SCIENCE CHINA
Information Sciences

• **REVIEW** •

July 2018, Vol. 61 070201:1–070201:21
https://doi.org/10.1007/s11432-017-9377-7

Special Focus on Learning and Real-time Optimization of Automotive Powertrain Systems

A survey on online learning and optimization for spark advance control of SI engines

Yahui ZHANG, Xun SHEN* & Tielong SHEN

Department of Engineering and Applied Sciences, Sophia University, Tokyo 102-8554*, Japan*

Received 11 December 2017/Accepted 31 January 2018/Published online 12 June 2018

Abstract One of the most important factors affecting fuel efficiency and emissions of automotive engines is combustion quality that is usually controlled by managing spark advance (SA) in spark ignition (SI) engines. With increasing sensing capabilities and enhancements in on-board computation capability, online learning and optimization techniques have been the subject of significant research interest. This article surveys the literature of learning and optimization algorithms with applications to combustion quality optimization and control of SI engines. In particular, this paper reviews extremum seeking control algorithms for iterative solution of online optimization problems, stochastic threshold control algorithms for iterative solution of probability control of stochastic knock event, as well as feedforward learning algorithms for iterative solution of operating-point-dependent feedforward adaptation problems. Finally, two experimental case studies including knock probabilistic constrained optimal combustion control and on-board map learning-based combustion control are carried out on an SI gasoline engine.

Keywords online learning, stochastic optimization, iterative solution, combustion control, spark ignition engine

Citation Zhang Y H, Shen X, Shen T L. A survey on online learning and optimization for spark advance control of SI engines. Sci China Inf Sci, 2018, 61(7): 070201, https://doi.org/10.1007/s11432-017-9377-7

1 Introduction

Today's growing concern about the fuel crisis and requirement for low emissions drive development of higher efficiency automotive engines and advanced engine control technologies. As known, one of the most significant factors influencing fuel efficiency and emissions of engines is combustion quality that includes combustion phase [1], knock intensity (KI) or probability [2], cyclic variability (CV) [3], etc. Combustion quality of spark ignition (SI) engine is controlled by managing the spark advance (SA) which is the spark timing determining the start of combustion.

Current on-board electronic control units (ECUs) usually control the combustion quality in an open-loop manner based on SA maps that require a pre-calibration process performed on the test bench [4]. However, this system has several disadvantages: the calibration process is labor-consuming thus it is impossible to be carried out for each individual engine and cannot explore every operating conditions; for the sake of security, maps are calibrated with enough margins, conservative maps cannot guarantee the best SA; the optimal value may drift due to many factors, such as the component aging, fuel quality, and environmental disturbances [1]. These statements imply that the open-loop control system has potential for performance improvement of each individual engine.

* Corresponding author (email: shenxun@eagle.sophia.ac.jp)

Fortunately, increased sensing capabilities of modern combustion engines provide large amount of information, hence, advanced combustion control systems may be available. Recently, in-cylinder pressure (ICP)-based combustion control has attracted wide research interests in the automotive industry, due to its potential to get extremely valuable information of in-cylinder engine behavior [5]. With increased sensing capabilities and enhancements in ECUs computation capability, online learning and optimization techniques have been the subject of significant research interest in combustion control [4, 6–9].

Online learning and optimization techniques in combustion control have several potential advantages over the open-loop control system. If included in an calibration process that is usually labor-consuming, online learning and optimization algorithms would allow speeding up the process [4]. They allow to set the best (or optimal) SA that is usually referred to as the maximum brake torque (MBT) SA, i.e., the maximum engine efficiency SA. Moreover, combustion control is usually subject to physical constraints of abnormal combustion such as knock probability (or knock intensity) and cyclic variability [2, 10]. Severe knock damages the cylinder and decreases engine efficiency, while large cyclic variability decreases engine efficiency and driving comfort of vehicles. Hence, physical constraints are required to determine a feasible region of SA and avoid the abnormal combustion. However, at many operating points the maximum brake torque SA may be out of the feasible region. The optimal SA in the feasible region usually locates on the boundary of physical constraints [2, 10]. Note that the abnormal combustion has a great deal of randomness and is affected by many factors, such as ambient pressure and temperature, and fuel quality. These factors result in the varying boundary of physical constraints. The on-board implementation of online learning and optimization algorithms would allow a real-time handling of the varying stochastic boundary for every operation condition [11].

This article surveys the literature of online learning and optimization algorithms with applications to combustion quality optimization and control of SI engines. Combustion quality optimization and control problems are generalized in a whole framework that includes online optimization of combustion phase, probability control on boundary of physical constraints, and operating-point-dependent feedforward online adaptation. This article first considers extremum seeking (ES) control algorithms for iteration solution of online optimization problems in spark advance optimization applications. Three classes of ES are successively reviewed: stochastic approximation-based ES [7, 12–20], sinusoid-based ES [21–35], and natural perturbation-based ES [36–39]. This article then focuses on stochastic threshold control algorithms for iterative solution of probability control problems in knock limit control applications [2, 10, 11, 40–45]. Two state-of-the-art stochastic limit control algorithms are reviewed: likelihood-based algorithm and statistical learning-based algorithm. This article finally deals with feedforward learning algorithms for iterative solution of operating-point-dependent feedforward adaptation problems in applications of spark advance map online adaptation [46–49]. At last, two experimental case studies including knock probabilistic constrained optimal combustion control and on-board map learning-based combustion control are carried out on an SI gasoline engine.

This article is organized as follows. Section 2 generalizes the problem formulation and framework of combustion quality optimization and control. In Sections 3–5, related learning and optimization methods under the general framework are reviewed: extremum seeking control algorithms for online optimization problem, stochastic threshold control algorithms for probability control problems, and feedforward online learning algorithm. Section 6 presents application related contents with two experiment case studies. Section 7 concludes this article.

2 Problem formulation

Statistic and stochastic properties of SA-CA50 (CA50 is crank angle of 50% burnt after top dead center (ATDC)), CA50-η (η is the thermal efficiency), and CA50-KI, where the distributions are constructed from 17000 cycle data, are shown in Figure 1. The data is collected on a 3.5-litre V-type SI gasoline engine, which runs at steady operating points: SA from 14° to 30° before top dead center (BTDC), throttle angle at 7°, engine speed at 1200 rpm. This experimental result in Figure 1(a) motivates the

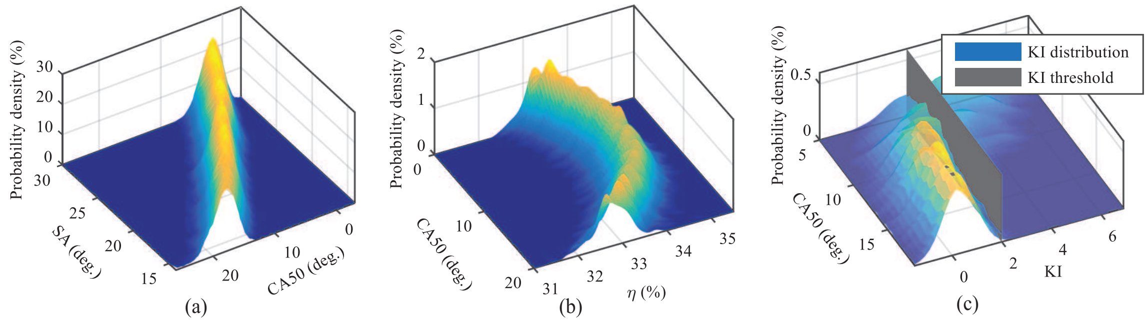

Figure 1 (Color online) Statistical analysis of experiment data. (a) SA-CA50 statistical distribution; (b) CA50-η statistical distribution; (c) CA50-KI statistical distribution and KI threshold.

following statistical representation of the causality SA-CA50.

$$\mathrm{CA50} = a \cdot \mathrm{SA} + b + e, \quad e \sim N(0, \delta^2), \tag{1}$$

where a, b are constants and e obeys the Gaussian distribution, i.e., from statistical point of views, for a given SA, the CA50 exhibits a Gaussian distribution with a mean $a{\cdot}\mathrm{SA}{+}b$ [50].

The stochasticity can also be observed in the relationship between CA50 and efficiency η, which is plotted in Figure 1(b). This property is represented statistically as a quadratic regression. The following statistical model is introduced to represent the causality CA50-η, which is a parabola coupled with random noise.

$$\eta = \alpha_0 \cdot \mathrm{CA50}^2 + \beta_0 \cdot + \gamma_0 + \omega, \quad \omega \sim N(0, \sigma^2). \tag{2}$$

Combustion phase online optimization problem can be described as follows:

$$\mathrm{SA}^* = \arg\min_{\mathrm{SA}} -\mathrm{E}\{\eta[\mathrm{CA50}(\mathrm{SA}, e), \omega]\}, \tag{3}$$

where $\mathrm{E}\{\cdot\}$ denotes the expected value.

For the relationship between CA50 and knock intensity (KI), Figure 1(c) shows a smooth trend of increasing averaged mean KI and variance of KI as CA50 is advanced. The grey vertical plane denotes the KI threshold. The knock event is thought to occur, i.e., knock = 1, when cyclic KI goes beyond the threshold. Otherwise, knock = 0 when cyclic KI is within the threshold.

Knock stochastic threshold control problem can be formulated as finding the spark timing SA^* such that

$$\Pr\{\mathrm{knock} = 1\} \to p_0, \quad \text{as} \quad k \to \infty, \tag{4}$$

where $\Pr\{\cdot\}$ denotes the probability of a stochastic event, p_0 is the desired target knock probability, and k represents the number of combustion cycles.

Moreover, in engine combustion control applications, the plant models (1) and (2) have short-term variations induced by changes in operating point and long-term drift caused by component aging or changes in the environment. To show the operating-point-dependent variations of the model (1), experiments have been carried out at different intake manifold pressure values (0.6 and 0.8 bar) and different engine speeds (800, 1200, 1600, 2000, 2400, 2800, and 3200 rpm), as shown in Figure 2. It is obvious that operating points significantly affect SA-CA50 relationship and hence affect the optimal spark timing in (3). SA feedforward online adaptation (or learning) is the state-of-the-art real-time combustion control technique.

Feedforward online adaptation problem can be formulated as learning the feedforward model Φ_{FF} such that

$$\mathrm{SA}_{\mathrm{FF}} \to \mathrm{SA}^* \quad \text{as} \quad k \to \infty, \quad \text{at each steady operating point}, \tag{5}$$

where $\mathrm{SA}_{\mathrm{FF}}$ is the feedforward provided by the learned feedforward model Φ_{FF}.

