

DATA DRIVEN BASED ESTIMATION AND CONTROL FOR AUTOMOTIVE SYSTEMS

By

Jian Tang

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Mechanical Engineering – Doctor of Philosophy

2022

ABSTRACT

This dissertation focuses on predicting the system responses and using them to improve the automotive system performance based on the data-driven based algorithms. Two applications included are multivariable borderline knock prediction and control and tire-road friction coefficient estimation.

Internal combustion engines are core components of traditional and hybrid passenger vehicles and also widely used for off-road applications. When the combustion is limited by the engine knock, it is desired to operate it as close to its borderline knock limit as possible to optimize combustion efficiency. Traditionally, this limit is detected by sweeping tests of related control parameters, which is expensive and time-consuming; and also, the detected borderline knock limit often is relatively conservative. When more advanced control parameters (subsystems) are added, these sweeping tests lead to tremendous higher test cost. An intelligent and efficient way to predict borderline knock without detailed knowledge of combustion dynamics is proposed. This supervised-learning based Bayesian optimization method is assisted by a surrogate model trained based on the system statistic properties. A two-control-parameter (spark timing and intake valve timing) case is demonstrated for optimizing two competing objectives (knock intensity (KI) and fuel economy).

A complete borderline knock control structure is proposed and divided into three parts. The first part is about offline training with necessary modifications of the Bayesian optimization algorithm. Engine tests are conducted under two different operational conditions to obtain knock borderline limit, indicating the proposed algorithm is able to reduce required experimental budget (cost and time) significantly. The predicted mean Pareto front and its variance can be used to find the optimum control parameters at borderline knock limit for the best fuel economy possible. Smooth response surfaces of surrogate models can also be used as the initial model to be updated in real-time.

The second part is an online updating process, based on the offline-trained surrogate model, using modified likelihood ratio controller. Principal component analysis indicated that spark timing is the most sensitive factor affecting the Pareto front. A two-buffer design was proposed to update the surrogate model under different rates so that both short-term compensation for environment changes and long-term for slow engine aging effect are covered. Both simulation and engine test

results indicate that the proposed control strategy is able to update the machine-learned surrogate models in real-time, which outperforms the conventional knock control strategy and offline-trained knock limit, and especially reduces the conservativeness of borderline knock control significantly.

Finally, to reduce cycle-to-cycle combustion variations, a real-time cycle-wised knock compensation scheme is developed based on the measured exhaust temperature when the engine is operated close to its knock borderline. To make model-based control possible, q -Markov COVER (COVariance Equivalent Realization) system identification was used to obtain a linearized engine exhaust temperature model from change of spark timing to associated variations of exhaust temperature and knock intensity (KI). Accordingly, a Linear-Quadratic-Gaussian (LQG) controller is designed to minimizing the KI fluctuations based on change (δ) of exhaust temperature. For the entire control architecture, results of three test scenarios indicated that the spark timing can be further advanced while maintaining the same knock intensity level due to reduced knock combustion variations.

For the vehicle dynamics research, estimation of tire-road friction coefficient is very important due to new active safety control systems, especially for autonomous vehicles that rely on the accurate estimation of road surface conditions to find vehicle operational boundary and achieve the best performance possible. Several cause- and effect-based methods were proposed with their own limitations. A new evaluation criterion associated with slip-ratio is found based on CarSim simulation data on different road conditions; and strong correlation between proposed criterion and tire-road friction under different road surface conditions is observed. Note that the data-driven based method proposed in this dissertation only utilizes the statistic information from existing production vehicle sensors without increasing hardware cost. A computational cheap black-box model of proposed criterion and tire-road friction can be obtained and augmented with the existing dual-Kalman filter estimation algorithm, which improves tire-road friction estimation.

Copyright by
JIAN TANG
2022

ACKNOWLEDGEMENTS

Time flies and four years have passed. It has been a journey filled with tons of learning and adventures. I still remembered the vivid scenes when I arrived at the beautiful campus of the Michigan State University (MSU). I have been feeling so blessed for the valuable opportunity to be a student again. Now it comes to a new milestone in my life, in addition to the excitement, I want to express my gratitude to many people who supported, inspired, and encouraged me in my life, especially for this dissertation and my Ph.D. study, all along the way.

Foremost, my greatest gratitude shall be paid to my most respected advisor, Professor Guoming (George) Zhu. It is my great fortune to study and work under his persistent support, extraordinary guidance and caring. His comprehensive knowledge of control theory and extensive automotive application experience continually and convincingly motivated my Ph.D. research. There have been times when failures and loneliness have disheartened me, but his strong enthusiasm and unique wisdom toward life always help me to overcome the challenges and keep me going during this pandemic period. He set an excellent model that I want to follow for the future. I also want to express my appreciation to my advisor's wife, Mrs. Hongli Zhu, who always provides me caring and makes me feel as an extended family member.

I would like to express my most sincere appreciation to my Ph.D. committee members, Dr. Harold Schock, Dr. Vaibhav Srivastava, and Dr. Zhaojian Li, for their valuable feedback in the research work that has helped improve the overall quality of this work. They always provide me instructions and help with fast responses under their busy work schedule.

Also, I would like to thank Dr. Wen Dai from Ford Motor company, Dr. Hussein Dourra from Magna International, Mr. Yi Wang from FiTech. They always provide me lots of resources and guidance throughout the projects associated with engine and vehicle control, which help me to move my research in a correct and efficient direction. I would especially like to thank Mr. Kevin Moran, Mr. Thomas Stuecken for helping me in building the experimental engine test bench during the pandemic.

Besides, I would like to thank the MSU ERC (Engineering Research Court) community for

sharing their knowledge and expertise, which is enjoyable and memorable. Great appreciation shall be expressed to all the lab peers: Dr. Jie Yang, Dr. Ruitao Song, Dr. Yingxu Wang, Dr. Yifan Men, Dr. Ruixue Li, Dr. Tianyi He, Dr. Shen Qu, Dr. Anuj Pal, Dr. Wenpeng Wei, Mr. Corey Gamache, Mr. Lingyun Hua, Prof. Liang Liu, Prof. Xihong Zou, Dr. Dawei Hu, Dr. Cyrus Atis, Dr. Sadiyah Sabah Chowdhury, and Mr. Ilias Anagnostopoulos-Politis. In addition, I would like to thank Mr. Jisheng Chen, Ms. Ming Liu, and Mr. Xinda Qi for the friendship we had developed over the past few years. Also I would like to thank Dr. Eduardo Mondragon, Dr Jianming Gu, Mr. Joe Blaylock, Mr. Jim Lu, Mr. Kun Su, Mr. James Lynch, Mrs. Kimberly Lynch, and Mr. Greg Bales, who always shared with me kind suggestions and selfless supports for my life and study in US.

Last but not least, the gratitude from my deep heart shall be paid to my beloved family, without whom I would be nothing. Thanks to my beloved parents for their unreservedly support and understanding throughout this endeavor. I know that this is a proud moment for them too. Thanks to my darling and wisdom wife, for her sacrifices, immense love and support, without which I could not have achieved anything with a solid determination and faith. Special thanks to my cute son for abundant joys he brings to us, he is my most motivation and reshapes the direction of my life. I would like to share my happiness with them.

Time flies with a changing world, but my passion to the life and Spartan Wills will not. Go Green!

TABLE OF CONTENTS

CHAPTER 1	INTRODUCTION	1
CHAPTER 2	DATA-DRIVEN BASED MODELING AND BAYESIAN TRAINING ALGORITHM	19
CHAPTER 3	IMPLEMENTATION OF ALGORITHM THROUGH SIMULATION AND REAL TIME ENGINE BENCH TEST	34
CHAPTER 4	ONLINE UPDATING OF SURROGATE MODEL	61
CHAPTER 5	CYCLE-BASED LQG KNOCK REGULATION USING IDENTIFIED EXHAUST TEMPERATURE MODEL	81
CHAPTER 6	TIRE-ROAD FRICTION COEFFICIENT ESTIMATION	105
CHAPTER 7	CONCLUSION AND FUTURE WORK	124
BIBLIOGRAPHY		126

CHAPTER 1

INTRODUCTION

1.1 Motivation

Automotive control technologies play a significant role in the sustainable development of automotive industry. Many researchers have devoted themselves to the field of advanced automotive controls for a long time. Thus, a few classical approaches have been developed and implemented for estimation and control of engines, drivelines and vehicles [1]. With the gradual increment of utilizing machine learning and intelligent algorithms along with the year-to-year improvement of micro-controller hardware, data-driven methodologies have the potential of replacing and/or enhancing the performance of existing estimation and control algorithms at both vehicle subsystem (component) and system levels.

From the component level, the internal combustion (IC) engine is one of the key automotive powertrain components. IC engines have been widely used in automobiles since 1860s, superseding the steamed engines gradually [2]. As the world entered into the 21st century, electrical vehicle (EV) attracted more attention in the automotive industry around the world due to policies of slowing down climate change such as the latest Carbon-Neutral demand. However, several challenges still exist regarding EV's life span emissions and energy consumption, such as CO₂ emissions generated during battery production and end-life recycling, battery durability and safety. During this transition period, hybrid vehicles equipped with both battery-electrical-motor and internal combustion (IC) engine could become the mainstream powertrain in the 21 century [3] before all the technical barriers of EVs are overcome. In addition, many off-highway applications still require IC engines. Therefore, it is still valuable and necessary to improve IC engine efficiency to satisfy the ever-increasing stringent emission requirements. Various technological innovations for improving combustion efficiency continued, where some engine subsystems, such as exhaust gas recirculation (EGR), homogeneous charge compression ignition (HCCI), and turbocharger are used along with the associated control strategies. These technologies were integrated into products through both hardware and software changes, resulting in a significant improvement in combustion efficiency

over the past decades. Combustion process needs to be accurately controlled (or calibrated) to further improve its efficiency. A good prediction of associated parameters and performances will lead to an improved control. Ideally, the engine is calibrated to deliver maximum brake torque (MBT) for a given fuel flow rate. However, the biggest obstacle for an SI (spark-ignited) engine to achieve MBT is its knock tendency, which drives our research in this dissertation. Knock is an undesirable combustion phenomenon caused by spontaneous ignition of a significant portion of end gas in the unburned zone, which releases extremely rapid heat and generates high magnitude pressure waves that may damage the engine [2]. Therefore under certain operational conditions (mainly, at the region with low engine speed and high load), the engine optimum control parameters are limited by engine knock [4]. In this case, the control parameters (such as spark timing, intake and/or exhaust valve timing(s) [5], EGR) need to be optimized to achieve the highest combustion efficiency (best fuel economy) possible without violating the knock constraint, resulting in a multi-variable optimization problem under a given operational condition. That is, to prevent engine damage, instead of operating the engine at its MBT timing, the engine shall be operated as close to its borderline knock limit as possible. In other words, maintaining knock intensity (KI) stays below a desired level while maximize the combustion efficiency. Note that KI can be calculated based on in-cylinder pressure and knock sensor signals [6].

Traditionally, the borderline knock limit can be detected by the engine mapping process on an engine dynamometer experimentally, and the optimal solutions can be found based on mapped test results. But this process is heavily dependent on user experience to generate good test points, and it also requires a high number of function evaluations or tests for a complex system. Normally the EGR rate and valve timings are controlled in an open-loop and spark timing is controlled in a closed-loop based on cycle-to-cycle knock intensity signal(s) detected from accelerometer-based knock sensor(s) to avoid engine knock [7] when the engine is knock limited. Due to the stochastic nature of engine knock, the borderline knock limit is often very conservative, leading to a loss of certain combustion efficiency due to excessive spark timing retarded. On the other hand, as more and more engine control parameters are used, to predict (find) the borderline knock limit will

consume a lot of time and test budget by running sweeping tests.

In order to maximize the engine combustion efficiency within the safe operational region, stochastic characteristics or statistic information of knock combustion can be utilized for knock control. The knock intensity distribution and its tolerance level were utilized for stochastic knock control in [8] and map-learning-based control in [9] so that engine can be operated close to its knock borderline limit stably. However, both studies only use one control parameter, spark timing, for knock control without considering the effect of other control parameters and system variations. From the control perspective, various control-oriented combustion model were developed for validating combustion control strategies and/or estimating combustion dynamics in real-time. They can provide an estimation of burn rate, in-cylinder temperature and pressure with high computational efficiency, when the computational capability of microprocessors became feasible. The estimation capability of these models can also be utilized for knock borderline prediction.

