, G. Mancini, D. Moro, Automatic combustion phase calibration with extremum seeking approach, Journal of Engineering for Gas Turbines and Power 136 (9) (2014) 091402.

[8] T. Shen, M. Kang, J. Gao, J. Zhang, Y. Wu, Challenges and solutions in automotive powertrain systems, Journal of Control and Decision 5 (1) (2018) 61–93.

[9] G. Shu, J. Pan, H. Wei, Analysis of onset and severity of knock in si engine based on in-cylinder pressure oscillations, Applied Thermal Engineering 51 (1-2) (2013) 1297–1306.

[10] J. D. Naber, J. R. Blough, D. Frankowski, M. Goble, J. E. Szpytman, Analysis of combustion knock metrics in spark-ignition engines, SAE Transactions (2006) 223–243.

[11] P. Bares, D. Selmanaj, C. Guardiola, C. Onder, Knock probability estimation through an in-cylinder temperature model with exogenous noise, Mechanical Systems and Signal Processing 98 (2018) 756–769.

[12] J. C. P. Jones, J. Frey, S. Shayestehmanesh, Stochastic simulation and performance analysis of classical knock control algorithms, IEEE Transactions on Control Systems Technology 25 (4) (2016) 1307–1317.

[13] U. Kiencke, L. Nielsen, Automotive control systems: for engine, driveline, and vehicle (2000).

[14] J. P. Jones, J. Frey, K. Muske, D. Scholl, A cumulative-summation-based stochastic knock controller, Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering 224 (7) (2010) 969–983.

[15] J. C. P. Jones, J. M. Spelina, J. Frey, An optimal cumsum-based knock controller, IFAC Proceedings Volumes 46 (21) (2013) 372–377.

[16] J. C. P. Jones, S. Shayestehmanesh, J. Frey, A bayesian knock event controller, IEEE Transactions on Control Systems Technology (2019).

[17] J. C. Peyton Jones, S. Shayestehmanesh, J. Frey, A dual-threshold knock controller, International Journal of Engine Research 18 (8) (2017) 837–846.

[18] S. Shayestehmanesh, J. C. Peyton Jones, J. Frey, Computing the closed-loop characteristics of a generalized multi-threshold knock controller, International Journal of Engine Research 19 (9) (2018) 952–962.

[19] K. Zhao, Y. Wu, T. Shen, Stochastic knock control with beta distribution learning for gasoline engines, IFAC-PapersOnLine 51 (31) (2018) 125–130.

[20] A. Thomasson, H. Shi, T. Lindell, L. Eriksson, T. Shen, J. C. P. Jones, Experimental validation of a likelihood-based stochastic knock controller, IEEE Transactions on Control Systems Technology 24 (4) (2015) 1407–1418.

[21] D. Selmanaj, G. Panzani, S. Van Dooren, J. Rosgren, C. Onder, Adaptive and unconventional strategies for engine knock control, IEEE Transactions on Control Systems Technology 27 (4) (2018) 1838–1845.

[22] J. Livengood, W. PC, et al., Correlation of autoignition phenomena in internal combustion engines and rapid compression machines (1955).

[23] E. Corti, C. Forte, A statistical approach to spark advance mapping, Journal of Engineering for Gas Turbines and Power 132 (8) (2010).

[24] Y. Zhang, X. Shen, T. Shen, A survey on online learning and optimization for spark advance control of si engines, Science China Information Sciences 61 (7) (2018) 70201.

[25] Y. Zhang, X. Shen, Y. Wu, T. Shen, On-board knock probability map learning–based spark advance control for combustion engines, International Journal of Engine Research 20 (10) (2019) 1073–1088.

[26] J. C. P. Jones, J. Frey, Threshold optimization and performance evaluation of a classical knock controller, SAE International Journal of Engines 8 (3) (2015) 1021–1028.

[27] J. Frey, J. C. P. Jones, S. Shayestehmanesh, Stochastic simulation of a cumsum knock controller, IFAC-PapersOnLine 49 (11) (2016) 210–216.

[28] C. Guardiola, B. Pla, P. Bares, A. Barbier, An analysis of the in-cylinder pressure resonance excitation in internal combustion engines, Applied Energy 228 (2018) 1272–1279.

[29] B. Pla, J. De La Morena, P. Bares, I. Jiménez, Knock analysis in the crank angle domain for low-knocking cicles detection, Tech. rep., SAE Technical Paper (2020).

[30] C. Guardiola, B. Pla, D. Blanco-Rodriguez, L. Eriksson, A computationally efficient kalman filter based estimator for updating

look-up tables applied to nox estimation in diesel engines, Control Engineering Practice 21 (11) (2013) 1455–1468.
[31] J. Gao, Y. Zhang, T. Shen, An on-board calibration scheme for map-based combustion phase control of spark-ignition engines, IEEE/ASME Transactions on Mechatronics 22 (4) (2017) 1485–1496.
[32] N. Cavina, A. Brusa, N. Rojo, E. Corti, Statistical analysis of knock intensity probability distribution and development of 0-d predictive knock model for a si tc engine, Tech. rep., SAE Technical Paper (2018).
[33] J. C. P. Jones, J. M. Spelina, J. Frey, Likelihood-based control of engine knock, IEEE Transactions on Control Systems Technology 21 (6) (2013) 2169–2180.
[34] J. M. Spelina, J. C. Peyton Jones, J. Frey, Stochastic simulation and analysis of a classical knock controller, International Journal of Engine Research 16 (3) (2015) 461–473.