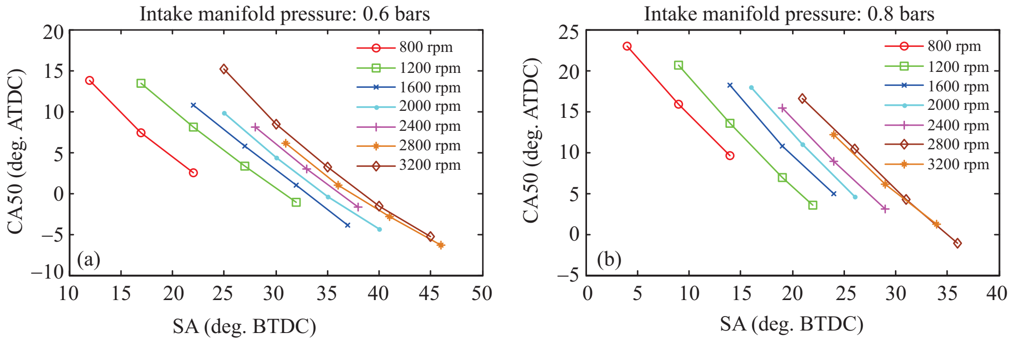

Figure 2 (Color online) Operating-point-dependent effect on SA-CA50. (a) Case 1: $p_m = 0.6$ bar; (b) case 2: $p_m = 0.8$ bar.

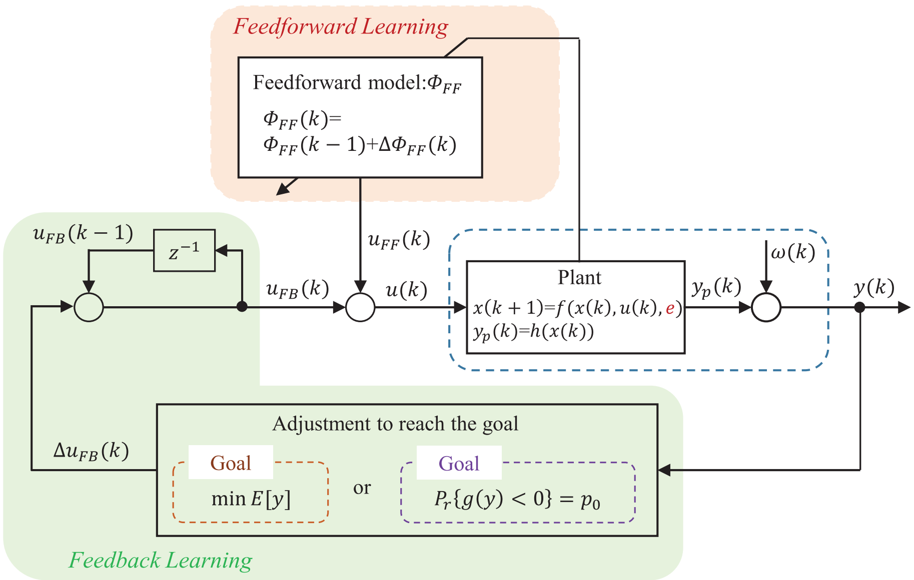

Figure 3 (Color online) General framework of online learning and optimization in combustion control applications.

SA online optimization problem, stochastic knock probability control problem, and SA feedforward adaptation problem are generalized in an online learning and optimization framework, as shown in Figure 3. A uniform iteration solution form for the online learning and optimization framework is written as

$$u(k) = u(k-1) + \Delta u(k). \tag{6}$$

Extremum seeking control algorithms for online optimization problems, stochastic threshold control algorithms for probability control of stochastic knock event, as well as feedforward learning algorithms for operating-point-dependent feedforward adaptation problems share the same iteration solution form. These online learning and optimization algorithms are then reviewed in the following sections.

3 Extremum seeking methods

Extremum seeking (ES) is a control system which is used to determine and to maintain the extremum value of a function. The steady-state input-output characteristic is optimized by ES when knowledge of the input-output is limited. Several variants of ES have been developed and have proven to be robust and efficient in many different applications. These applications can be mainly classified into two categories: ES of static map and ES of dynamical system. ES in combustion control application is essentially an

ES of static map. Hence, this section mainly reviews several ES methods for static map: stochastic approximation-based ES, sinusoid-based ES, and natural perturbation-based ES.

3.1 Stochastic approximation-based extremum seeking

We introduce some concepts and notations to generalize the problem of maximizing or minimizing a random objective function. Let $U \in \mathbb{R}^m$ and u be the set of all potential decisions and a specific decision, respectively. We would like to seek a decision that minimizes a cost function f. ω denotes randomness that is obtained only after the decision is made. Let $F(u,\omega)$ be the random objective function. Since we cannot directly optimize $F(u,\omega)$, we instead minimize the expected value, $\mathrm{E}[F(x,\xi)]$. The general problem of ES of a static map becomes

$$y^* = \min_{u\in U}\{f(u) = \mathrm{E}[F(u,\omega)]\}. \tag{7}$$

Let u^* be the optimum, $f(u^*) = y^*$. Note that both U and $F(u,\omega)$ are assumed to be convex.

Gradient descent is a classic method for the formulated problem. To seek the local minimizer, a decision u_k is updated recursively by moving in the direction opposite the gradient.

$$u_{k+1} = u_k - a_k g(u_k), \tag{8}$$

where $g(u_k)$ represents the gradient. However since direct measurements of the gradient are usually unavailable, the recursive form takes the approximated gradient, $u_{k+1} = u_k - a_k\hat{g}(u_k)$, where $\hat{g}(u_k)$ is the approximated gradient determined by specific algorithms using noisy measurements [12]. Hence, the main consideration for the gradient descent form is estimating a gradient $\hat{g}(u_k)$.

Stochastic approximation methods for gradient estimation can be classified into three categories [15]: finite difference stochastic approximation (FDSA) [12, 13], random direction stochastic approximation (RDSA) [16], and simultaneous perturbation stochastic approximation (SPSA) [14, 18].

FDSA approximates the gradient by symmetrically perturbing the current decision, u_k, to collect two measurements at $u_k + c_k e_j$ and $u_k - c_k e_j$. $(c_k)_{k=1}^{\infty}$ is a sequence of perturbation sizes and e_j is a unit vector in the jth entry, $j = 1, 2, \ldots, m$. The gradient of the jth entry is approximated by

$$[\hat{g}(u_k)]_j = \frac{F(u_k + c_k, \omega_k^+) - F(u_k - c_k, \omega_k^-)}{2c_k}. \tag{9}$$

These full finite difference computations require $2m$ perturbations for every state update in an m dimensional system. Thus FDSA may be not well suited to fast implementation when $F(u,\omega)$ is computationally expensive or m is large. Alternatives include RDSA, SPSA, and one-sided differences.

In the RDSA algorithm, all the entries of u experience perturbations simultaneously, $\hat{g}(u_k) = \Delta_k[F(u_k + c_k\Delta_k, \omega_k^+) - F(u_k - c_k\Delta_k, \omega_k^-)]/(2c_k)$, where the entries of the perturbation vector $\Delta_k = [\Delta_{1,k} \ldots \Delta_{m,k}]^{\mathrm{T}}$ may have separate distributions. Only two measurements are required for each iteration in this case. Similarly, the SPSA algorithm also only requires two perturbations and measurements per iteration, however in this approach the gradient estimation in the direction of the perturbation is normalized by the magnitude of the perturbation, i.e., $[\hat{g}(u_k)]_j = [F(u_k + c_k\Delta_k, \omega_k^+) - F(u_k - c_k\Delta_k, \omega_k^-)]/(2c_k\Delta_{j,k})$.

Whether the stochastic approximation estimator converges to a local optima and the rate at which it converges are determined by the step size sequences $(a_k)_{k=1}^{\infty}$ and $(c_k)_{i=k}^{\infty}$. In general, if the following conditions are satisfied, convergence for stochastic approximation will be guaranteed.

$$\sum_{i=k}^{\infty} a_k = \infty, \qquad \sum_{k=1}^{\infty} a_k^2 < \infty, \qquad a_k > 0. \tag{10}$$

$(a_k)_{k=1}^{\infty}$ is usually chosen such that $a_k = Ck^{-\alpha}$ for $\alpha \in (0.5, 1]$ and a positive constant C. A balance between the robustness of the estimator and the rate of convergence is available: in many situations $\alpha = 1$ guarantees the best asymptotic rate, but is sensitive to non-strong convexity of f or improper design of C [19].

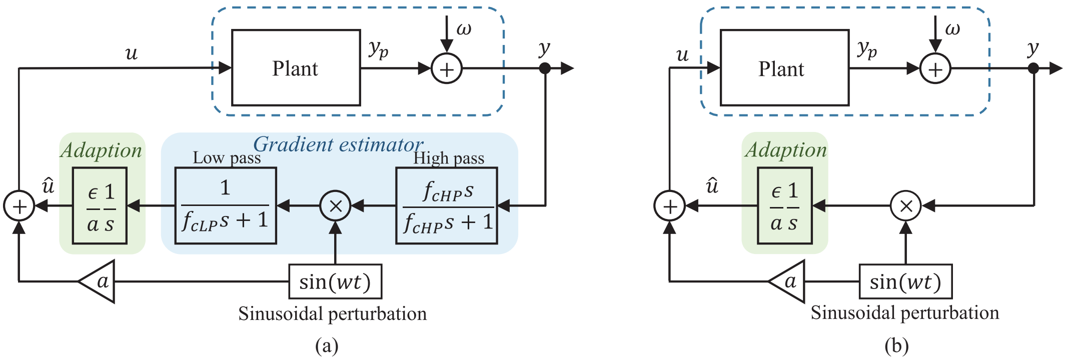

Figure 4 (Color online) Sinusoid-based ES schemes. (a) Classic ES; (b) minimal ES.

The perturbation sequence $(c_k)_{k=1}^{\infty}$ should be designed in conjunction with $(a_k)_{k=1}^{\infty}$ for gradient estimation. In addition to constraint 10, $(a_k)_{k=1}^{\infty}$ and $(c_k)_{k=1}^{\infty}$ should also satisfy the following constranints [13]:

$$\sum_{k=1}^{\infty}\frac{a_k^2}{c_k^2}<\infty,\qquad \sum_{k=1}^{\infty}a_kc_k<\infty,\qquad c_k>0. \tag{11}$$

With several specific assumptions, the output of the FDSA, the RDSA, and the SPSA converges to the optima with probability one, i.e., $u_k \to u^*$ with probability one. More detailed conditions, assumptions and proofs for convergence of the three algorithms are stated in [16–18], respectively.

Note that bias exists in the gradient approximation, and it can be totally eliminated by setting the perturbation sequence $c_k \to 0$ [18]. This may be suitable for static problems, however, it is restrained for online optimization when the system involves drift in the optimum, set-point shifting or varying conditions [15]. Thus stochastic approximation-based ES algorithms in combustion control applications usually adopt constant step sizes [7].