A detailed survey was conducted to better understand the applications of control-oriented combustion models [10]. They can be divided into two categories: physics-based models and black-box models developed mainly based on system input and output correlations obtained from either simulations or experiments [11]. They are also referred as physical models and input-output models [12, 13]. The main distinction of these two model groups is the level of model transparency or the level of physical system knowledge used. The input-output model is also labeled as a black box one [14] since it is based entirely on the system inputs and its responses (outputs) without clear knowledge of system internal dynamics. The black box model is a pure data-driven based model because the model is expressed using an empirical function calibrated by the experimental data only. Some typical methods, such as polynomial assumption [15, 16], Gaussian correlation [17, 18], or low-order parameterized function [19], are described, respectively, in the modeling literature. Each model group has its own applicable scenarios. Among all these control-oriented models, it is worth to introduce that , a reaction-based control-oriented combustion model along with a pressure wave model [6, 20] was developed with a strong capability of predicting engine knock, but it still requires significant calibration efforts. Note that the black-box model is cheap to evaluate

and widely used in machine learning [16] recently to predict the steady-state characteristics of the engine combustion process within the operating range of training data set. With the advancement and widespread adoption of machine learning methods for control applications, it is now possible to use a black-box model with its intelligence to efficiently predict knock borderline and develop online control based on it without the knowledge of system dynamics. This is one of the main contributions in this research.

On the other hand, the knock intensity varies cycle-to-cycle even over a fixed operational condition with fixed control parameters. The statistic-based algorithms discussed above do not provide any immediate control action to reduce cyclic variations caused by the inherent combustion dynamics. Therefore, the recommended optimal baseline control parameters are set for 'averaged' cycle behavior and are relatively conservative. The cyclic combustion variations are caused by many factors [2] such as variations in gas motion, amount, composition, and temperature of gas and fuel mixture, and engine design (combustion chamber shape).

Some detailed investigations are conducted in reference [21]. The percentages of knock and cycle-to-cycle variations under different intake temperature levels are studied using extensive experimental data, which shows that high temperature at intake valve closing increases the knock occurrence percentage and significantly influence the knock cycle-to-cycle variability. These variations also reflect different burn-rates in the combustion process which is important since they are correlated with engine torque and performance. Note that normally the fastest burning cycle determines the engine compression ratio limit; while the slowest one affects the most retarded spark timing. Under certain engine operational conditions such as operated close to borderline knock, the combustion variations are much higher due to the high-frequency shock and pressure waves in the combustion chamber caused by uneven heat distribution that have been visualized in reference [22] using high-speed camera. The visualization results indicate that strong shock waves exist in knock combustion. The consequence of these large variations is more significant when engine is run close to its knock borderline. For example, the faster burning cycles have substantially higher values of peak cylinder pressure than these with slower burning cycles; and with faster burning cycles, peak

pressure occurs earlier, and closer to top dead center, most-likely leading to knock combustion. The simulation-based study in reference [20] provides a new insight about cycle-to-cycle varying knock characteristics using reaction-based combustion model and pressure wave model [20]. It shows strong correlations among current-cycle exhaust valve open (EVO) temperature, next cycle intake valve closing (IVC) temperature and knock intensity. It can be explained as follows: the high current-cycle intake manifold temperature will lead to heavy knock, which leads to short combustion duration with high combustion temperature and large heat loss to the wall and results in low trapped gas temperature; and as a result, low trapped gas temperature would consequently reduce the in-cylinder mixture temperature at IVC of the next-cycle, which further influences the next-cycle knock intensity. An experimental research [23] confirms knock prediction using exhaust gas temperature. It shows that when the engine has knock combustion, the exhaust gas temperature decreases considerably. The heavier knock will cause higher heat loss than that of normal combustion. The main reasons for this decrement are that knock combustion is relatively short and knock pressure wave changes the thermal exchanges on the cylinder wall. These early results inspire our work about model-based prediction of next cycle knock intensity based on the measured combustion parameters (such as exhaust temperature) at the current-cycle.

With the gradual adoption of machine learning and intelligent algorithms, data-driven methodology has demonstrated its potential of enhancing the estimation and control performances. Consistent progress has been made to improve the vehicle system estimation and control [24, 25, 26, 27, 28], which plays a significant role in improving automotive performance [1]. Note that the controlled tire traction torque determines the vehicle dynamic performance [29] and is heavily dependent on the tire-road interaction [30, 31]. Therefore, an accurate estimation of road condition in terms of tire-road friction coefficient is a key factor for optimizing vehicle performance such as reducing the skid, delivering maximum available torque, and maintaining vehicle stability.

Road condition can be represented by normalized traction force (also called the coefficient of traction) [32, 33]. It is defined as the ratio of longitudinal and normal forces exerted on the wheel

when only the longitudinal vehicle dynamics is considered.

$$\rho = \frac{F_x}{F_z} \quad (1.1)$$

where F_x and F_z are the longitudinal and normal forces, respectively. The maximum value of the normalized traction force needs to be estimated in real-time for an advanced accurate traction force control. This maximum value, namely the tire-road friction coefficient μ [32], is an index representing the road surface characteristic, such as icy, snow, wet and dry. Accordingly, the value of it varies between zero and one for different road conditions [32]. It is important to have a better understanding of the road friction coefficient so that both the vehicle active and passive safety [32, 34] associated with traction performance can be further enhanced. For example, a correct real-time estimation of the tire-road friction coefficient would allow an anti-lock brake system to start braking with the optimal brake pressure which will provide an efficient way to stop the vehicle at a shorter distance. Also it will be useful to control the torque profile of the engine and/or motor to deliver the maximum allowed torque for the best tracking performance.

For estimating the tire-road friction coefficient, several approaches have been discussed and reviewed in literature [32, 34, 35, 36, 37], and are summarized below briefly. The estimation methods can be mainly divided into two approaches: 'cause-based' and 'effect based'. The cause-based approach uses special sensors (for example, optical/image sensor for detecting road surface, accelerometer for tire vibration) to identify friction related condition such as dry, wet, snow, or ice road based on the correlation between sensor information and road condition. Note that changes of estimation parameters are resulted by the cause of corresponding tire-road friction coefficient. It is able to estimate road condition even when the vehicle is operated under a constant speed, but it often fails under certain conditions, for instance, with bad quality of image. Meanwhile, the required sensors, such as optical sensor(s) or camera(s), are generally expensive and not available for most of production vehicles.

The effect-based (also called model-based) approach focuses on response behaviour excited by the tire-road friction, which can be divided into three groups: a) wheel and vehicle dynamic based, b) tire model-based, and c) slip-slope based. For the wheel and vehicle dynamic based approach,

vehicle motion dynamics is utilized to estimate friction coefficient, where some vehicle states can be acquired directly from sensors, while some need to be estimated using different estimation algorithms (recursive Least Squares, extended Kalman filter (EKF)) [30]. Tire models [36] are widely used to estimate tire forces after model parameters are calibrated. Based on the measured/estimated tire force, the maximum tire road friction coefficient in the tire model can be solved. The slip-slope based method assumes linear relationship within the small slip-ratio region. For all these models, accurate vehicle velocity is essential to calculate the slip-ratio, which may require additional high-accuracy GPS or other vehicle speed sensors. However, in the production vehicles, the velocity is often derived from these sensors indirectly measuring the vehicle speed such as wheel speed signals and it is often not accurate enough for precise slip-ratio calculation, leading to inaccurate estimation of friction coefficient. In summary, the 'cause-based' approach is not practical for production vehicles due to requirement of additional sensor(s), and the 'effect-based' approach is more attractive.

Based on all the discussion above, this dissertation focuses on investigating new methods to address these concerns: 1) to improve the engine efficiency by predicting knock borderline and cycle-to-cycle variations; 2) strengthen the traction performance by estimating the road friction coefficient. A data-driven methodology is going to be introduced for two different application scenarios. Based on different applications, it will either replace or assist the current predictions obtained by the existing algorithms; see Figure 1.1. The detailed approaches are introduced in the next section.

1.2 Research Approaches Using Data-driven Based Algorithm

In the last section, motivations of this dissertation are reviewed regarding the importance of estimation and control for automotive systems in the rapid development era of data-driven learning algorithms. The specific research topics about estimations in both engine and vehicle areas are discussed with their limitations highlighted. Two different approaches using the data-driven algorithm are proposed for two applications: engine borderline knock detection and vehicle tire-road friction estimation.

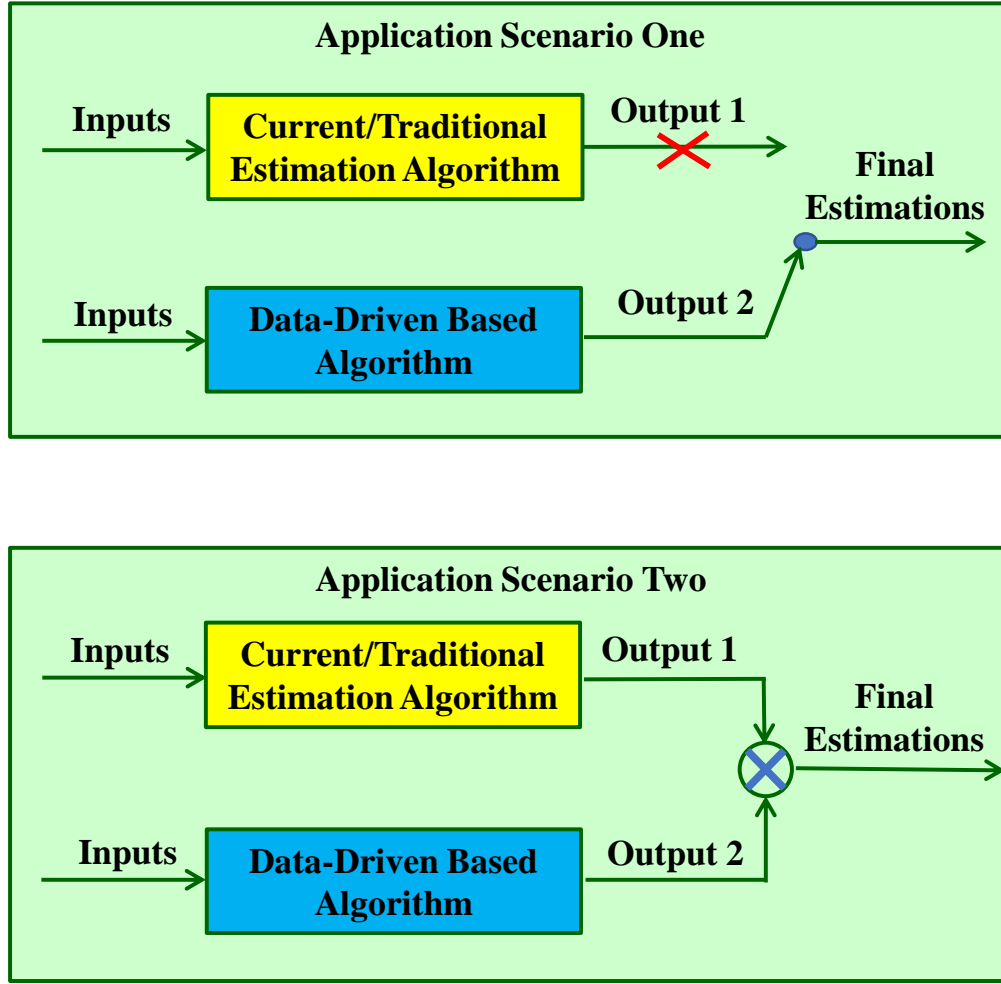


Figure 1.1 Application scenarios using a proposed data-driven estimation algorithm

1.2.1 Scenario One: Engine Borderline Knock Detection

This subsection focuses on engine borderline knock prediction and cycle-wised variation control for spark ignited engines. As introduced, the majority of existing work focuses on one control parameter for knock borderline limit management with massive calibration efforts and does not consider short-term and long-term environmental and system variations. Especially, there is no control scheme developed for real-time cyclic combustion variation compensation. In other words, the proposed algorithm needs to address three main issues for knock borderline control. First is to significantly reduce the expensive engine test budget used to solve the multi-dimensional optimization problem; second is to compensate the possible environmental changes and engine aging after offline training; the last (third) is to reduce cycle-to-cycle variations caused by the inherent combustion dynamics. To

address these three challenges, the proposed borderline knock prediction and control architecture is divided into three parts as shown in Figure 1.2. Note that the shaded-blue block, the first part, is the offline training process for predicting the knock borderline limit under multiple control parameters utilizing the stochastic knock profile knock. An intelligent Bayesian optimization method is used and assisted by the statistic models trained with the paired data set; and the shaded-green block, the second part, is used for online updating process to compensate the environmental changes and engine aging utilizing the trained Kriging models; finally, the shaded-red block, the third part, is used for reducing the cycle-wised variation by developing a cycle-based knock intensity model through data-driven system identification and model-based control for cycle-to-cycle deviation of spark timing, which is complement to the 'averaged' cycle behavior. The detailed algorithm for each part will be introduced in the corresponding chapter.