3.2 Sinusoid-based extremum seeking

Since 1950s, sinusoid-based ES has been a popular research topic. Fruitful achievements in theoretical developments and applications of ES control can be found in the recent two decades. A continuous-time classic sinusoid-based ES scheme is shown in Figure 4(a). The design parameters include the gain a and the pulsation ω of the excitation signal, the cut-off frequency f_{cHP} and f_{cLP} of the high pass filter and the low pass filer, the gain ϵ of the integrator. The local stability of the classic sinusoid-based ES feedback scheme was strictly assessed and firstly reported in 2000 by Krstic and Wang [29,32]. Different classes of sinusoid-based ES schemes and applications were summarized by Ariyur and Krstic [28]. Then a simpler sinusoid-based ES scheme named minimal ES scheme was developed in [27,30], as shown in Figure 4(b). The design in the minimal ES is simpler and easier since a, ω and ϵ are the only design parameters. Semi-global practical stability of sinusoid-based ES schemes with respect to design parameters was researched in [30]. The same guiding principle of the sinusoid-based ES schemes reported above is that no knowledge of the plant model is required, thus they often being referred to as black-box schemes. The histories of theoretical developments and applications were well elucidated in the review [21].

One recent application of sinusoid-based ES scheme in engine combustion control was in [34], where the architecture in Figure 4 was implemented for direct injection timing optimization on a gasoline engine which is equipped with both port fuel injection and direct injection. Addressing the multi-variable combustion optimization problem from the sinusoid-based ES view was reported in [26]. Two sinusoidal excitations with distinct different pulsation were designed for simultaneous determination of optimal SA and air/fuel ratio (AFR).

The sinusoid-based ES schemes reported above are black-box and continuous-time optimization algorithms. However, in many applications their plant systems may be inherent discrete, for example, the combustion of engines. Moreover, the plant system may be not absolutely unknown. When a part of the knowledge about the plant is available, it may contribute to fast ES. Fortunately, further developments of

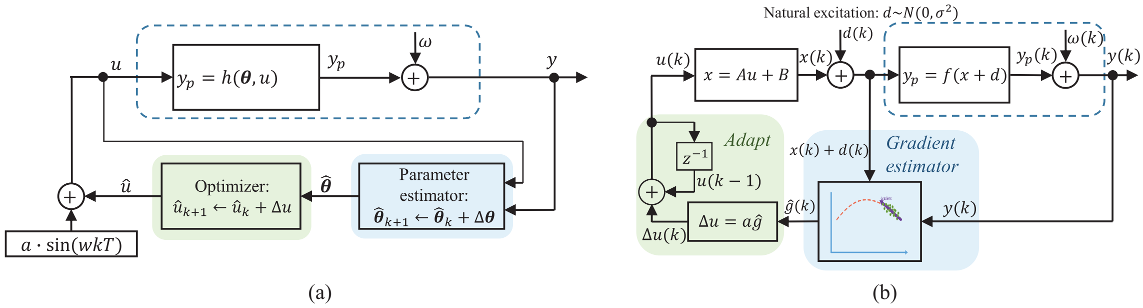

Figure 5 (Color online) Grey-box and natural excitation-based ES schemes. (a) Sampled-data grey-box ES framework; (b) natural excitation-based ES scheme.

the sinusoid-based ES schemes include the sampled-data ES that is applied in discrete time optimization field [31, 33], and the grey-box ES [23] that utilizes knowledge about the plant system.

A sampled-data grey-box ES framework is shown in Figure 5(a). For the sake of explanation, the relationship between u and y_d is assumed to be $y_p = h(\boldsymbol{\theta}, u) = \phi(u)^{\mathrm{T}}\boldsymbol{\theta}$, where $\phi(\cdot)$ is known and $\boldsymbol{\theta}$ is a vector of unknown parameters. A parameter estimator is required instead of the gradient estimator in Figure 4. The estimator $\boldsymbol{\theta}$ provides non-local information about the relationship between u and y_d. Gradient algorithm and least-squares algorithm and their variants for parameter estimation were conducted in [23]. Moreover, an optimizer that drives u_k towards u^* is also required. A class of different algorithms of optimizer including gradient descent algorithm, Newton method, Jacobian matrix transpose, etc., were summarized in [23].

In spark timing optimization of SI engines, the effect of SA on torque (or thermal efficiency, Indicated mean effective pressure (IMEP)) seems a parabola [22, 35]. A quadratic polynomial model can be used relating y and u: $y = \theta_1 u^2 + \theta_2 u + \theta_3 = \phi(u)^{\mathrm{T}}\boldsymbol{\theta}$, where $\boldsymbol{\theta} = [\theta_1 \quad \theta_2 \quad \theta_3]^{\mathrm{T}}$, $\phi(u) = [u^2 \quad u \quad 1]^{\mathrm{T}}$. One of the earliest successful applications of the grey-box ES and the quadratic polynomial model in spark timing optimization was in [22]. The parameter $\boldsymbol{\theta}$ was estimated by a recursive least squares algorithm. However, since the optimizer used tried to directly locate to the optimum that was calculated from the parameter estimator, the initial values of the parameter estimator should be close to the true for the purpose of guaranteeing convergence. More recently, grey-box ES schemes were conducted on alternative fueled engines for online optimization of spark timing [8]. Although the composition of fuel varying, the schemes are robust enough to deal with this situation. Ref. [8] also demonstrated that the grey-box ES is a flexible framework in which different parameter estimation algorithms can be combined with different optimizers.

3.3 Natural excitation-based extremum seeking

Both the stochastic approximation-based ES and the sinusoid-based ES employ periodic excitations. However, periodicity may results in predictability which may be undesired or disadvantageous in some tracking applications [15]. Fortunately, system noise or other signals may naturally exist in some applications. Thanks to the system noise and natural signals that can preform as the excitation, an external excitation signal does not necessary [4, 36–39].

In combustion phase optimization applications, random distribution of combustion phase indicator may performs as the natural excitation [4, 37, 38]. The ES schemes in these applications share the same structure shown in Figure 5(b). The variables u, x and y correspond to spark timing, combustion phase and torque (or IMEP, thermal efficiency). d is a Gaussian distributed random excitation on the combustion phase. A number of recent samples, $\{x(k) + d(k), y(k)\}$, form a 2-dimensional distribution. Then, a linear regression model is applied to obtain a gradient estimator, $\hat{g}$. Then the estimated gradient is used to update the input, i.e., $u(k) = u(k-1) + a\hat{g}(k)$.

The gradient estimator largely depends on the number of samples, i.e., sample size N. Larger N tends to provide more reliable gradient estimator and thus avoids severe input oscillations, however, it takes more time and results in slower response. On the contrary, smaller N provides faster but noisy gradient estimation. The acceptable sample size between 50 and 100 was recommended in [4].

Recently, a probabilistic guaranteed gradient estimation method is applied instead of the gradient estimation using a constant size of samples [37, 38]. A gradient estimator with a desired precision can be obtained, by adaptively adjusting sample size N. The desired precision of gradient estimator can be expressed in the following probability constraint inequity.

$$Pr(\hat{g} - \Delta < g < \hat{g} + \Delta) \geqslant \alpha, \tag{12}$$

where g is the unknown real gradient, Δ and α $(0 < \alpha < 1)$ are pre-set parameters of the desired precision of the gradient estimator $\hat{g}$.

Denote by x_{dn} the shorthand for $x(n) + d(n)$. Assume the noise of measurement obeys Gaussian distribution, $\omega \sim N(0, \sigma^2)$, where σ is unknown. For N samples, $\{(x_{dn}, y_n), n = 1, 2, \ldots, N\}$, define the sums:

$$S_{xx} = \sum_{n=1}^{N} (x_{dn} - \bar{x}_d)^2, \tag{13}$$

$$S_{xy} = \sum_{n=1}^{N} (x_{dn} - \bar{x}_d)(y_n - \bar{y}), \tag{14}$$

$$S_{yy} = \sum_{n=1}^{N} (y_n - \bar{y})^2, \tag{15}$$

where $\bar{x}_d = \frac{1}{N}\Sigma_{n=1}^{N} x_{dn}$, $\bar{y} = \frac{1}{N}\Sigma_{n=1}^{N} y_n$. The unbiased estimator of σ^2 is

$$s^2 = \frac{1}{N-2}\left(S_{yy} - \frac{S_{xy}^2}{S_{xx}}\right). \tag{16}$$

For N samples, $\{(x_{dn}, y_n), n = 1, 2, \ldots, N\}$, the confidence interval, $(\hat{g} - \Delta_{\alpha,N}, \hat{g} + \Delta_{\alpha,N})$, for g with confidence level α is given by

$$\Delta_{\alpha,N} = t_{\frac{1+\alpha}{2},N-2} \cdot s\sqrt{\frac{\Sigma_{n=1}^{N} x_{dn}^2}{N S_{xx}}}, \tag{17}$$

where $t_{\frac{1+\alpha}{2},N-2}$ is obtained by looking up the table of t-distribution with $N-2$ degrees of freedom. The probability constraint inequity (12) can be meet by adaptation of N that satisfying $\Delta_{\alpha,N} \leqslant \Delta$.

The natural excitation-based ES scheme using probabilistic guaranteed gradient estimation method not only guarantees the accuracy of gradient estimator but also adaptively adjusts the sample size, achieving a tradeoff between a rapid response and a stable control input sequence.

4 Stochastic threshold control

4.1 Likelihood-based controller

As shown in Figure 6, the likelihood-based controller determines $\Delta u(k)$ by examining the likelihood of the observed system state sequence relative to the desired rate target. Here, x denotes a binary variable calculated from the raw data x_r which is a random signal without correlationship between $x_r(i)$ and $x(j)$, $i \neq j, i, j \in \{1, \ldots, k, \ldots\}$. And the relationship between x and x_r can be written as

$$x(k) = \begin{cases} 1, & \text{if } x_r(k) \geqslant T, \\ 0, & \text{if } x_r(k) < T, \end{cases} \tag{18}$$

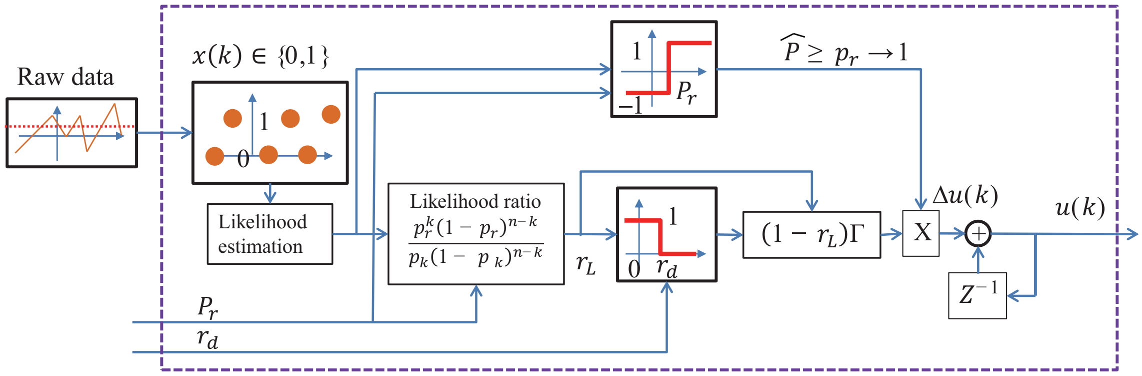

Figure 6 (Color online) Structure of likelihood controller.

and the probability that event $x(k) = 1$ happens, which defined as $p = \Pr\{x(k) = 1\}$, should be kept as reference value p_r. Thus, the goal of likelihood-based boundary exploration is to find u_L^* which satisfies

$$u_L^* = \arg\min_u \{|p - p_r|^2\}. \tag{19}$$

p is a constant when the control input u is fixed. Thus, x obeys binomial distribution under fixed control input u. Although the observation information is l and n not directly p, based on the principle of maximal likelihood estimation, the estimated probability which maximizes the likelihood can be calculated as

$$p_{\max} = \frac{l}{n}. \tag{20}$$

Then, the probability of obtaining l times of $x(k) = 1$ during n times of observation can be therefore given by

$$P_n(l) = C_n^l p_{\max}^l (1 - p_{\max})^{n-l}. \tag{21}$$

However, the absolute values of the probabilities are hard to use for control purposes. For instance, if the p_r is required as 1% and the p is truly 1%, then the probability of exactly one $x(k) = 1$ arriving in the first 40 steps is quite low ($P_{40}(1) = 0.27$). This might suggest that the $p_{\max}$ is not 1% and that a control input adjustment is required. However, even the occurrence of exactly one $x(k) = 1$ in the first 100 cycles (which perfectly matches the target rate), has a probability, $P_{100}(1) = 0.37$, which is not that dissimilar.

A more useful measure is the likelihood ratio, $L_n(l)$, which compares the probability of obtaining a given outcome if the underlying rate of $x(k) = 1$ is truly p, relative to the same probability computed for the estimated $p_{\max} = \frac{l}{n}$. The likelihood ratio may, therefore, be expressed as

$$L_n(l) = \frac{C_n^l p^l (1-p)^{n-l}}{C_n^l p_{\max}^l (1 - p_{\max})^{n-l}}, \tag{22}$$

where, in the limiting cases, 0^0 is to be 1. Figure 7 shows the plot of likelihood ratio as a function of cycle number for 0, 1, 2, and 3 $x(k) = 1$ events, respectively. Likelihood ratio is 1 when observations match the demanded underlying rate, for instance, at $n = 100$ for one event, $n = 200$ for two events and so on. Since likelihood ratios are in the range 0 to 1, using them to evaluate the consistency to the observations with the desired rate is easier than using probabilities directly. With threshold for likelihood ratio as 0.4, for example, the controller firstly judges that whether a control input adjustment is necessary or not. When $L_n(l) > 0.4$, adjustment is not necessary, while, conversely, adjustment is required as $L_n(k) \leqslant 0.4$. This test should be applied to test the likelihood ratio of the last 0, 1, 2, and 3 events through a short four-element first-in-first-out (FIFO) buffer in order to maintain the sensitivity of the algorithm which is discussed in details in [2]. Moreover, if control input adjustment is necessary, the controller will determine whether increase or decrease the control input through a comparison of the target rate and the calculated rate which is also depend on the relationship between x and u.

$$\Delta u(k) = R_L[1 - L_n(l)], \tag{23}$$

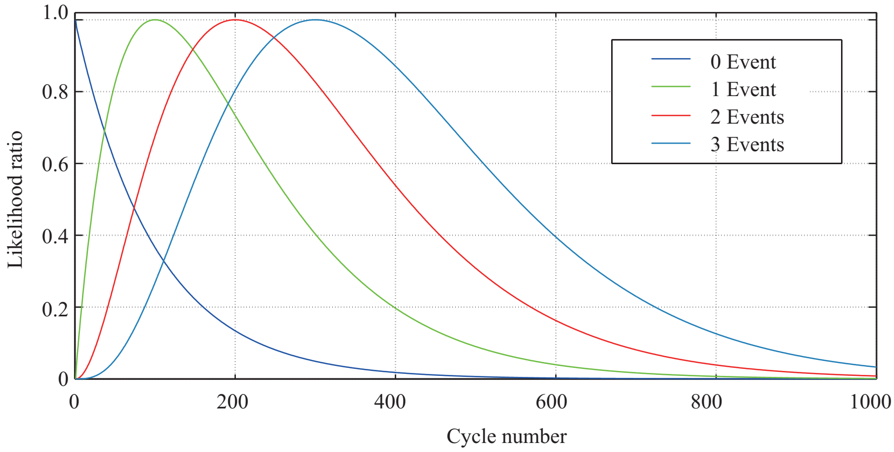

Figure 7 (Color online) Likelihood ratio for different $x(k) = 1$ events.

where R_L is the adjustment gain.

4.2 Statistical learning-based controller

In this subsection, an alternative boundary exploration method, statistical learning-based controller, is presented which is motivated by the likelihood-based method. Likelihood-based strategy is essentially a proportional controller which decides the output magnitude according to the estimation of probability for binary distribution. In statistical learning-based strategy, instead of taking binomial signals as feedback, the raw data is used as feedback signal whose probability distribution is supposed to be normal distribution. Exponential moving mean and variance method is implemented to provide the mean and variance estimation values of the raw data.

As shown in Figure 8, under the same control input, probability distribution function of the raw data x_r is identical step by step. Then, with the same threshold T as the green dash line, the probability $\Pr\{x_r > T\}$ is identical in every step which means $p = \Pr\{x_L = 1\}$ is also identical. Besides, x_r is independent step by step. x_L is then step-to-step independent and the probability distribution is binomial and identical which is the same as x in previous discussion about likelihood-based controller. Here, we use x_L to make a distinguish from the raw data signal x_r since we essentially regard x_r as the feedback signal for statistical learning-based controller. Based on this, maximum likelihood estimation of p can be calculated from the binomial feedback signal $\{x_L(k), \dots, x_L(k+n)\}$ and the control input adjustment is given after obtaining the likelihood ratio between estimation $p_{\max}$ and the target p_r, which is the basic idea of likelihood-based control.

Different from likelihood-based control, statistical learning-based control focuses on the continuous raw data x_r as shown in Figure 9. The mean $\bar{x}_r$ and deviation σ_r^2 are estimated from the raw data measurement x_r based on exponential moving average and variance which can be expressed as

$$\bar{x}_r(k+1) = \bar{x}_r(k) + \beta(x_r(k+1) - \bar{x}_r(k)), \tag{24}$$

$$\sigma_r^2(k+1) = (1-\beta)(\sigma_r^2(k) + \beta(x_r(k+1) - \bar{x}_r(k))^2). \tag{25}$$

The reference mean value $\bar{x}_s$ can be calculated as

$$\bar{x}_s(k) = T - 2.33\sigma_r(k). \tag{26}$$

Then, the closed-loop control strategy can be expressed as

$$u_s(k+1) = u_s(k) + R_b(\bar{x}_r(k) - \bar{x}_s(k)). \tag{27}$$

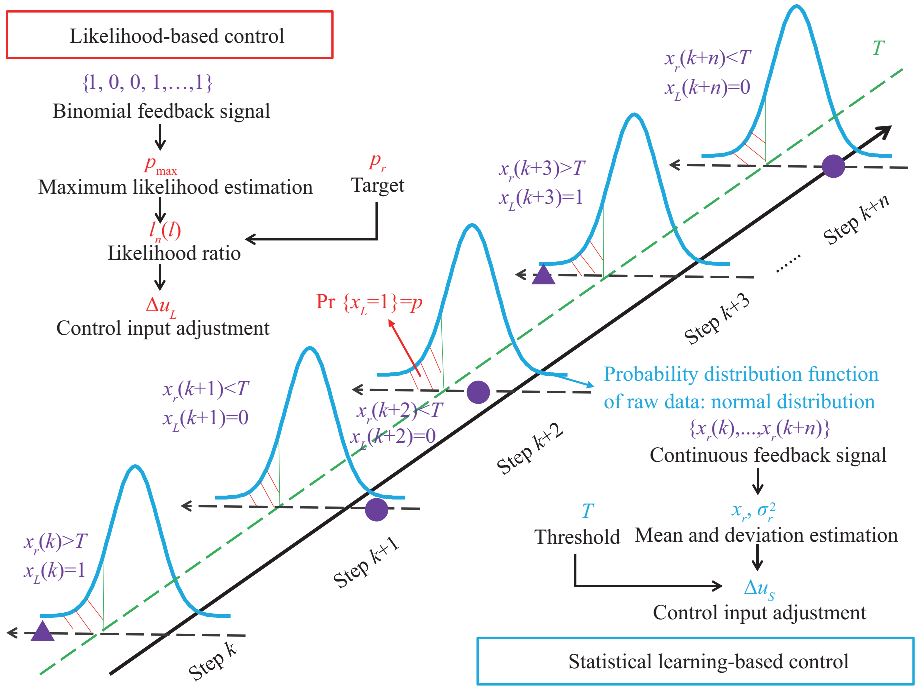

Figure 8 (Color online) Relationship between likelihood-based method and proposed method.

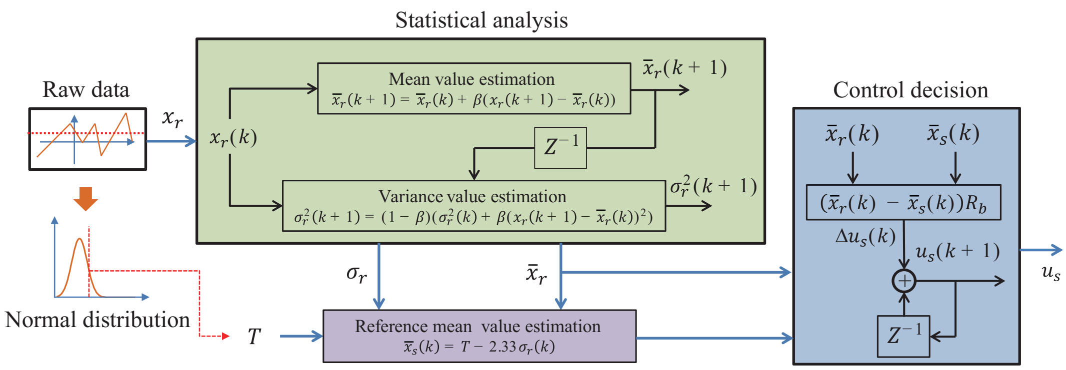

Figure 9 (Color online) Structure of statistic learning-based controller.

5 Feedforward learning control

In engine combustion control applications, the optimal controllable variable values have short-term variations induced by changes in operating point and long-term drift caused by component aging or changes in the environment. Feedforward learning control algorithms that succeed in making the engine an intelligent system capable of learning the optimal controllable variable values are state-of-the-art real-time engine control techniques [9].