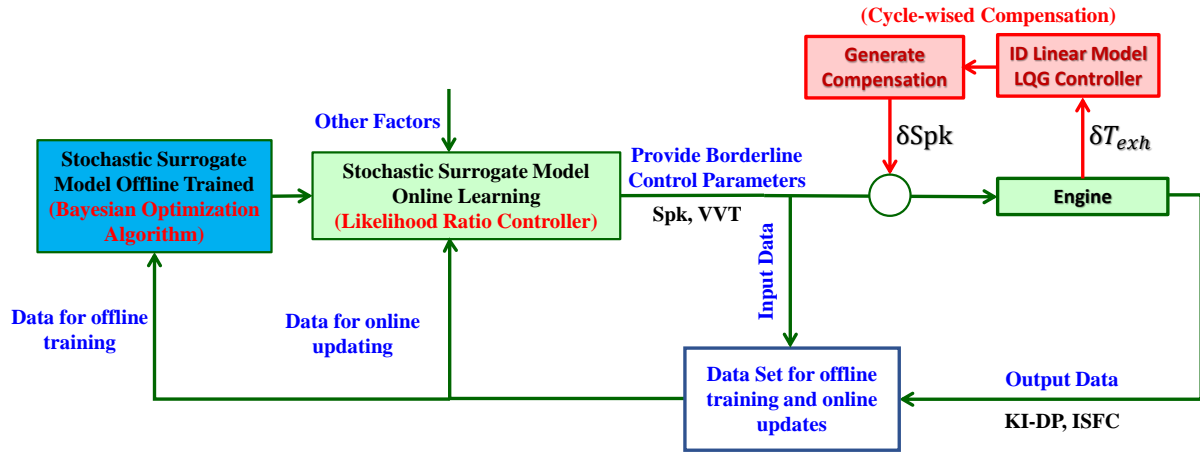


Figure 1.2 Overall architecture of proposed knock control algorithm

For the offline training process in this research, it focuses on minimizing two competing objectives, knock intensity (KI) and indicated specific fuel consumption (ISFC), with two control parameters (spark and intake valve timings). Solving multi-objective optimization problems has a long history and many distinguished methods have been developed [38], where one of the most successful methodologies is to use a surrogate-model (or meta-model), developed based on high fidelity simulation (or experimental) evaluations [39], to approximate the objective function. In this dissertation, a surrogate-model was adopted to assist the entire Bayesian optimization process,

which is called surrogate-model assisted optimization (SMAO). The whole optimization process starts from a supervised training-based surrogate-model development process using an initial data set, and then, the maximum expected improvement of the next sample point [40] is estimated by actively learning the cost landscape from all previously tried points to improve (optimize) the model accuracy and/or to obtain an improved solution. The uncertainty of the next sample point is predicted by the assumption of Gaussian distribution. The active learning process is sequentially run until the convergence criterion or computational budget is met. The high potential efficiency of SMAO process has been validated in terms of low cost and fast convergence by using classical test problems [41] and SMAO has been used for different applications such as groundwater reactive transport [42] and aerodynamic problems [43] with limited applications to the automotive industry. Recently, Pal et al. [44, 45] successfully optimized the diesel engine calibration process using the SMAO method.

There is no existing work, to the best of our knowledge, implementing the stochastic Bayesian optimization methodology for engine knock prediction and control. The capability of using Kriging model to predict mean and variance inspires us to apply this method to knock prediction and control. It is expected that, after the SMAO, a tradeoff (Pareto front) between two competing objectives, knock intensity (KI) and fuel economy, will be generated, as shown in Figure 1.3. Note that when the engine MBT spark timing is limited by knock combustion in the advanced direction after all control parameters are optimized, the higher the KI is, the higher the combustion efficiency (low ISFC). Hence, a tradeoff, called Pareto front [46], between KI and ISFC can be obtained by non-dominant sorting after Bayesian optimization. The offline trained Kriging model is a Gaussian process model providing both mean and variance, so the further decision can be made by evaluating the mean Pareto front and associated variances along the front (see Figure 1.3); The region enclosed by lower and upper boundaries can be formed using predicted mean and variance of Kriging model, which corresponds to the 99.7% confidence range of log-normal distribution using the empirical three-sigma rule in statistics. The knock borderline considering the given allowed KI level (see desired KI level in Figure 1.3) can be predicted as the solid-dot along the mean Pareto front in Figure 1.3.

This point can be projected back to the design space to find the optimal control parameters in the design space with the engine KI staying below the desired KI (in this study, 1 bar). This makes it possible to operate the engine below the desired knock limit by using the recommended control parameters (spark and intake valve timings) with minimal sacrifice of engine combustion efficiency. On the other hand, the obtained upbound varies due to the engine environment changes as shown in the shaded yellow region of Figure 1.3, leading to the adaptive change of corresponding optimum design sites.

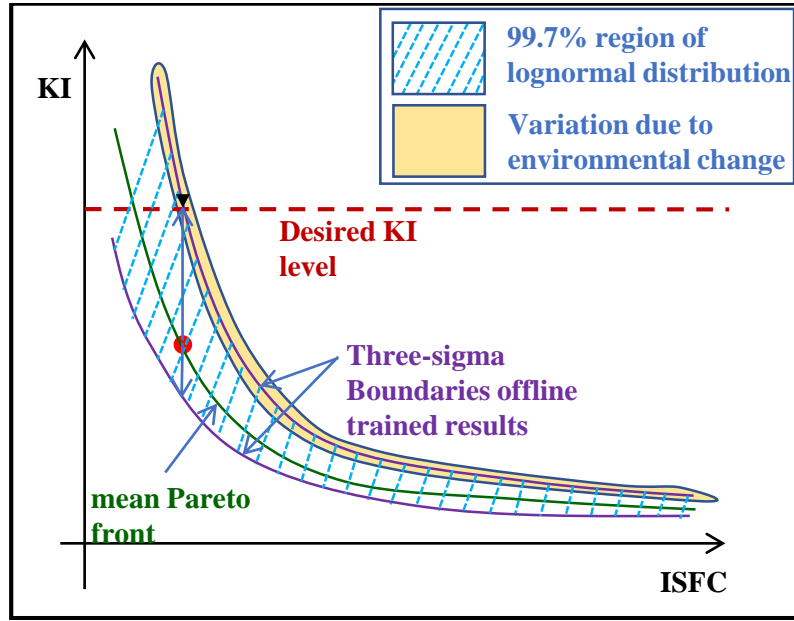


Figure 1.3 Pareto front of KI and ISFC

Since the data-driven black-box model is extremely cheap to evaluate, it can be updated in real-time to compensate for engine aging and operational environmental changes such as fuel type, temperature, humidity, etc. The proposed online update process (the second part) integrates the off-line trained surrogate models using a modified likelihood controller. The relationships between offline trained surrogate models and online updating process is shown in Figure 1.4.

A principal component analysis (PCA) was conducted for the data points along the Pareto front. PCA results indicated that spark timing is the most sensitive factor affecting the Pareto front used to determine the borderline knock limit. A likelihood ratio controller with two buffers was proposed for updating the surrogate model. Offline trained surrogate model provides a Pareto front

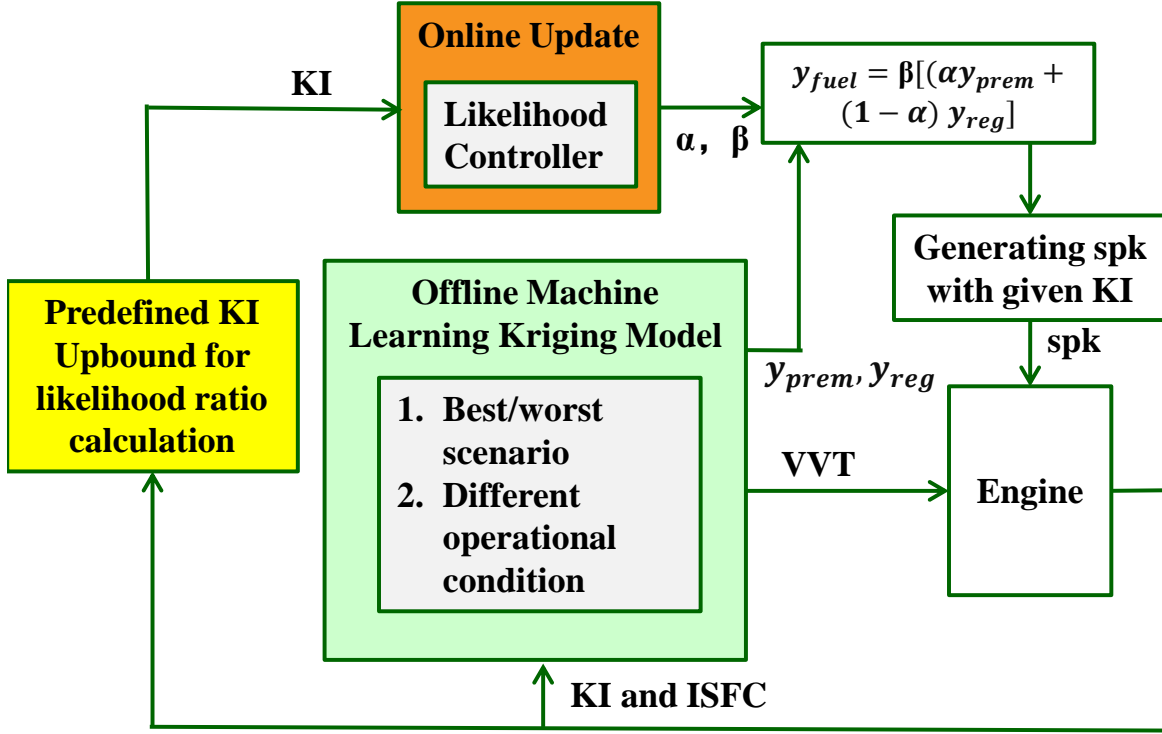


Figure 1.4 Online updating structure

as the baseline when engine is operated at knock limited condition. Next, the new optimum knock borderline intersection can be located by comparing the updated KI upbound with the given KI limit after mapping the updated KI upbound from Gaussian back to log-nominal distribution.

Further advance the spark timing for improved fuel economy over the traditional knock control logic can be achieved by reducing the combustion cycle-to-cycle variations (the third part). The developed control strategy reduces the knock intensity variations while maintaining the knock intensity (KI) below a desired level for spark ignition engines when it is operated close to or at knock borderline. The PRBS (pseudo-random binary sequence) q -Markov COVER (COVariance Equivalent Realization) system identification was used in this dissertation to obtain linearized model from δ spark timing to δ exhaust temperature and KI. This method has been shown to work well for different applications with complex nonlinear dynamics [47, 48, 49]. A 4th order linear system model from deviated spark timing to deviate exhaust temperature and KI was identified after using PRBS as spark timing excitation. Note that the identified model captures the dynamic tendency of δ exhaust temperature and KI excited by δ spark timing, and an LQG (linear quadratic Gaussian)

controller was designed based on the identified model accordingly for reducing KI variations using δ exhaust temperature as feedback. By adding this compensation loop into the entire knock control architecture (shown in Figure 1.2), the baseline spark timing, obtained through offline trained stochastic surrogate model [50] and online updating with likelihood ratio controller, can be further optimized.

1.2.2 Scenario Two: Vehicle Tire-road Friction Coefficient Estimation

For vehicle applications, this dissertation covers the tire-road friction coefficient estimation assisted by the data-driven model. As discussed, all the existing estimation algorithms have their own limitations, including but are not limited to the following aspects: 1) adopting empirical model with intensive calibration, 2) requiring additional sensors with added cost to production vehicles, 3) not guaranteeing slip-ratio estimation accuracy due to the error of vehicle speed measurement, and 4) low estimation accuracy due to not enough excitation. The proposed algorithm is expected to only utilize the data from available production sensors with system and measurement noises. A new stochastic-based evaluation criterion based on existing vehicle wheel shaft sensor signals with the help of data-driven Kriging model is proposed, along with a sequential extended Kalman filter (S-EKF) based on a vehicle dynamic model (an existing model-based approach), to form an estimation fusion structure as shown in Figure 1.5.

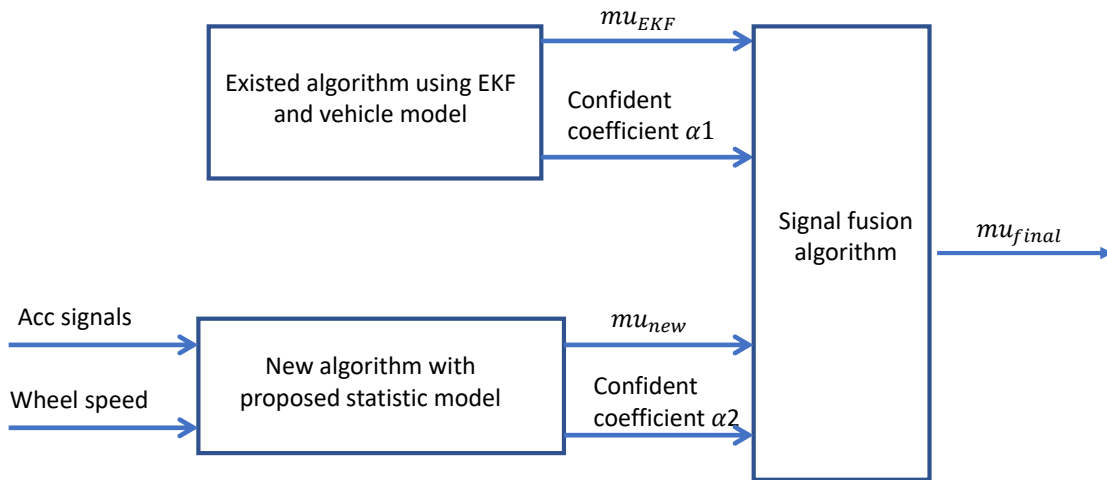


Figure 1.5 Structure of proposed algorithm for tire-road friction coefficient estimation

As demonstrated in Figure 1.5, two estimation algorithms work in parallel and independently.

A classical vehicle dynamic model with extend Kalman filter (EKF) algorithm is used to estimate friction coefficient (the top path); and the proposed algorithm only uses the vehicle acceleration and wheel speed signals, available for almost all current production vehicles to train a statistic model and provide a new estimation (the lower path). More details can be referred in Chapter 6. The proposed data driven model requires much less calibration efforts, comparing with the EKF approach. Unlike other model-based approaches, this statistic model does not require any physical model. To integrate two predictions, a weighted algorithm is used to obtain the final predicted friction coefficient; see Equation (1.2)

$$\mu_{final} = (\alpha_1 * \mu_{EKF} + \alpha_2 * \mu_{new}) / (\alpha_1 + \alpha_2) \quad (1.2)$$

where α_1 and α_2 are two confident level coefficients assigned to each algorithm based on a certain predefined logic, respectively. The proposed estimation method is verified using both CarSimTM and vehicle test data in Chapter 6.

This section provides an entire picture of applying data-driven model-based algorithms to automotive estimation and control in our research. First one reduces the existing exhaustive calibration efforts required for the engine knock borderline prediction with multiple control parameters, where a supervised-learning based Bayesian optimization method, assisted by data-driven based surrogate model, is proposed and Kriging model is used due to its ability to predict statistic features of the objectives including both mean and variance. This surrogate model and offline training results are continually integrated with online updating and cycle-to-cycle control scheme. Second application is to offset the existing algorithm for predicting tire-road friction by using a data-driven model along with a novel evaluation criterion for tire-road friction estimation based on statistic information of vehicle acceleration and wheel speed signals. This data-driven based black-box model needs little calibration efforts as well.

1.3 Contributions

Overall, the main contributions of research work conducted in this dissertation are applying the data-driven model-based approaches to classical automotive estimation and control problems, especially for engine knock borderline control and tire-road friction estimation. The statistical

characteristics of the associated systems are well studied by considering system and measurement noises. The proposed prediction and control algorithms improve estimation and control efficiency and their robustness to the operational environment and system aging. The major contributions of this dissertation can be summarized as below.

- 1) The Bayesian learning algorithm is modified to predict engine knock borderline along with associated optimized control variables. The main modifications include a dual-surrogate-model structure and log-normal to Gaussian distribution mapping. A dual-surrogate architecture is proposed to obtain stochastic information of Kriging model with unknown noise characteristics that is true for real-world applications; and a distribution mapping is proposed to implement the transformation between actual system log-normal KI distribution and Gaussian one assumed in the Bayesian optimization algorithm.
- 2) The entire learning algorithm is implemented into a physical engine test bench for performing experimental optimization with actual system and measurement noises. Experimental results indicate that the active learning logic, embedded in this algorithm, efficiently predicts the optimum operational conditions with a significantly reduced test budget, compared with more than 200 tests used in the conventional sweeping process. Using the Bayesian learning algorithm, a mean Pareto front can be generated along with its upper and lower boundaries obtained based on estimated variances and three sigma rules statistically. By using the final trained surrogate model, a smooth Pareto front can be found. Compared with a predefined desired knock limit, control variables in the design space can be found with the best engine efficiency possible. This provides a new perspective about the knock borderline prediction and control.
- 3) An online surrogate-model updating scheme is proposed using the offline learned Kriging models based on a principle component analysis result beforehand. A likelihood controller with two different buffers is integrated into the model updating structure to compensate the knock intensity variations caused by both the short- term and long-terms effects, respectively. The proposed scheme is verified through both co-simulations and tests on an actual engine.

The results confirm the effective compensation with a relatively advanced averaged spark timing.

- 4) The properties of pseudo random binary signals (PRBS) and q Markov COVER are utilized to identify a linear control design model with δ spark timing as input and δ exhaust temperature and KI as outputs. Accordingly, an LQG regulator is designed to reduce KI cycle-to-cycle variations based on measured δ exhaust temperature. This compensation strategy is successfully integrated into the complete knock control architecture, and implemented onto the engine bench test. The validation results indicate that the LQG cycle-wised control with exhaust temperature feedback is able to reduce KI variation by 28.6%, which is able to further advance the spark timing for improved fuel economy.
- 5) A new evaluation criterion associated with the derivative of slip-ratio is introduced. The proposed new criterion is fully investigated through the simulation studies and experimental validation will follow. Both simulation and test results confirm the capability of proposed new criterion for friction coefficient estimation. With this new criterion, a statistic model could be trained when more test data is available. Finally, it is combined with the traditional estimation method to further improve the prediction accuracy of tire-road friction coefficient.

1.4 Dissertation Structure

This complete dissertation is organized as shown in Figure 1.6. Chapter 2 starts with a basic introduction of data statistic characteristics, and then provides a detailed discussion about the development of stochastic and deterministic Kriging models trained by input-output paired data. Finally, a Bayesian optimization methodology along with the acquisition function selected for the stochastic scenario with the unknown noise level and the proposed distribution mapping for this specific application to engine knock borderline prediction.

Chapter 3 presents the first application scenario about using the data driven models during the prediction process using both simulation and actual engine test results. The simulation study of proposed training algorithm is developed before applying to the actual engine test bench. The simulation studies were run using two selected test problems and one reaction-based combustion

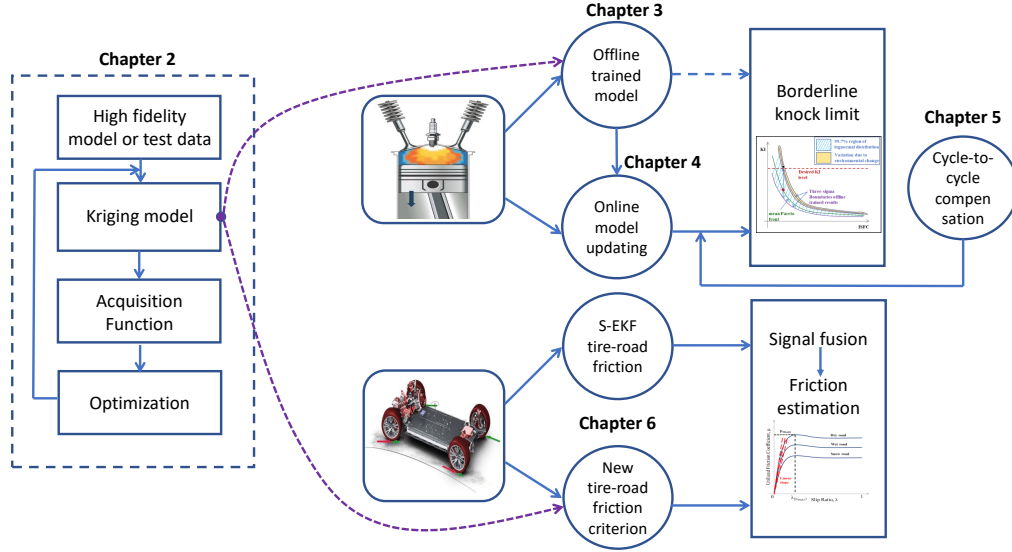


Figure 1.6 Structure of dissertation

model. A detailed introduction about the hardware and software configurations is introduced and followed with a sweeping test for comparison purpose in Chapter 3. Two different operational conditions are selected and 40 iterative training process is defined. After analyzing the experimental data from the engine test, the corresponding response surfaces (Kriging models) are presented. Due to the simplicity of obtained surrogate-model, it can be further updated for online compensations.

Chapter 4 discusses the extension of utilizing the offline trained surrogate models and the associated online updating structure. The entire chapter describes the likelihood ratio control strategy and the corresponding control scheme using two buffers used for this specific application. The simulation studies for the proposed online update algorithm are conducted in a LabView and Simulink co-simulation environment. After discussing the simulation results, the engine experimental validation results are presented under different scenarios.

Chapter 5 provides a brief review about the PRBS q -Markov COVER system identification and LQG control and the high pass-filter design of exhaust temperature signal for real-time sampling of δ temperature, then a preliminary test for studying the properties of exhaust temperature and associated δ one is presented. At last, the system identification results are studied with designed open-loop test and the identified model is verified using impulse test data, followed by the cycle-to-cycle compensation strategy implemented to the physical engine with and without the online

updating structure.

The second application of data-driven algorithm to assist estimation of tire-road friction coefficient is introduced in Chapter 6. A review of the existing estimation algorithm is provided first, followed by a discussion of associated drawbacks. A sequential EKF model with the slope method is verified through the CarSim simulations. Simulation results match with estimation of associated parameters based on simulated actual vehicle speed using carefully calibrated models. However, this algorithm comes up with some issues when it is applied to test data without accurate vehicle speed measurement. A new criterion for estimating tire-road friction is proposed in detail with verification using CarSim simulation data. By using both simulation and vehicle test data, the proposed criterion provides a strong correlation to tire-road friction coefficient.

Finally, Chapter 7 provides conclusions of proposed research, along with recommendations for future study.

CHAPTER 2

DATA-DRIVEN BASED MODELING AND BAYESIAN TRAINING ALGORITHM

The last chapter portrayed the whole picture of proposed research. It is essential to introduce the basic statistic features behind the proposed data-driven algorithms before final results are presented, especially for those modifications proposed in the dissertation.

This chapter starts with basic concepts of statistic characteristics and detailed derivation of deterministic and stochastic Kriging models. Note that a Kriging model is not only a data-driven statistic process model, but also the key part of the intelligent iterative SMAO algorithm. Besides the Kriging model, a high fidelity model and an acquisition function constitute the whole SMAO algorithm in a closed-loop. The details of proposed algorithm, the motivation of using dual surrogate model structure, and distribution mapping are further discussed for our specific application. These theories with modifications form a foundation for implementing the algorithm presented in the next chapter.

2.1 Data Statistic Features

Note that data is the core for data-driven based algorithm. The system dynamics and measurement noises are the sources of uncertainties or variations of the sampled data. Statistics provide tools to understand and interpret the source of variations embedded in the empirical data. A few basic concepts will be used throughout the rest of dissertation.

Probability is a mathematical property used to describe uncertain events and plays a key role in statistics. Probability is a number used to describe how likely these specific events will occur for a given experiment or input. The distribution of probability is described by the probability density function (PDF). In a more precise sense, the PDF is used to specify the probability of a random event falling within a particular range of values. For example, if the engine runs for 100 cycles at a fixed operational condition, the knock intensity (KI) of each cycle is distributed across a band with its possible outcomes due to the cycle-to-cycle stochastic characteristic of combustion. The KI distribution is shown with several bars in the histogram plot of Figure 2.1, where each bar represents the relative number of observations within the bar range. Note that the sum of bar areas

is equal to one. Since the width of bin is the same, the higher the bar is, the larger possibility of a KI event stays within the bin.