Existing on-board map learning algorithms mainly include efficient recursive least-squares techniques-based, extended Kalman filter-based algorithms (see [46–48]). A new stochastic gradient-based online learning algorithm for map learning and its application to feedforward map learning of combustion control will be reviewed as follows.

We start the map learning with the well-known bilinear interpolation method for 2-dimensional (2D) map, as shown in Figure 10(a). $\boldsymbol{x} = [x^1, x^2]^{\mathrm{T}}$ is a 2-dimensional vector. i and j are the normalized coordinate values of x^1 and x^2, respectively. Each regular grid point possesses its corresponding value $w_m, m = 1, 2, 3, 4$. For the point that is not precisely on the regular grid, its value is calculated by bilinear

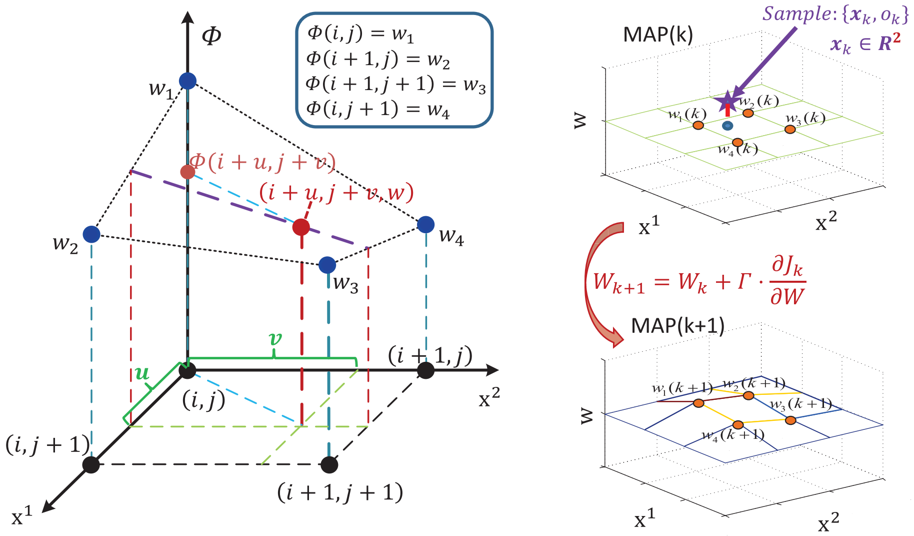

Figure 10 (Color online) 2D map learning. (a) Bilinear interpolation method; (b) 2D map learning process.

interpolation

$$
\begin{aligned}
\Phi(i+u, j+v) &= \phi^{\mathrm{T}}(u, v)\, \boldsymbol{W} \\
&= [(1-u)(1-v), u(1-v), uv, (1-u)v] \cdot [w_1, w_2, w_3, w_4]^{\mathrm{T}}.
\end{aligned} \tag{28}
$$

Define the training set

$$
\{(\boldsymbol{x}_k, o_k)\}, \quad k = 1, 2, \ldots, K, \quad \boldsymbol{x}_k \in \mathbb{R}^d, \quad o_k \in \mathbb{R}, \tag{29}
$$

and the model to be learned from the training set

$$
\Phi(\boldsymbol{x}) = \phi^{\mathrm{T}}(\boldsymbol{x}) \cdot \boldsymbol{W}, \tag{30}
$$

where $\boldsymbol{W} = [w_1, w_2, \ldots, w_L]^{\mathrm{T}}$.

The learning problem is to find the optimal model parameter $\boldsymbol{W}$ minimizing the following loss function:

$$
J(\boldsymbol{W}) = \sum_{k=1}^{K} J_k(\boldsymbol{W}) = \sum_{k=1}^{K} \frac{1}{2} ||o_k - \Phi(\boldsymbol{x}_k)||^2, \tag{31}
$$

where $J_k(\boldsymbol{W})$ is the learning objective for the sample $(\boldsymbol{x}_k, o_k)$.

$$
J_k(\boldsymbol{W}) = \tfrac{1}{2}||o_k - \phi_k^T \cdot \boldsymbol{W}||^2 = \tfrac{1}{2} o_k^2 + \tfrac{1}{2} \boldsymbol{W}^{\mathrm{T}} \phi_k \phi_k^T \boldsymbol{W} - o_k \phi_k^{\mathrm{T}} \boldsymbol{W}, \tag{32}
$$

where $\phi_k = \phi(\boldsymbol{x}_k)$. The partial derivative with respect to $\boldsymbol{W}$ can be deduced.

$$
\frac{\partial J_k}{\partial \boldsymbol{W}} = \phi_k \phi_k^{\mathrm{T}} \boldsymbol{W} - \phi_k o_k = -\phi_k (o_k - \phi_k^{\mathrm{T}} \boldsymbol{W}) = -\phi_k e_k, \tag{33}
$$

where e_k is the error between the observation o_k and the model output $\phi_k^{\mathrm{T}} \boldsymbol{W}$.

Let $\{(\boldsymbol{x}_k, o_k)\}, k = 1, 2, \ldots, K$ be the streaming data. Then the sample will be considered one-by-one and $\boldsymbol{W}$ will be updated based on the gradient calculated from $\{(\boldsymbol{x}_k, o_k)\}$ in (33), as shown in Figure 10(b). This is the so called stochastic gradient descent algorithm:

$$
\boldsymbol{W}_{k+1} = \boldsymbol{W}_k - \gamma \phi_k e_k, \tag{34}
$$

where γ represents the update gain or step size for the gradient descent algorithm. Note that the convergence is guaranteed if γ is chosen such that $\gamma < 2$ [49].

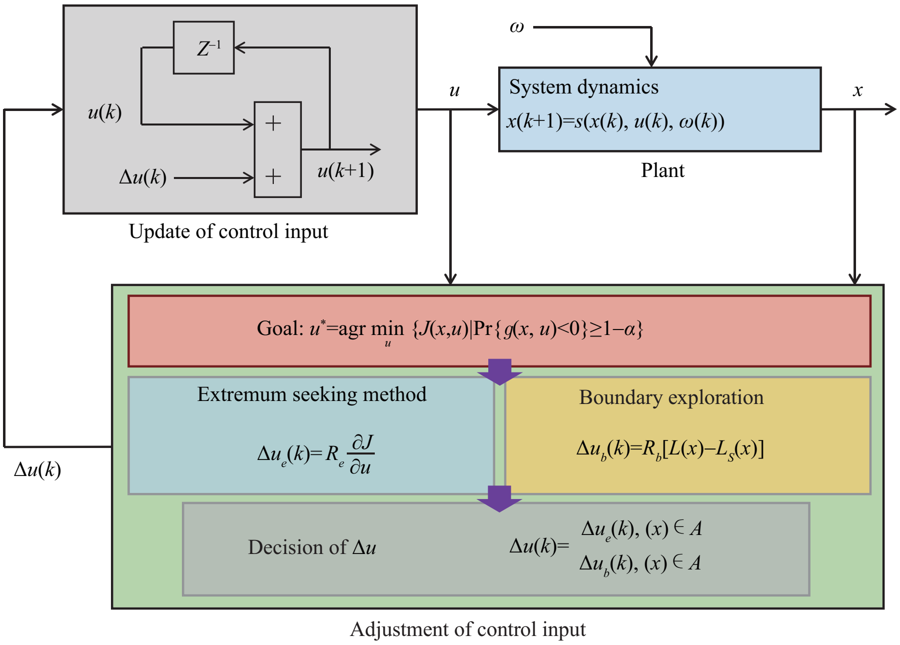

Figure 11 (Color online) Structure of iteration learning-based solver for probabilistic constrained optimization problem.

6 Experimental case studies

6.1 Knock probabilistic constrained optimal combustion control

6.1.1 *Iterative learning-based solution*

In this section, we describe a model-free methodology for solving the probabilistic constrained optimization problem described in Section 2, which we call the iterative learning-based solution. The key observation behind the model-free methodology is that, by extracting and analyzing the measurement sequences of the control input and plant output, the optimal solution can be approaching iteratively in real-time.

Essentially, this is a data driven-based methodology. As shown in Figure 11, the iteration learning-based solver aims at obtaining the solution which satisfies

$$u^* = \arg\min_u \{J(x,u)|\Pr\{g(x,u) < 0\} \geqslant 1-\alpha\}. \tag{35}$$

The system dynamics of the addressed plant is discrete and can be written as

$$x(k+1) = s(x(k), u(k), \omega(k)), \tag{36}$$

where the system state at k-th step is a function of the system state at k-th step $x(k)$, the control input $u(k)$ and the system noise $\omega(k)$ whose probability distribution can be arbitrary. The iteration process can be expressed as (6).

To approach the optimal solution under probabilistic constraint, the iteration learning-based method consists of two independent parts. One is the extremum seeking block and the other one is the boundary exploration block. Extremum seeking method drives control input to the unconstrained optimal solution. Although the unconstrained optimal solution may satisfy the probabilistic constraint in some cases, there exists the risk that the probabilistic constraint is violated. Thus, the boundary exploration block is to explore the control input boundary for the probabilistic constraint and operate the control input along the boundary. Obviously, the goals of extremum seeking block and boundary exploration block are inconsistent. Thus, these two methods can be combined by defining a judgement process that decides which adjustment should be adopted. $h(x)$ is a function defined as an indicator: $h(x) \in A$ means that

Table 1 Specifications of the engine

Specification	Description	Specification	Description
Type	2GR-FSE Engine, TOYOTA Inc.	Valve mechanism	24-Valve DOHC
Cylinder arrangement	6-Cylinder, V Type	Fuel system	SFI D-4S
Displacement	3456 ml	Cylinder diameter	94 mm
Compression ratio	11.8:1	Max. torque	375 Nm @ 4800 rpm

the control input is operated in the safe area and the adjustment from extremum seeking method $\Delta u_e(k)$ is used while the adjustment from boundary exploration method $\Delta u_b(k)$ is adopted if $h(x) \in \bar{A}$. In this way, extremum seeking method drives the control input to the optimal solution iteratively. While, if the optimal solution cannot be approached because of the probabilistic constraint, the controller switches to the boundary exploration mode to keep the control input along the boundary where we can obtain the best performance under the probabilistic constraint.

In Section 3, we briefly introduce the extremum seeking method which searches the optimal solution by estimating the stochastic gradient on-line based on the obtained measurements. Subsection 4.1 gives the brief introduction of likelihood-based controller which is for probability threshold control. Moreover, statistical learning-based controller is presented in Subsection 4.2. Both likelihood-based controller and statistical learning-based controller can be used for boundary exploration.

6.1.2 *Experimental conditions*

The experimental validations were carried out on the six-cylinder SI gasoline engine test bench. The specifications are tabulated in Table 1.

The engine was operated at steady state with engine speed as 1200 rpm and throttle valve as 12° during experiments. The cooling water control system maintained the cooling water temperature at 80 ± 1.5°C. Besides, AFR control is based on simple adaptive control (SAC) and feedforward map [45]. Experimental validation concerned both steady and transient AFR cases. In the steady case, the AFR was fixed at 17.52, i.e., $\lambda = 1.2$ where λ represents the air/fuel equivalence ratio. While, λ was controlled from 1.1 to 1.3 in the transient case.