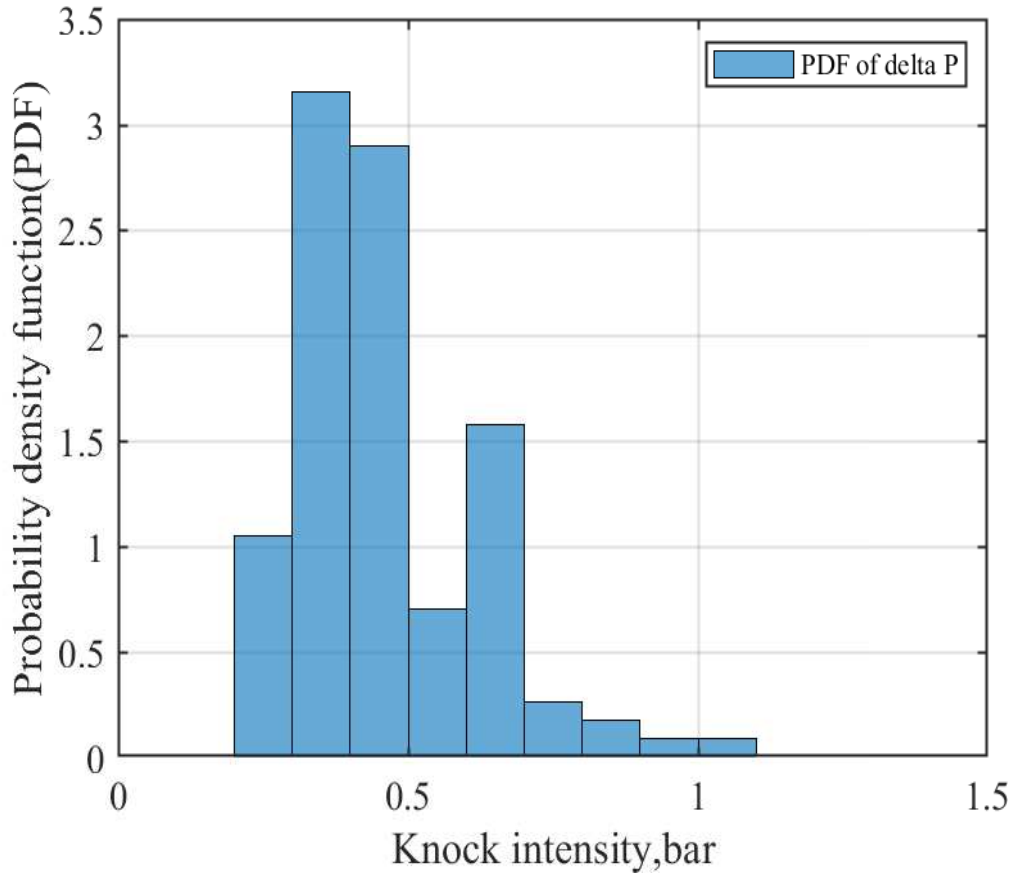


Figure 2.1 Knock intensity (bar) PDF

The data distribution information is important for data driven based algorithms, since it reveals the statistic tendency out of 'random' surroundings. There are two important indexes regarding the central tendency and dispersion of a probability distribution in statistics: mean and variance. In this dissertation, 'mean' is referred to the concept of arithmetic mean, which is the average of collected data. On the other hand, the variance measures the level of deviations away from the mean value for all sampled data, which is important information for data mining, machine learning, and so on. The variance is defined as the mean of squared differences between mean and sampled data in the distribution. Note that square root of variance is the standard deviation. In statistics, the empirical 3-sigma rule means that nearly all values stay within three standard deviations of mean, and thus it is empirically useful to treat 99.7% probability as the confidence level. For a

stochastic process, a well-trained statistic model is expected to predict both the mean and variance of unknown parameters. By utilizing the 3-sigma rule of thumb, the final estimation and control with a 99.7% confidence level can be decided within a safe zone.

2.2 Kriging Model Development

A surrogate-model is simple to evaluate since it does not include the physical dynamics that maps from inputs to outputs. How to train a black-box surrogate-model based on a collection of input-output data, obtained from either simulating a detailed physical model or conducting experiments on a physical system, is reviewed in reference [42]. Among these methods, Kriging (Gaussian Process) model has been widely-used recently since it predicts not only mean value but also variance that includes both the intrinsic uncertainty inherent in a stochastic process and extrinsic uncertainty of the unknown response surface [51].

The standard Kriging model is deterministic as it only includes model uncertainty, also called extrinsic uncertainty, and does not consider system intrinsic noises. It is an interpolation model based on the Gaussian process governed by prior covariances, and can provide the best unbiased prediction of mean and normally distributed variance for unsampled locations. Several literature have discussed the Kriging modeling in detail [52] with few modified for stochastic processes [53, 51]. In a stochastic Kriging model, by allowing the Kriging model to regress the data instead of interpolating, some researchers have extended the original deterministic approach to a broad one by adding a regression constant, also called regularization constant, to the leading diagonal of the Kriging correlation matrix R_d [53], which is shown in Equation (2.3) and will be explained in the model development part later. For its application to knock prediction, a stochastic Kriging model is needed due to the cycle-to-cycle knock variation caused by inherent stochastic combustion characteristics. Both deterministic and stochastic Kriging models are introduced in this dissertation and modeled based on the extension of Design and Analysis of Computer Experiments (DACE) Matlab toolbox by Nielsen [54] and the extension SK code package developed by Nelson's group [51]. For our application to knock borderline prediction and the selected acquisition function, both deterministic and stochastic surrogate models are needed. The reasons of using both models in this research will

be explained in the next section.

Basically, a scalar stochastic Kriging model includes three terms (see Equation (2.1)), where the first term, $\mathbf{f}^T(\mathbf{x})\boldsymbol{\beta}$, is a function of input vector \mathbf{x} to capture the general trend; the second, $z(\mathbf{x})$, is a realization of a zero mean random field so that the response of unknown design sites is modeled within a statistical framework; and the third, the last term, $\epsilon(\mathbf{x})$, is introduced for system and measurement noise consideration to model the nature of stochastic processes. The probabilistic form of a scalar response surface model is shown in Equation (2.1).

$$y(\mathbf{x}) = \mathbf{f}^T(\mathbf{x})\boldsymbol{\beta} + z(\mathbf{x}) + \epsilon(\mathbf{x}) \quad (2.1)$$

where \mathbf{x} is a p dimensional vector of design variables; $\mathbf{f}(\mathbf{x})$ is a q dimensional vector of known functions; $\boldsymbol{\beta}$ is a constant parameter vector to be determined with compatible dimension; $z(\mathbf{x})$ is a statistical model of a random process with zero mean and covariance based on a spatial correlation function between two design points \mathbf{x}^i and \mathbf{x}^j , where $i, j \in [1, n]$ and n is a total number of evaluation points; $\epsilon(\mathbf{x})$ is an intrinsic noise that is assumed to be Gaussian with its variance $\lambda^2(\mathbf{x}) = \text{Var}(\epsilon(\mathbf{x}))$, and it is also called as the regularization constant.

Without noise term $\epsilon(\mathbf{x})$, the expression for elements of n sample covariance matrix R_n can be expressed in Equation (2.2) below.

$$[R_n(i, j)] = [\text{Cov}\{z(\mathbf{x}^i), z(\mathbf{x}^j)\}] = \sigma^2 R_d(\theta, \mathbf{x}^i, \mathbf{x}^j) \quad (2.2)$$

where σ^2 is the variance of process to be determined, R_d is an $n \times n$ correlation function matrix based on the spatial distance for all pairs \mathbf{x}^i and \mathbf{x}^j ($i, j \in [1, n]$). For Kriging model, R_d is a Gaussian correlation function and can be calculated using Equation (2.3) based on the spatial distance. The correlation between points decreases as their distance increases.

$$[R_d(i, j)] = [\text{Cor } z(\mathbf{x}^i), z(\mathbf{x}^j)] = [\exp(-\sum_{l=1}^p \theta_l |x_l^i - x_l^j|^2)] \quad (2.3)$$

where θ_l and x_l^i are the l^{th} elements of parameter vector $\boldsymbol{\theta}$ associated with the width of Gaussian function and design variable vector \mathbf{x}^i , respectively. The effect of parameter vector $\boldsymbol{\theta}$ can be thought of determining how quickly the function changes as \mathbf{x}^j moves away from \mathbf{x}^i . A large θ_l implies a significant activity or low correlation in dimension l . From this, with the known input-output data set D formed by n input points, an $n \times n$ covariance matrix $R_n = \sigma^2 \mathbf{R}_d$ can be constructed. For

a stochastic case with independent noises, the corresponding covariance matrix \mathbf{R}_s is obtained by adding a regulation constant to the diagonal components such that $\mathbf{R}_s = \sigma^2 \mathbf{R}_d + \lambda^2 \mathbf{I}$. With all these, the model has three unknowns to be solved for the Kriging model and they are scalar σ , parameter vectors $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$. They are determined by maximizing the likelihood function over the observed points. To simplify the likelihood maximization, normally the natural logarithm will be taken to have

$$\ln L(\sigma, \boldsymbol{\beta}, \boldsymbol{\theta}) = -\frac{1}{2} \left[n \ln(2\pi\sigma^2) + \ln(\det(\mathbf{R}_s)) + (\mathbf{Y} - \mathbf{F}\boldsymbol{\beta})^T \mathbf{R}_s^{-1} (\mathbf{Y} - \mathbf{F}\boldsymbol{\beta}) / \sigma^2 \right] \quad (2.4)$$

where \mathbf{R}_s is a covariance matrix for the observed points; and $\mathbf{Y} = [y(\mathbf{x}^1), \dots, y(\mathbf{x}^N)]^T$ and $\mathbf{F} = [\mathbf{f}^T(\mathbf{x}^1), \dots, \mathbf{f}^T(\mathbf{x}^N)]^T$ are vectors of n observed function responses and vector function matrix evaluated at n design vectors, respectively. After the likelihood function is maximized, the stochastic Kriging model is obtained. Accordingly, the predicted mean μ and mean square error s^2 for $y(\mathbf{x})$ are predicted by the following Equations:

$$\mu_y(\mathbf{x}) = \mathbf{f}(\mathbf{x})^T \boldsymbol{\beta} + \mathbf{r}^T(\mathbf{x}) \mathbf{R}_s^{-1} (\mathbf{Y} - \mathbf{F}\boldsymbol{\beta}) \quad (2.5)$$

$$s_y^2(\mathbf{x}) = \sigma^2 + \mathbf{u}^T(\mathbf{x}) \left(\mathbf{F}^T \mathbf{R}_s^{-1} \mathbf{F} \right)^{-1} \mathbf{u}(\mathbf{x}) - \mathbf{r}^T(\mathbf{x}) \mathbf{R}_s^{-1} \mathbf{r}(\mathbf{x}) \quad (2.6)$$

where $\mathbf{r}^T(\mathbf{x}) = \sigma^2 [R_d(\mathbf{x}, \mathbf{x}^1), \dots, R_d(\mathbf{x}, \mathbf{x}^N)]$ and $\mathbf{u}(\mathbf{x}) = \mathbf{f}(\mathbf{x}) - \mathbf{F}^T \mathbf{R}_s^{-1} \mathbf{r}(\mathbf{x})$. Note that the exact stochastic property of $z(\mathbf{x})$ is not modeled in this process, instead, the stochastic property of $y(\mathbf{x})$, required for optimization, is. The general structure of deterministic and stochastic Kriging models are summarized in Figure 2.2. For the deterministic case, variance λ^2 is set to zero to obtain predicted mean and mean square errors, where each point provides an exact correlation with itself, forcing the predictor to interpolate evaluated points. Detailed derivative is presented in reference [52]. Note that existing research has shown that the regression function used in the Kriging model has less impact on the model performance [55]. People usually start with 0-th order regression function (which is a constant mean), assuming that there is not any knowledge of the system behavior. However, the zero-order function in the ordinary Kriging model was proven to be erroneous for certain nonlinear applications [56]. When the regression function is

believed to have a trend or some application-specific parametric structure, a polynomial regression function is taken as the mean function often with a low-order polynomial format. The universal Kriging model adopts the polynomial functions, mostly linear or quadratic, as the regression part to capture the simple spatial trend [56, 57]. A polynomial function with an order higher than two should be avoided because complicated data trends are better described by the stochastic model component (Gaussian process) [57]. Based on the observed trend for the sweep-test results shown in Figure 3.10 in Chapter 3 (see Section of Test Results under T1 condition), a smooth response surface was obtained with a second order polynomial regression component for the Kriging model. With all these details, a surrogate Kriging model can be developed. Initial parameters for both Kriging models developed in Chapter 3 are shown in Table 2.1.

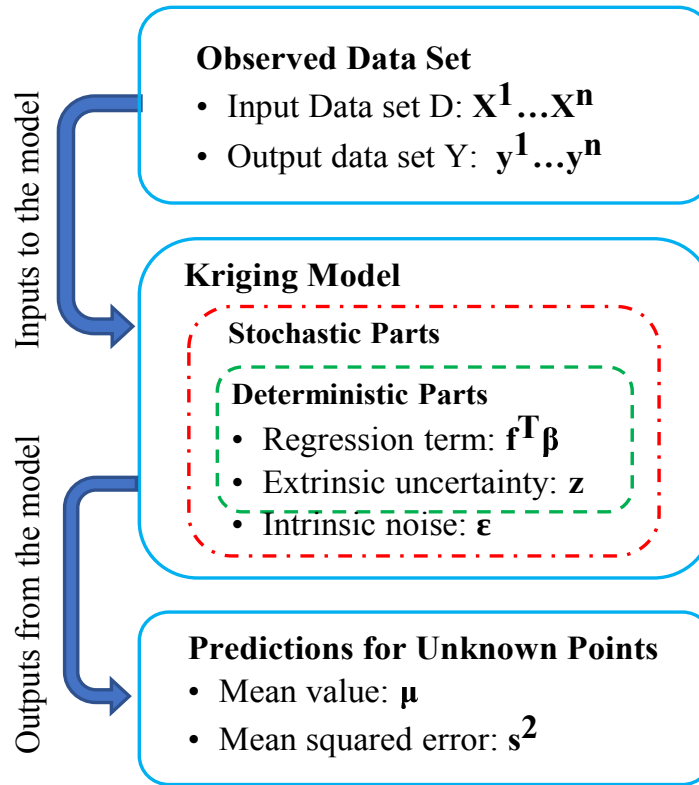


Figure 2.2 Structure of deterministic and stochastic Kriging models

2.3 Bayesian Training Algorithm with Modifications

For the first application to knock borderline prediction using a data-driven based methodology, a Bayesian optimization algorithm with the help of a data-driven based surrogate model is adopted

Table 2.1 Parameter setting for Kriging Model

Parameter	Value
Correlation function R_d	Gaussian (θ)
Regression function $\mathbf{f}(\mathbf{x})$	2nd order polynomial
Initial θ_l of likelihood function	10 ($l = 1, 2, \dots, p$)
Range of θ_l ($l = 1, 2, \dots, p$)	[0.01 100]

due to its efficient ability to obtain the optimum points of an expensive-to-evaluate high fidelity model, such as physical engine calibration. The essence of Bayesian statistics is to update a previous belief about the distribution over the objective function after obtaining new data. Accordingly, a posterior distribution is formed. The detailed algorithm with necessary modifications is going to be introduced along with the formulated problem for knock borderline prediction.

2.3.1 Problem Formulation and Bayesian Training Algorithm

The current research focuses on the engine's feasible operational condition when it is constrained by the borderline knock limit. The goal for engine knock control calibration is to maximize the engine thermal efficiency or minimize indicated specific fuel consumption (ISFC) and at the same time, maintain the engine knock intensity (KI) below a desired threshold. When the engine combustion is knock limited, there is a tradeoff between KI and ISFC. That is, reducing ISFC leads to an increment of KI. Therefore, a two-objectives min-min optimization problem of KI-ISFC can be formulated and described following.

$$\begin{aligned}
 &\text{Minimize} \quad y_1(\mathbf{x}), y_2(\mathbf{x}) \in \mathbb{R} \\
 &\text{subject to} \quad g_1(\mathbf{x}), g_2(\mathbf{x}), g_3(\mathbf{x}) \in \mathbb{R} \\
 &\quad \quad \quad \mathbf{x} = [x_1, x_2]^T \in \mathbb{R}^2
 \end{aligned} \tag{2.7}$$

where y_i ($i = 1, 2$) are two objectives, namely KI and ISFC; g_i ($i = 1, 2, 3$) are three constraints that are predefined lower and upper bounds (g_1 and g_2) of two design variables (x_1 and x_2), and combustion stability constraint (g_3) in terms of Coefficient of Variation (CoV_{imep}) based on indicated mean effective pressure (IMEP) to be less than 2.5%; and x_1 and x_2 are two control parameters in the design space, spark timing and VVT which is intake valve opening (IVO) timing, forming a design variable vector \mathbf{x} . Minimizing two competing objectives will generate a Pareto front in the objective space. Note that constraints imposed on control variables are used to avoid

engine running in the infeasible region, thus a safe operational region is defined in the design space. To solve this optimization problem, a full-factorial design of experiments (DOE) method is normally used for finding the optimal solution systematically in the past. Using this DOE-based method, each control parameter is divided into multiple bins distributed uniformly over the design space. This method is simple, but it results in a large number of evaluating points when variable resolutions are high. This will increase the evaluation cost, which is a challenge for physical engine tests due to limited resource and high test cost. In order to solve the optimization problem efficiently, surrogate-model-assisted optimization (SMAO) algorithm is adopted. Note that most of the existing work on SMAO is simulation based without considering measurement noises. The main goal of this study is to find tradeoff between KI and ISFC within limited engine bench tests using the SMAO algorithm, considering measurement and system noises.

The overall SMAO learning process is depicted in Figure 2.3, utilizing both mean $\mu_y(\mathbf{x})$ and variance $s_y^2(\mathbf{x})$ estimated by the Kriging model. The active learning process based on Bayesian optimization can be implemented iteratively in the following 5 steps with evaluations of high fidelity models or plants:

- S1. Evaluate the high fidelity model (or physical system) with a predetermined parameter set in the design space, for example, generating test data set using points generated by the Latin hypercube method.
- S2. Construct an initial Kriging model of the objective function using optimized hyper parameters (σ , β , and θ) by maximizing the likelihood function (2.4).
- S3. Find new location(s) in the design space that maximizes the expected improvement calculated by the acquisition function (see the next subsection).
- S4. Add evaluation point(s) recommended from S3. Train the Kriging model using the new expanded experimental data point(s).
- S5. Repeat S3 and S4 until the iteration budget is consumed or the error criterion has been met.

As shown in Figure 2.3, both stochastic and deterministic Kriging models need to be developed in a sequential iterative loop in this paper. For the engine knock prediction problem, a stochastic

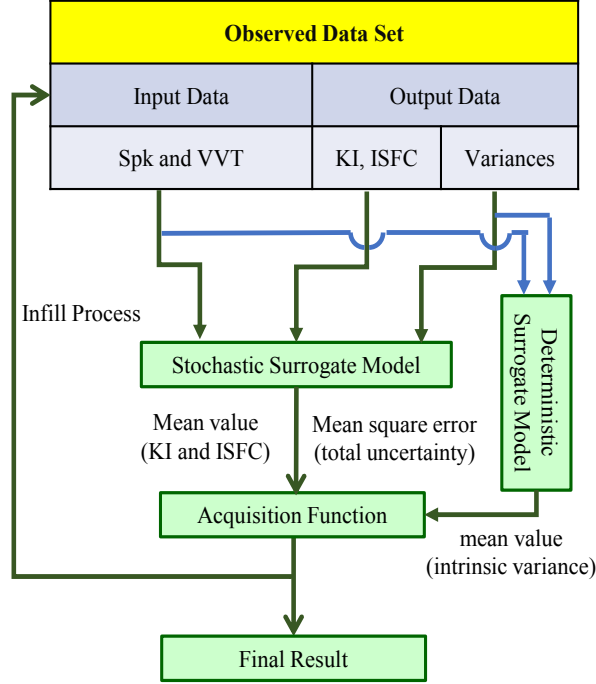


Figure 2.3 Iterative sequential optimization process

surrogate model is necessary due to its intrinsic cycle-to-cycle variations. To train this stochastic surrogate model, the data set includes an input vector in the design space (spark timing and VVT) and output data in terms of calculated mean and variance of two objectives (KI and ISFC). Kriging model training process follows the main center path shown in Figure 2.3. Then an iterative optimization process, called Bayesian optimization, is used to make it converge to the mean Pareto front assisted by an additional deterministic surrogate model (see the right side of Figure 2.3). Using two parallel surrogate models is mainly due to the calculation of acquisition function (to be explained later) for generating a new input vector in the design space to be evaluated by high fidelity model (or experimental data) so that the training process can be continued as described in Steps S3 and S4 of Bayesian optimization process. This new input vector is supposed to be in the design space with corresponding objective outputs closer to the true mean Pareto front.

Note that in the selected acquisition function, all the predicted mean values, extrinsic uncertainty, and intrinsic variance are required for calculating the maximum expected improvement of an unsampled design point. This calculation would be more straightforward for the deterministic case because intrinsic variance is zero and other two required variables can be predicted by the Kriging

model; but for the stochastic Kriging model, the predicted mean squared error consists of both system uncertainty and intrinsic variance that cannot be separated. This problem did not occur in associated stochastic literature in which the intrinsic noise variance is assumed to be a function of mean value, which is not true for practical applications. In reference [51], an intuitive theoretical idea is proposed to address this issue, where the intrinsic variance is represented by a spatial correlation model and fitted by a standard deterministic Kriging model. However, this idea was not actually verified in the literature. In this report, this idea is implemented for both test problems and actual engine bench test. To be more specific for our application of knock prediction, first, the deterministic surrogate model is trained using the data set including the input data (spark timing and VVT) and output data (KI and ISFC variances) as shown in Figure 2.3; and second, for knock prediction, the adopted cycle-to-cycle variance of the unknown site is important so that the engine can run strictly under the knock limit without unnecessary loss of efficiency. The deterministic model is used to predict the intrinsic variance after completing the training process.

2.3.2 Acquisition Function

As shown in Equations (2.5) and (2.6), both mean and uncertainty at any unknown point can be estimated. An intelligent search for next candidate(s) in the design space is performed using those estimations and current best solutions. Several acquisition functions are expressed in a closed form in the literature, and thus, are cheap to calculate. The main difference behind each method is the logic to balance the exploration in an unknown region and exploitation in the local region with the current best value. Focusing on exploration would lead to slow convergence but provide more room to achieve the global optima; on the other hand, focusing on exploitation will accelerate the convergent process but very likely be stuck in local optima. Out of all the methods, the standard expected improvement (EI) [58] is the most common one that reaches the balance and is originally developed for a single deterministic objective problem. The corresponding acquisition function for the stochastic Kriging model is an extension of the standard EI function. Note that the acquisition function is defined to guide the search towards the global optimal region, e.g., the Pareto front of two competing objectives, after balancing exploration and exploitation. This characteristic indicates

that the approximated surrogate model cannot fully replace the high fidelity model, but it is accurate enough to predict responses along the Pareto front that is essential for making the final decision of knock borderline.