6.1.3 *Comparison with standard control algorithm*

Standard extremum seeking method suits for the unconstrained optimization problem. The cost function considers only about thermal efficiency and is written as

$$f_s(\mathrm{SA}) = \eta(\mathrm{SA}). \tag{37}$$

The comparison of more than 900 cycles control performance between standard extremum seeking method and proposed method is plotted in Figures 12 and 13. As shown in Figure 12(a), the red line is the SA operated by standard extremum seeking method which is driven to much higher value than operated by the proposed method. As shown in Figure 12(b), the standard extremum seeking method estimated gradients were around zero while the proposed method estimated gradients by the extremum seeking part deviated from zero. This is because, the standard extremum seeking method generated control input around optimal value where the gradient should be zero. Consequently, the standard extremum seeking method achieved better thermal efficiency than the proposed method as shown in Figure 13(b) and Table 2. However, the resulted knock probability by standard extremum seeking method is much higher than the proposed method as shown in Figure 13(a), which is 42.18% to 0.85% given in Table 2. This is because that the standard extremum seeking method does not consider the constraint of knock probability while the proposed method can make a tradeoff between thermal efficiency and knock probability constraint and the loss in the thermal efficiency is less than 0.4% while the knock probability drops around 41.3%, which is worthy.

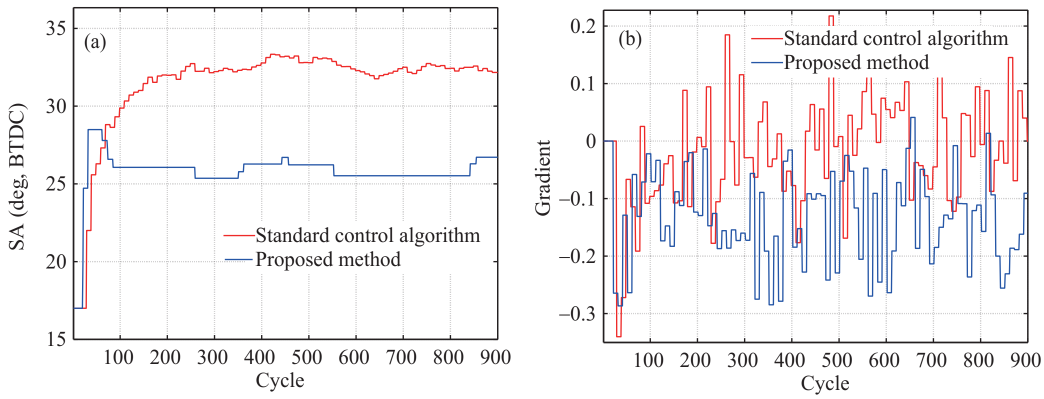

Figure 12 (Color online) Comparison between standard control algorithm and the proposed method. (a) Spark advance inputs; (b) gradient variation ($\lambda = 1.2$).

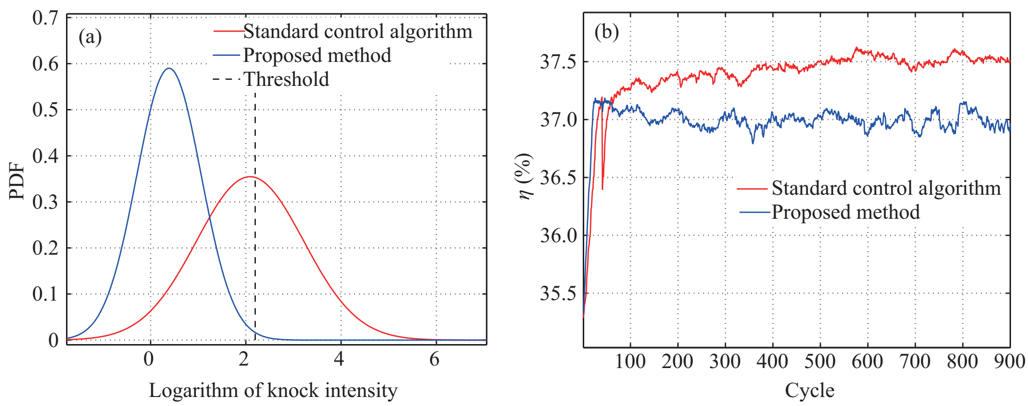

Figure 13 (Color online) Comparison between standard control algorithm and the proposed method. (a) Logarithm of knock intensity; (b) thermal efficiency ($\lambda = 1.2$).

Table 2 Comparison of results from different control methods

Item	Standard	$\beta = -1$	$\beta = -5$	Proposed
Average thermal efficiency (%)	37.41	37.27	37.16	37.02
Knock probability (%)	42.18	23.21	17.63	0.85
Number of knock cycles	380	209	159	8

6.1.4 *Comparison with other control algorithm*

The method proposed in [34] deals with the physical constraints in SA optimization though converting the constraints in the performance function by introducing a penalty factor, as written in the following equation:

$$f(\mathrm{SA}) = \eta(\mathrm{SA}) + \beta KI(\mathrm{SA}), \tag{38}$$

where $f(\mathrm{SA})$ is the new constructed performance function including both thermal efficiency and knock intensity, β is the penalty factor. But litter knowledge about β determination at various operation conditions is exist. In this experimental validation, two different β values, -1 and -5 were concerned. As shown in Figures 14 and 15 together with the results listed in Table 2, although SA was relatively retarded with smaller β and resulted in reduced knock probability, the precise value of β for borderline of one percent knock probability is unknown and difficult to be determined. While, proposed method kept operating SA near the borderline of one percent knock probability.

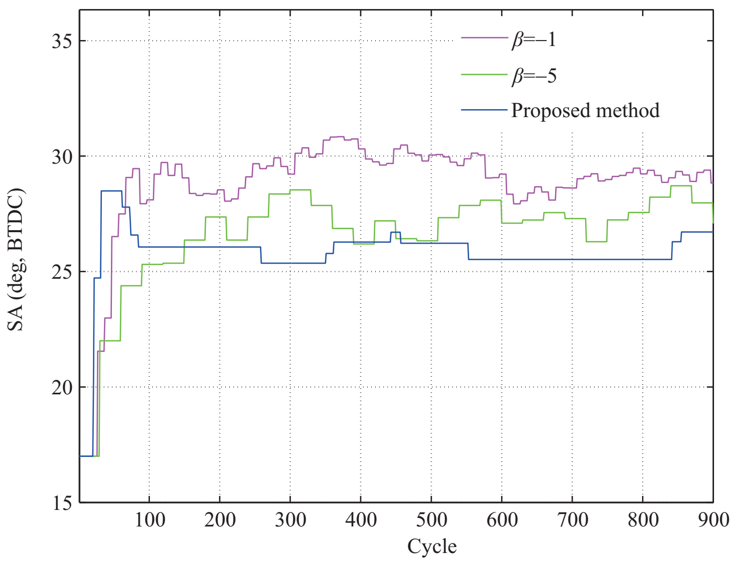

Figure 14 (Color online) Comparison between other control algorithm and the proposed method: spark advance inputs ($\lambda = 1.2$).

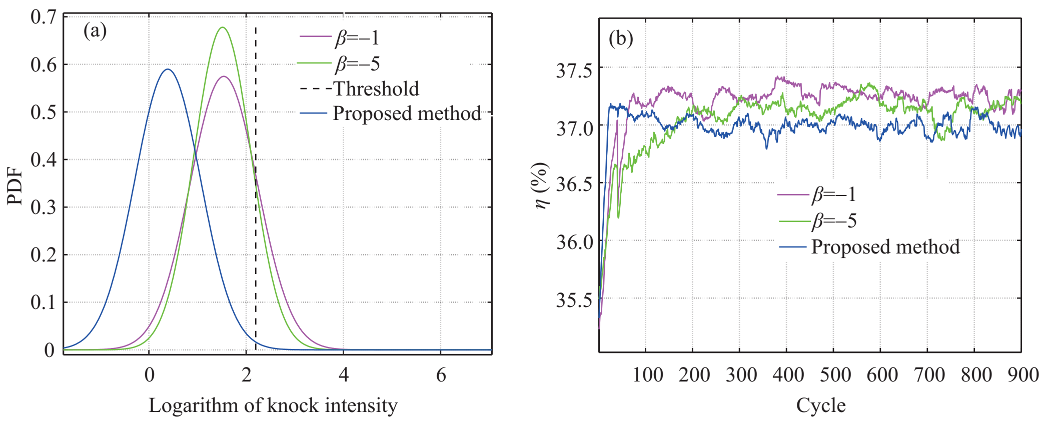

Figure 15 (Color online) Comparison between other control algorithm and the proposed method. (a) Logarithm of knock intensity; (b) thermal efficiency ($\lambda = 1.2$).

6.1.5 *Transient case*

In the transient case, the air/fuel equivalence ratio λ was adjusted from 1.1 to 1.3 as in Figure 16(a). The proposed self-optimization method with knock probability threshold adjusted SA to new borderline and the fuel thermal efficiency was improved as plotted in Figure 16(b) and (c). Figure 16(d) shows that the knock rate was limited and the accurate value is 0.78%.

6.2 On-board map learning-based combustion control

It is known that the optimal spark timing and combustion phase are operating point dependent. Thus, the combustion control in this case study combines the probabilistic guaranteed gradient learning-based ES scheme and the 2D feedforward map learning algorithm, as shown in Figure 17. When engine running at steady operating point or mild transient operating condition, the ES loop searches the optimal SA that maximizes the thermal efficiency, η, in real-time. When engine running at transient operating condition, SA should be adjusted to compensate the drift of optimal value caused by varying speed and/or load. This operating-point-dependent drift could be learned in the map and thus provides a proper feedforward, $\mathrm{SA_{FF}}$.