For single deterministic objective problems, the definition of an EI function for the i -th objective function is shown in Equation (2.8).

$$EI_i(\mathbf{x}) = E \left\{ \max \left[y_i(\mathbf{x}^*) - Y_{p,i}(\mathbf{x}), 0 \right] \right\} \quad (2.8)$$

where \mathbf{x}^* is the current best solution, $y_i(\mathbf{x}^*)$ is the corresponding best value of the i -th objective function, $Y_{p,i}(\mathbf{x}) \sim N \left[\mu_i(y_i(\mathbf{x})), s_i^2(y_i(\mathbf{x})) \right]$. Note that $Y_{p,i}(\mathbf{x})$ can be interpreted as the Bayesian estimation for the posterior distribution of an unknown function. However in the context of multi-objective with stochastic functions, there are two challenges of calculating expected improvement: 1) expensive calculation due to multi-variable integration in improvement function and 2) the current best solution is not well defined for a stochastic case with exact values unknown. Based on the promising work developed in references [41, 59, 45], the standard EI functions EI_i ($i = 1, 2, \dots, m$) are extended to a single augmented matrix-based EI (AMEI) function to counter these two issues for multi-objective stochastic case. The closed expression of AMEI function for the n^{th} unknown design vector \mathbf{x}_n is formulated by Equation (2.9) below.

$$AMEI(\mathbf{x}_n) = \min_{j=1}^k \left[\prod_{i=1}^m (r_i + MEI_i^j(\mathbf{x}_n) - \mu_i^j) - \prod_{i=1}^m (r_i - \mu_i^j) \right] \quad (2.9)$$

where r_i is the self-chosen reference point for objective $i = 1, 2, \dots, m$ ($m = 2$ for our study) and should be dominated by all points in the current mean Pareto front, k is the total number of "effective best solutions" (to be defined later in Equation (2.12)), μ_i^j is the current i^{th} objective value estimated by a surrogate model with the j^{th} "effective best solutions", $MEI_i^j(\mathbf{x}_n)$ is the element at the i^{th} column and j^{th} row of the matrix-based function as shown in Equation (2.10) below and is obtained by adding a multiplication factor to the matrix based expected improvement function $EI_{i_{new}}^j(\mathbf{x}_n)$. This multiplication factor is suggested in reference [41] for discouraging the repetition of points after several function evaluations.

$$MEI_i^j(\mathbf{x}_n) = EI_{i_{new}}^j(\mathbf{x}_n) \left(1 - \frac{\lambda_i(\mathbf{x}_n)}{\sqrt{s_i^2(\mathbf{x}_n) + \lambda_i^2(\mathbf{x}_n)}} \right) \quad (2.10)$$

where $\lambda_i(\mathbf{x}_n)$ and $s_i^2(\mathbf{x}_n)$ is the estimation of noise and mean squared error of the i^{th} objective at any unknown point, respectively. $EI_{i_{new}}^j(\mathbf{x}_n)$ is a modified matrix-based expected improvement function of unknown point \mathbf{x}_n relative to the j^{th} effective best solution. Its value delegates the expected improvement of an unknown design point \mathbf{x}_n for the i^{th} objective value with respect to the j^{th} "effective best solution". In terms of two-objective problem in this paper, the j^{th} row represents the EIs in all objectives (KI and ISFC) at the j^{th} "effective best solution", and the i^{th} column represents the EIs in the i^{th} objective at all the "effective best solution". The expression of the new EI for stochastic Kriging model is:

$$EI_{i_{new}}^j(\mathbf{x}_n) = h(\mathbf{x}_j^{**}, \mathbf{x}_n) \Phi\left(h(\mathbf{x}_j^{**}, \mathbf{x}_n)/s_i(\mathbf{x}_n)\right) + s_i(\mathbf{x}_n) \phi\left(h(\mathbf{x}_j^{**}, \mathbf{x}_n)/s_i(\mathbf{x}_n)\right) \quad (2.11)$$

$$h(\mathbf{x}_j^{**}, \mathbf{x}_n) = \mu_i(\mathbf{x}_j^{**}) - \mu_i(\mathbf{x}_n)$$

where \mathbf{x}_j^{**} is defined as the j^{th} "effective best solution" set \mathbf{x}^{**} over these observed points using Equation (2.12), Φ and ϕ represent cumulative distribution and probability density functions, respectively.

$$\mathbf{x}^{**} = \operatorname{argmax}[-\mu_{i...m}(\mathbf{x}) - cs_{i...m}(\mathbf{x})] \quad (2.12)$$

where $c = 1$ is used in this report.

As shown from Equations (2.9) to (2.12) for calculating the acquisition function, the predicted mean value μ_i , total uncertainty s_i and intrinsic noise λ_i at unknown points are required to complete the calculation. The proposed structure of two parallel Kriging models is able to provide all required unknown information. After creating the acquisition function, optimization is performed to decide candidates for the next iteration. The AMEI approach aggregates the matrix based MEI function into a scalar value to measure the overall improvement of the studying unknown point. To solve a single-objective optimization problem, evolutionary optimization algorithm (for example, genetic algorithm) is widely used. In this paper, the "real-coded Genetic Algorithm (rGA)" developed by Deb [60] is used to maximize the AMEI acquisition function due to its high efficiency. This algorithm is easy to use because the crossover and mutation operators are applied directly to real parameters instead of the typical binary-coded value [60]. Parameters for the optimization process are listed in Table 2.2.

Table 2.2 Parameters used in "rGA" algorithm

Parameter	Value
Population size	$20 \times n$ ($n = 2$ - design variable #)
Total generation #	100
Pcrossover	0.9
Pmutation	$1/n$

2.3.3 Distribution Mapping

Kriging model is a Gaussian process model and the extrinsic variance of unknown location introduced before is also assumed to be Gaussian, as shown in Figure 2.4. This figure shows the prediction with a vertical Gaussian distribution which represents the uncertainty of this prediction. The center and spread of this Gaussian distribution are the mean and variance predicted by the Kriging model. This uncertainty is useful for the calculation of expected improvement and the final decision after training process with 3-sigma rules considered. However, this assumption of Gaussian distribution does not always valid for practical applications. A non-Gaussian distribution needs to be transformed to a Gaussian one.

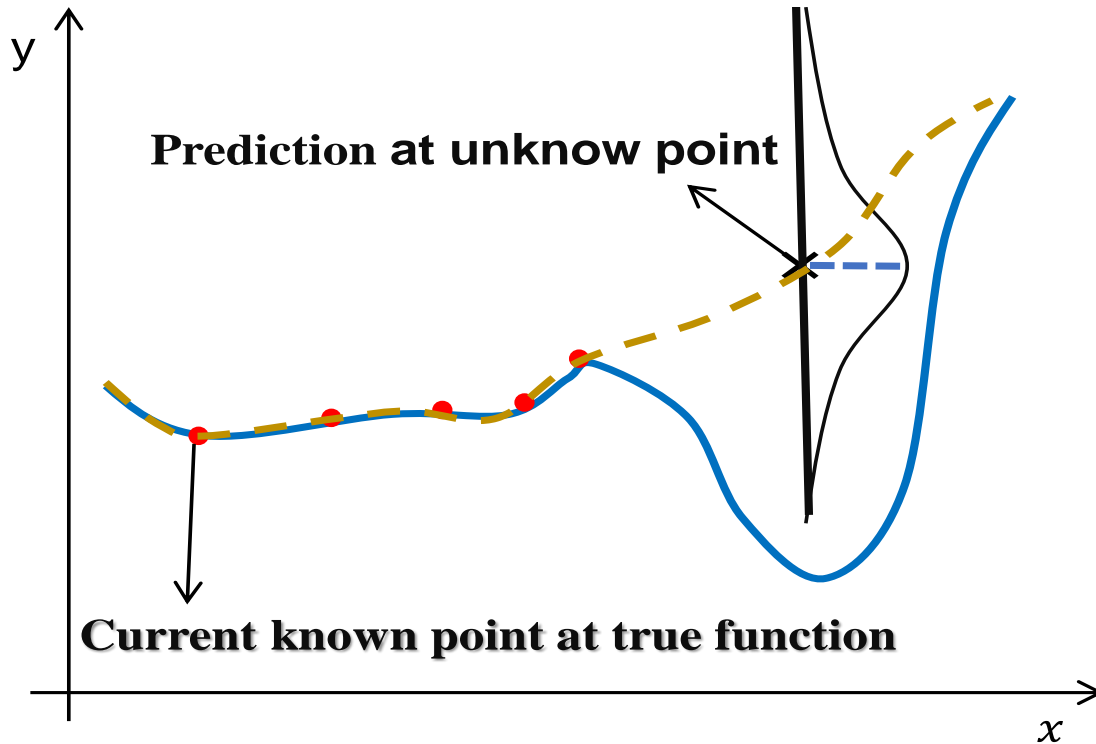


Figure 2.4 Graphical demo about the Gaussian distribution of the intrinsic variance

For the knock application in this dissertation, KI distribution with its corresponding statistic features is vital for Kriging borderline knock model since the final selection of control parameters depends on the predicted means and variances from the Kriging model. Our early study on stochastic knock control [8] revealed that the knock intensity probability density function (PDF) is not symmetric and obviously not a Gaussian random process, but has a log-nominal KI distribution that was also verified by other researchers [61]. Thus a distribution mapping is necessary to make it possible to implement the Bayesian learning process. Referring to our test data obtained by running the engine close to the knock borderline, the knock intensity distribution matches the log-nominal distribution better than the Gaussian one, as shown in Figure 2.5. It can be noticed that in the region with high KI, Gaussian distribution has a very large error. If a Gaussian distribution were used during the optimization of KI surrogate model, the high KI event would not be included, and therefore, knock prediction based on Gaussian distribution Kriging model would likely cause heavy engine damage.

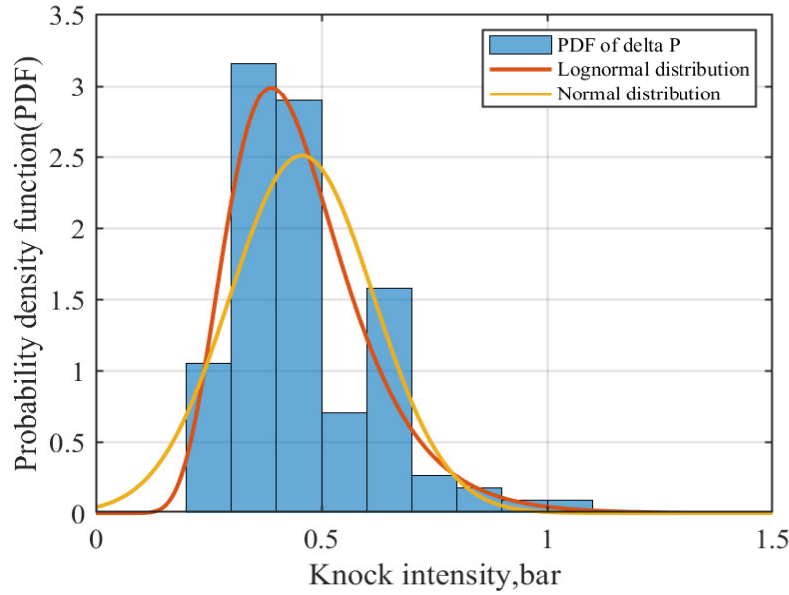


Figure 2.5 Knock intensity (bar) PDF with proposed distributions

Before starting the iterative learning process, the log-nominal distribution KI data is mapped to the Gaussian one to meet the assumption for Bayesian learning algorithm; and after completing the learning, the predicted values (means and variances) are mapped back to the original log-normal distribution using the exponential mapping method. The mapping function can be referred to

Figure 2.6. If $x \sim N(\mu_x, s_x^2)$, $y = e^x \sim \ln(\mu_y, s_y^2)$ and its mean and variance are given in Equation (2.13). Then the interval containing 99.7% of probability can be obtained according to the 3-sigma rules in the statistic. Conversely, μ_x and s_x^2 are found from μ_y and s_y^2 .

$$\begin{cases} \mu_y = e^{\mu_x + \frac{1}{2}s_x^2} \\ s_y^2 = e^{2\mu_x + s_x^2} (e^{s_x^2} - 1) \end{cases} \quad (2.13)$$

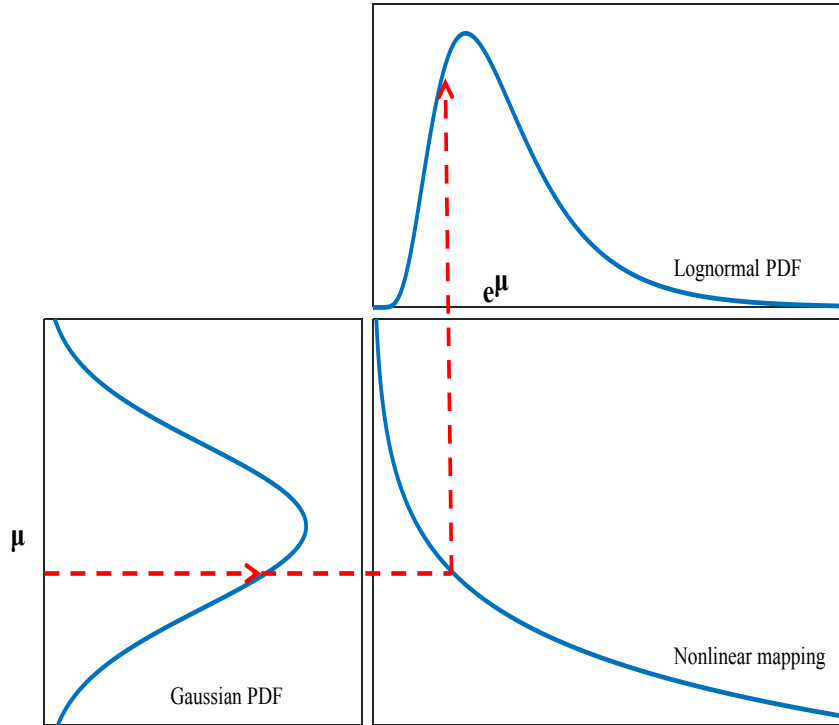


Figure 2.6 Knock intensity (bar) PDF transformation

2.4 Summary

In this chapter, a few key statistic concepts were introduced, which are important roles in our proposed data-driven based research. After that, the data-driven model and algorithm is elaborated with the analysis of one specific application. The development of the Kriging model and its role in the SMAO algorithm is described in detail. The modified workflow of SMAO is explained, which includes two essential adjustments for fitting the stochastic variation of knock combustion. With all these modifications, the proposed data-driven based estimation algorithm is ready for implementation in the next a few chapters.