To implement on-board map learning-based combustion control scheme in Figure 17, the map $(p_m, n) \rightarrow$

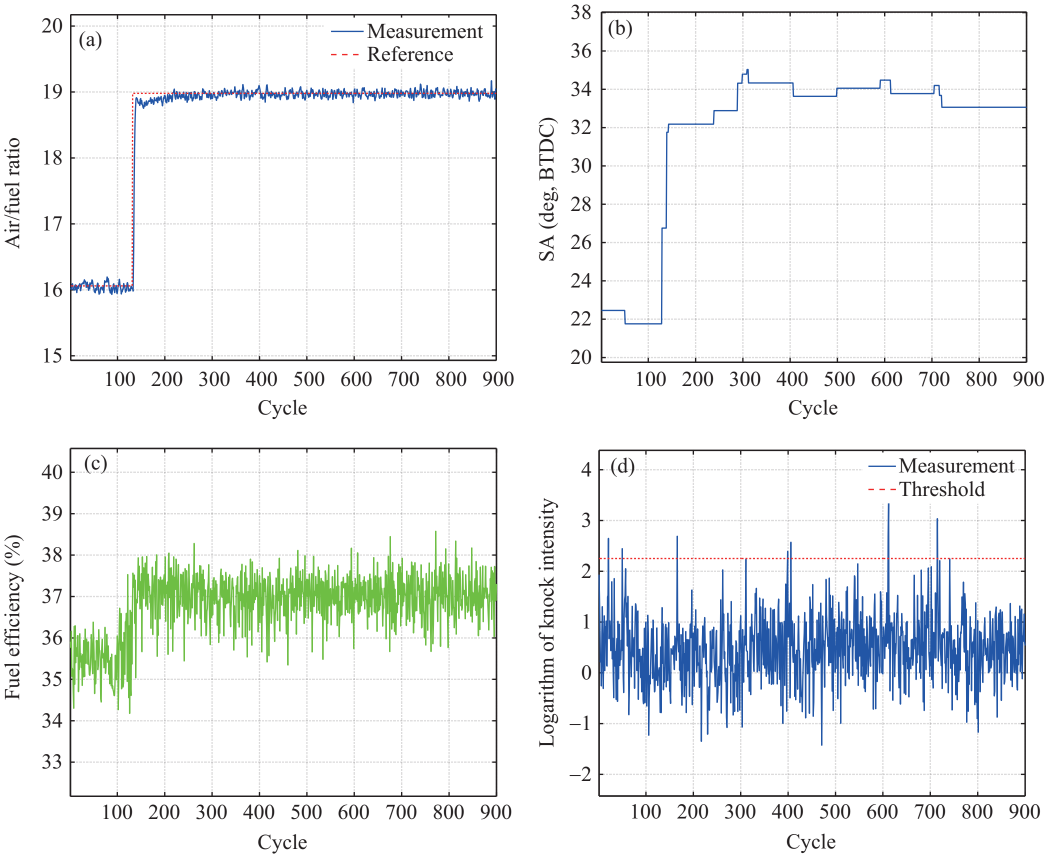

Figure 16 (Color online) Transient result from $\lambda = 1.1$ to $\lambda = 1.3$. (a) Air/fuel ratio from 16.06 to 19.98; (b) spark advance; (c) fuel efficiency; (d) logarithm of knock intensity.

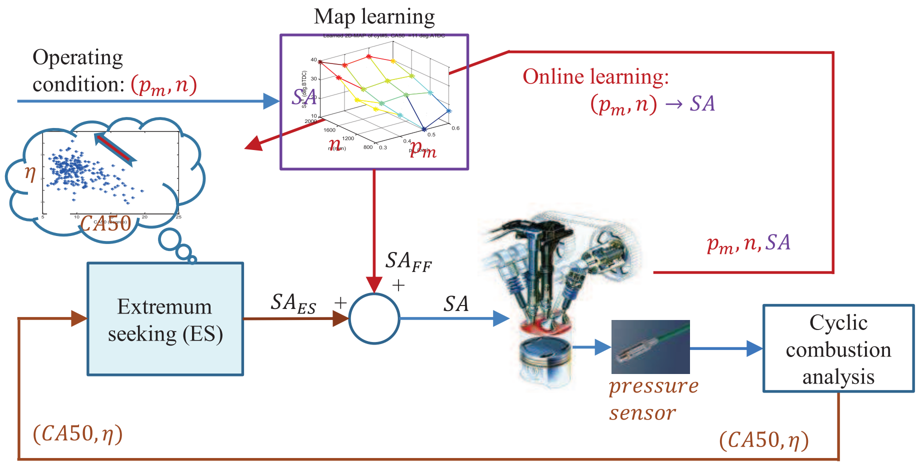

Figure 17 (Color online) A framework of map learning and extremum seeking-based combustion control scheme.

SA should be firstly cerated. The manifold pressure p_m and engine speed n are discretized and then form the grid: $p_m = [0.3, 0.4, 0.5, 0.6]$ bar, $n = [800, 1200, 1600, 2000]$ rpm. The map includes 16 operating grid points.

Experiments at the 16 steady operating points are then carried out respectively. Take one operating point, $(p_m, n) = (0.5, 1200)$, as example. The parameters of the probabilistic guaranteed gradient estimation algorithm are initialized as follows: $\Delta = 0.04$, $\alpha = 0.68$. The extremum seeking process is shown in Figure 18(a)–(f). Figure 18(a) shows the enable signal of SA ES controller and the iteration number k. From the enable signal, it is easy to know that the experiment is repeatedly performed. SA

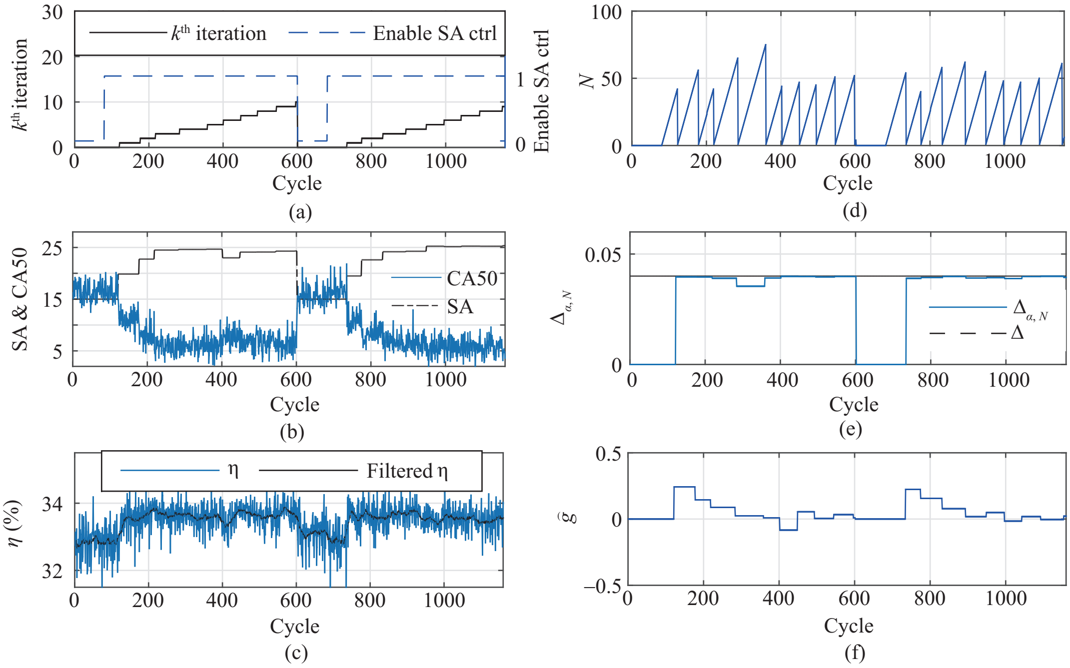

Figure 18 (Color online) Control performance at the operating point $(p_m, n) = (0.5, 1200)$.

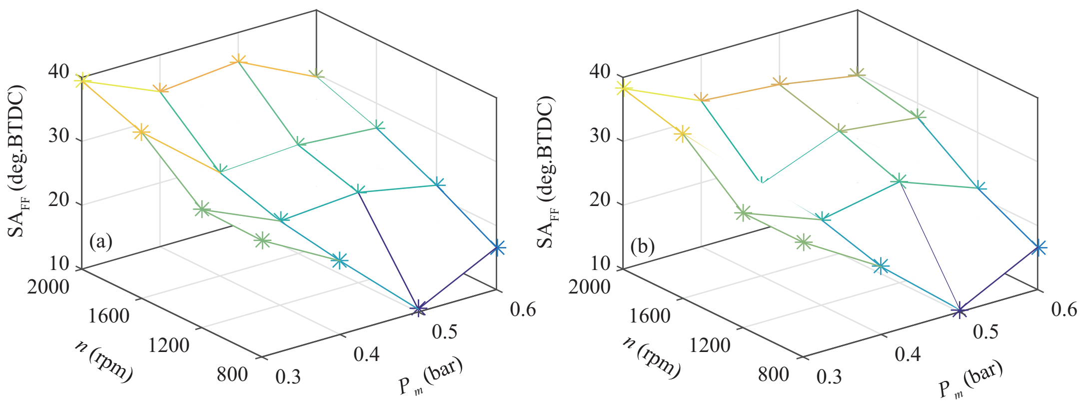

Figure 19 (Color online) Learned SA map $(p_m, n) \to$ SA. (a) The 5th cylinder; (b) the 6th cylinder.

decision sequence and CA50 trace are shown in Figure 18(b). In Figure 18(c), the raw trace of thermal efficiency η, is shown together with the low-pass filtered trace. Figure 18(d) shows the sample number N for gradient estimation in the kth iteration. The confidence interval parameter $\Delta_{\alpha,N}$ for gradient estimator is shown in Figure 18(e). The probabilistic guaranteed gradient estimator, $\hat{g}$, in each iteration is shown in Figure 18(f). Comparing the repeated two runs (0–600 cycle is the first run and 600–1200 cycle is the second run), extremum seeking controller maintains SA around the optimal value where the maximal η is obtained; however, two SA decision sequences are different due to its randomized samples for gradient estimation. It is apparent that the sample number N for gradient estimation in Figure 18(d) can be adjusted adaptively along the iterative process, such that $\Delta_{\alpha,N}$ in Figure 18(e) is within the pre-set boundary Δ. Repeat the experiments at other 15 steady operating points, and then the initial map can be created, as shown in Figure 19. Figures 19(a) and (b) show the resulting map of the 5th and 6th cylinder, respectively.

With the learned initial map, transient experiment at constant torque mode is then carried out, as shown in Figure 20. Figure 20(a) shows the step throttle angle command. Figures 20(b)–(d) show the traces of IMEP, p_m, and n. The map-based feedforward SA_{FF} and spark advance command are plotted

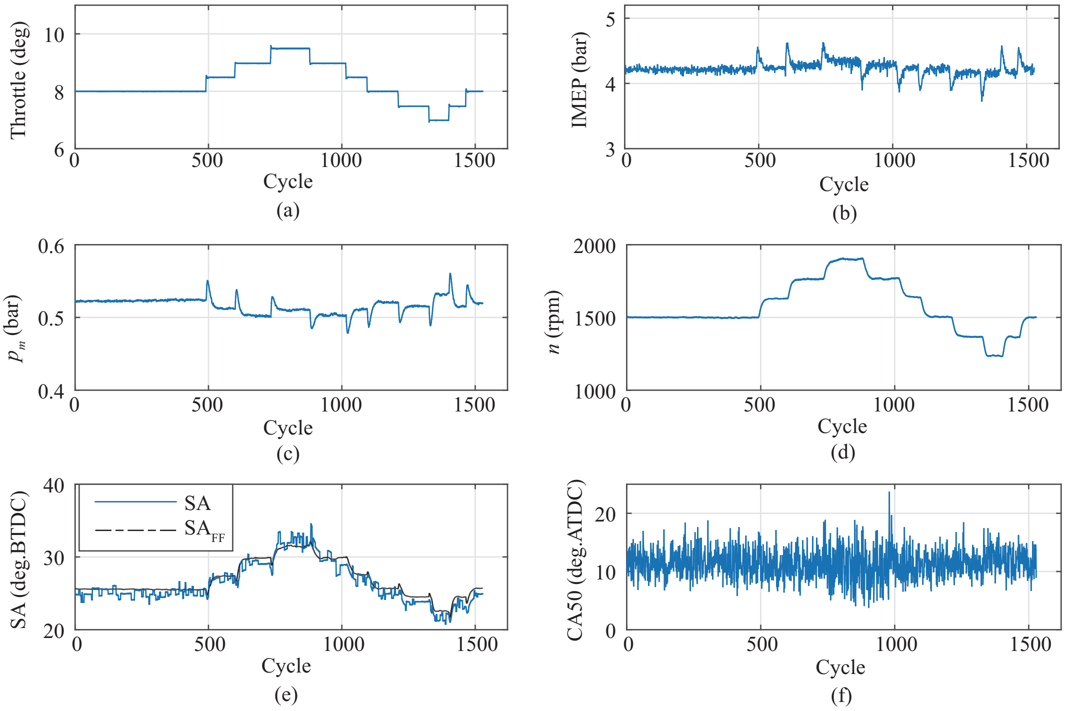

Figure 20 (Color online) Transient experiment result of map learning and ES-based combustion control scheme.

in Figure 20(e). The resulting CA50 is plotted in Figure 20(f).

7 Conclusion

This article has surveyed the literatures on real-time iterative learning and optimization algorithms for the application in combustion control, especially the spark advance control. The related contributions and conclusion can be summarized as follows:

- The general framework for combustion quality optimization and control problems has been formulated based on the exist related references for the first time, including online optimization of combustion phase, stochastic threshold control, and operating-point-dependent feedforward online adaptation.
- The iterative solution-based algorithms under the general framework has been summarized.
- Stochastic approximation-based ES, natural perturbation-based ES are reviewed for online optimization problems.
- Likelihood-based algorithm and statistical learning-based algorithm are reviewed for stochastic threshold control problems.
- On-board map learning algorithm is presented for operating-point-dependent feedforward adaptation problems as example.
- Two application case studies: knock probabilistic constrained optimal combustion control and on-board map learning-based combustion control, are presented with experimental results as examples for showing how to implement these algorithms in practical applications under the summarized general framework.

Acknowledgements The authors would like to thank Toyota Motor Corporation, Japan, for the financial and technical supports in this research.

References

1 Emiliano P. Spark ignition feedback control by means of combustion phase indicators on steady and transient operation. J Dyn Syst-T ASME, 2014, 136: 051021

2 Jones J C P, Spelina J M, Frey J. Likelihood-based control of engine knock. IEEE Trans Contr Syst Technol, 2013, 21: 2169–2180
3 Kaleli A, Ceviz M A, Erenturk K. Controlling spark timing for consecutive cycles to reduce the cyclic variations of SI engines. Appl Thermal Eng, 2015, 87: 624–632
4 Corti E, Forte C. Spark advance real-time optimization based on combustion analysis. J Eng Gas Turbines Power, 2011, 133: 092804
5 Shen T L, Kang M X, Gao J W, et al. Challenges and solutions in automotive powertrain systems. J Control Dec, 2017, 5: 61–93
6 Janakiraman V M. Machine learning for identification and optimal control of advanced automotive engines. Dissertation for Ph.D. Degree. Ann Arbor: University of Michigan, 2013. 1–32
7 Popovic D, Jankovic M, Magner S, et al. Extremum seeking methods for optimization of variable cam timing engine operation. IEEE Trans Contr Syst Technol, 2006, 14: 398–407
8 Mohammadi A, Manzie C, Nešić D. Online optimization of spark advance in alternative fueled engines using extremum seeking control. Control Eng Practice, 2014, 29: 201–211
9 Malikopoulos A. Real-time, self-learning identification and stochastic optimal control of advanced powertrain systems. Dissertation for Ph.D. Degree. Ann Arbor: University of Michigan, 2008. 1–13
10 Zhu G M, Haskara I, Winkelman J. Stochastic limit control and its applications to spark limited control using ionization feedback. In: Proceedings of American Control Conference, Portland, 2005. 5027–5034
11 Shen X, Zhang Y H, Shen T L, et al. Spark advance self-optimization with knock probability threshold for lean-burn operation mode of SI engine. Energy, 2017, 122: 1–10
12 Blum J R. Multidimensional stochastic approximation methods. Ann Math Statist, 1954, 25: 737–744
13 Kiefer J, Wolfowitz J. Stochastic estimation of the maximum of a regression function. Ann Math Statist, 1952, 23: 462–466
14 Spall J C. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. IEEE Trans Automat Contr, 1992, 37: 332–341
15 Manzie C, Krstic M. Extremum seeking with stochastic perturbations. IEEE Trans Automat Contr, 2009, 54: 580–585
16 Kushner H J, Clark D S. Stochastic Approximation Methods for Constrained and Unconstrained Systems. New York: Springer, 1978. 19–58
17 Chin D C. Comparative study of stochastic algorithms for system optimization based on gradient approximations. IEEE Trans Syst Man Cyber, 1997, 27: 244–249
18 Spall J C. Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control. Hoboken: John Wiley & Sons, 2003. 95–204
19 Hannah L A. Stochastic optimization. In: International Encyclopedia of the Social & Behavioral Sciences. 2nd ed. London: Elsevier, 2015, 2: 473–481
20 Khong S Z, Tan Y, Manzie C, et al. Extremum seeking of dynamical systems via gradient descent and stochastic approximation methods. Automatica, 2015, 56: 44–52
21 Tan Y, Moase W H, Manzie C, et al. Extremum seeking from 1922 to 2010. In: Proceedings of the 29th Chinese Control Conference, Beijing, 2010. 14–26
22 Scotson P G, Wellstead P E. Self-tuning optimization of spark ignition automotive engines. IEEE Control Syst Mag, 1990, 10: 94–101
23 Nesic D, Mohammadi A, Manzie C. A framework for extremum seeking control of systems with parameter uncertainties. IEEE Trans Automat Contr, 2013, 58: 435–448
24 Manzie C, Moase W, Shekhar R, et al. Extremum seeking methods for online automotive calibration. In: Optimization and Optimal Control in Automotive Systems. Berlin: Springer, 2014. 23–39
25 Hellstrom E, Lee D, Jiang L, et al. On-board calibration of spark timing by extremum seeking for flex-fuel engines. IEEE Trans Contr Syst Technol, 2013, 21: 2273–2279
26 Corti E, Forte C, Mancini G, et al. Automatic combustion phase calibration with extremum seeking approach. J Eng Gas Turbines Power, 2014, 136: 091402
27 Tan Y, Nešić D, Mareels I. On the choice of dither in extremum seeking systems: A case study. Automatica, 2008, 44: 1446–1450
28 Ariyur K B, Krstic M. Real-Time Optimization by Extremum-Seeking Control. Hoboken: John Wiley & Sons. 2003. 1–89
29 Wang H H, Krstic M. Extremum seeking for limit cycle minimization. IEEE Trans Automat Contr, 2000, 45: 2432–2436
30 Tan Y, Nešić D, Mareels I. On non-local stability properties of extremum seeking control. Automatica, 2006, 42: 889–903
31 Choi J Y, Krstic M, Ariyur K B, et al. Extremum seeking control for discrete-time systems. IEEE Trans Automat Contr, 2002, 47: 318–323
32 Krstić M, Wang H H. Stability of extremum seeking feedback for general nonlinear dynamic systems. Automatica, 2000, 36: 595–601
33 Khong S Z, Nešić D, Tan Y, et al. Unified frameworks for sampled-data extremum seeking control: Global optimisation and multi-unit systems. Automatica, 2013, 49: 2720–2733
34 Song K, Xie H, Jiang W, et al. On-line Optimization of Direct-Injection-Timing for SI-CAI Hybrid Combustion in a PFI-DI Gasoline Engine. SAE Technical Paper, 2016-01-0757. 2016
35 Larsson S, Andersson I. Self-optimising control of an SI-engine using a torque sensor. Control Eng Practice, 2008, 16:

505–514
36 Carnevale D, Astolfi A, Centioli C, et al. A new extremum seeking technique and its application to maximize RF heating on FTU. Fusion Eng Des, 2009, 84: 554–558
37 Zhang Y H, Shen T L. Stochastic approximation for combustion phase optimization of SI gasoline engines. In: Proceedings of European Control Conference (ECC), Aalborg, 2016. 1265–1270
38 Gao J W, Wu Y H, Shen T L. On-line statistical combustion phase optimization and control of SI gasoline engines. Appl Thermal Eng, 2017, 112: 1396–1407
39 Zhang Y H, Shen T L. Cylinder pressure based combustion phase optimization and control in spark-ignited engines. Control Theor Technol, 2017, 15: 83–91
40 Donald E K. The Art of Computer Programming: Seminumerical Algorithms. 3rd ed. Boston: Addison-Wesley, 1998. 232
41 Myung I J. Tutorial on maximum likelihood estimation. J Math Psychology, 2003, 47: 90–100
42 Allmaras M, Bangerth W, Linhart J M, et al. Estimating parameters in physical models through bayesian inversion: a complete example. SIAM Rev, 2013, 55: 149–167
43 Aguilar O, Allmaras M, Bangerth W, et al. Statistics of parameter estimates: a concrete example. SIAM Rev, 2015, 57: 131–149
44 Shen X, Shen T L. Real-time statistical learning-based stochastic knock limit control for spark-ignition engines. Appl Thermal Eng, 2017, 127: 1518–1529
45 Khajorntraidet C, Ito K. Simple adaptive air-fuel ratio control of a port injection SI engine with a cylinder pressure sensor. Control Theor Technol, 2015, 13: 141–150
46 Jones J C P, Muske K R. Identification and adaptation of linear look-up table parameters using an efficient recursive least-squares technique. ISA Trans, 2009, 48: 476–483
47 Höckerdal E, Frisk E, Eriksson L. EKF-based adaptation of look-up tables with an air mass-flow sensor application. Control Eng Practice, 2011, 19: 442–453
48 Guardiola C, Pla B, Blanco-Rodriguez D, et al. A learning algorithm concept for updating look-up tables for automotive applications. Math Comput Model, 2013, 57: 1979–1989
49 Gao J W, Zhang Y H, Shen T L. An on-board calibration scheme for map-based combustion phase control of spark-ignition engines. IEEE/ASME Trans Mechatron, 2017, 22: 1485–1496
50 Pipitone E. A comparison between combustion phase indicators for optimal spark timing. J Eng Gas Turbines Power, 2008, 130: 052808