CHAPTER 3

IMPLEMENTATION OF ALGORITHM THROUGH SIMULATION AND REAL TIME ENGINE BENCH TEST

The previous chapter introduces the Kriging model along with its associated algorithm. The development and characteristics of deterministic and stochastic Kriging models have been fully discussed along with the essential modifications, including the dual surrogate model structure for the stochastic environment and distribution mapping for a non-Gaussian noise. The implementation of the learning algorithm will be made available for both simulation-based model (function) and actual physical engine.

Before being applied to a physical engine bench, it is important to verify the algorithm under some simulation scenarios. This chapter starts with simulation study for two selected problems and reaction-based combustion model. Meanwhile, the engine bench is developed at MSU Energy and Automotive Research Laboratory, where the associated algorithm was implemented under two different operating conditions. Based on the analysis of test data, the effect of proposed algorithm is discussed and the conclusion is drawn at the end of this chapter. The obtained surrogate model is going to be used for online updating process in the next chapter.

3.1 Verification of Algorithm on Test Problem and Combustion model

Two classical numerical problems (with and without constraints), along with a control-oriented reaction-based combustion model, are selected for the simulation-based verification, respectively. Note that Gaussian noise with zero mean is added to the true functions/models for both scenarios. Therefore, for this simulation study, there is no distribution mapping required.

For both test problems, firstly a noise level is assumed to be known in the calculation process of the acquisition function. The results will be compared with the proposed dual surrogate model structure when the noise level is assumed to be unknown. The reaction based combustion model [20] is previously developed in our research group which is a control-oriented two zone combustion model with a real-time zero dimensional knock pressure wave model for spark ignited engines. The knock pressure wave model is capable of predicting the in-cylinder pressure oscillations under

knocking combustion and can be used for model-based knock prediction and control. For our verification purpose, the proposed learning algorithm is applied to the combustion model for both deterministic and stochastic cases.

3.1.1 Simulation on Numerical Test Problems

The proposed method is first validated on standard multiobjective numerical test problems available in literature [46]. Two standard problems, ZDT1 and BNH, are selected for unconstrained and constrained cases, respectively. For test problems, a predefined noise level is added to the true problem. The noise level is set to 6% of the mean value at each point. The comparison simulation was run for two different settings. Firstly, the noise is assumed to be known and is fed into the acquisition function while in the second scenario, the noise is unknown and the dual surrogate model structure is used for the calculation of acquisition function. The optimization results for two test problems are shown in Figures 3.1 and 3.2.

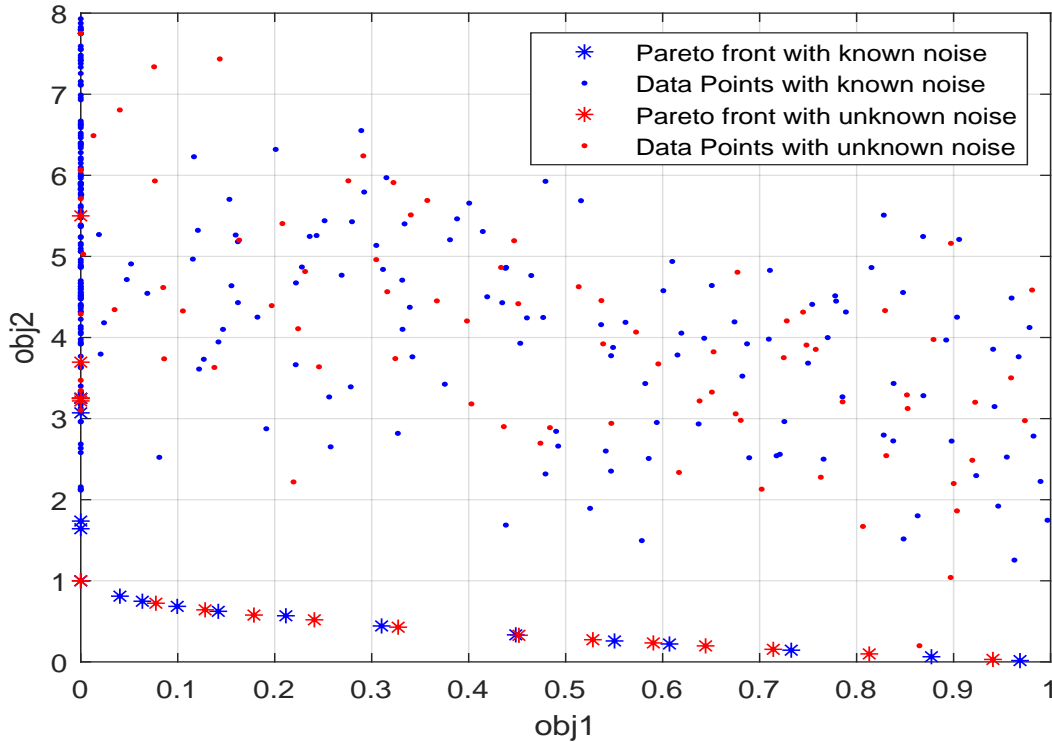


Figure 3.1 Simulation for ZDT1 problem with predefined noise level

As shown in both figures, the proposed algorithm was able to explore the unknown regions of the objective space and finally find a few points along the Pareto front within 100 high fidelity

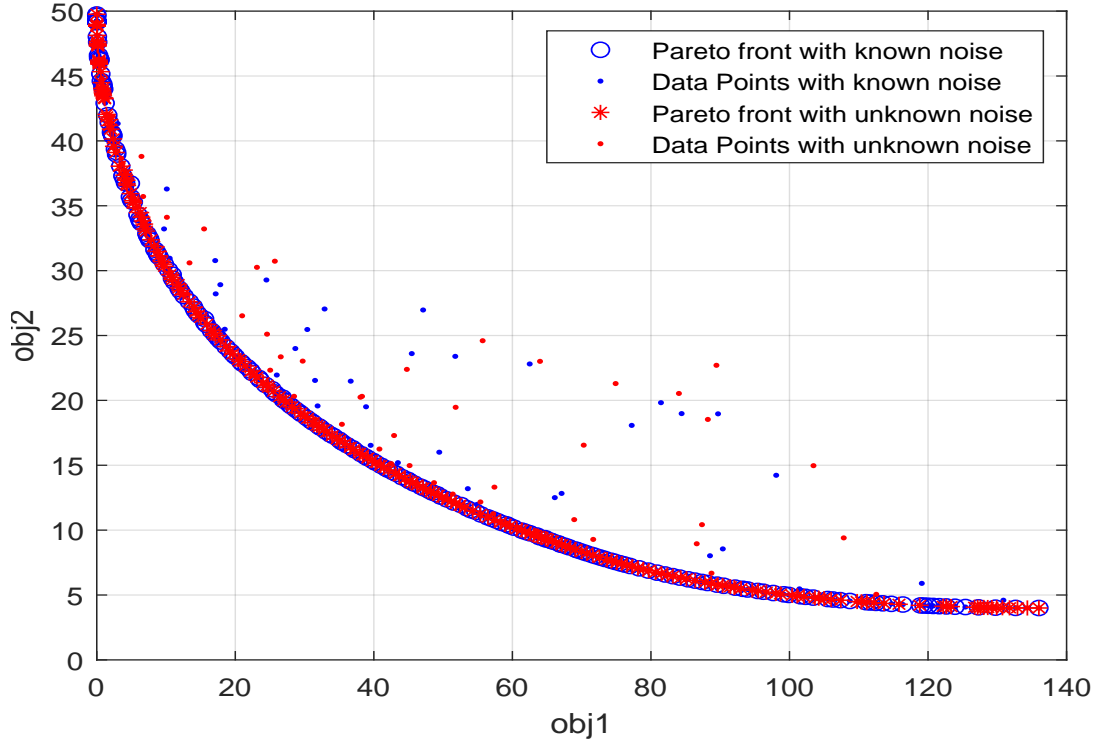


Figure 3.2 Simulation results for BNH problem with predefined noise level

evaluations. Primarily, we can notice that it can still track the Pareto front well by using the dual surrogate model structure, comparing with the front obtained by using the known noise level. Note that a conventional optimization search algorithm, such as NSGA-II [46], would take a few hundred iterations to find the true Pareto front [62].

3.1.2 Simulation on Reaction-Based Combustion Model

The reaction-based combustion model was developed for real-time knock control research to predict the SI engine knock, which was fully calibrated and validated by experimental data. It can provide knock intensity and in-cylinder pressure under different control inputs. Spark timing and EGR rate are defined as two control parameters for this model. Thus, this model is useful for verifying two things: 1) The existence of trade-off relationship between KI and fuel economy with varying two control parameters. For this model, the brake-specific fuel consumption (BSFC) is selected as the index of fuel economy; and 2) The ability to detect the Pareto front within limited simulation budgets after the noise is added.

The combustion model was built in the Simulink platform, and it is a deterministic model.

Using the deterministic surrogate model directly, the simulation result is shown in Figure 3.3. The result shows that within 100 iterations, the detected Pareto front is approaching the true one. The efficiency of algorithm is significant after we compared it with the baseline Pareto front which is obtained by using 450 sweeping simulations. Since the stochastic characteristic of knock is ignored in this run, the results cannot help us to find the upper bound of KI.

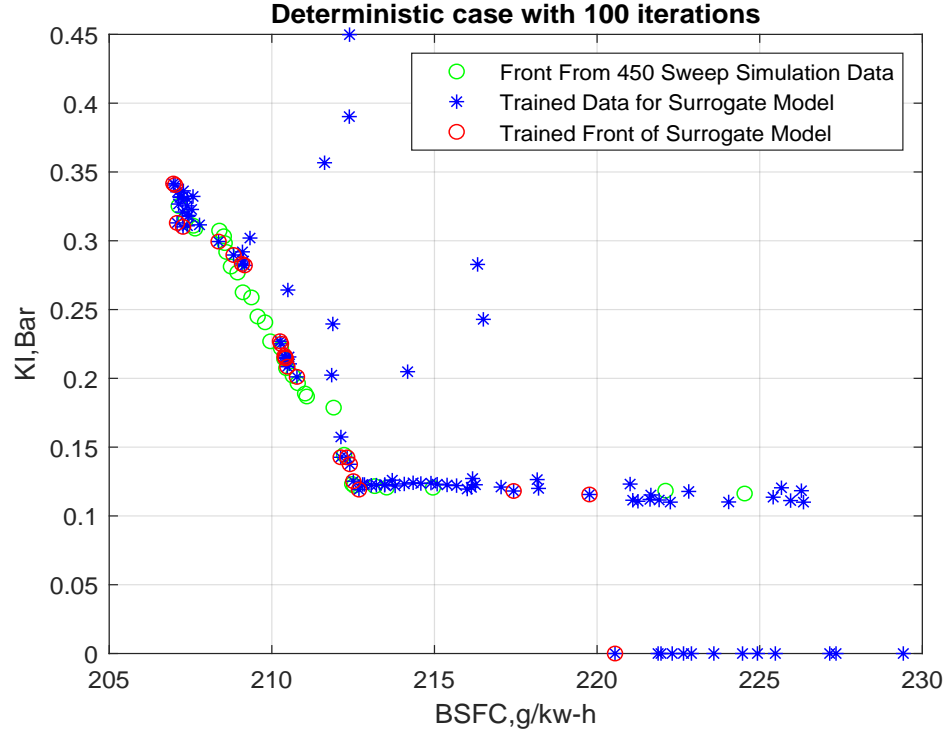


Figure 3.3 Simulation results of reaction-based model without noise

Next, the external noises are added to the outputs of two objectives to make the process stochastic (realistic). The simulation results are shown in Figures 3.4 and 3.5, indicating that more iterations (200) are required for detecting the true Pareto front. Meanwhile, the statistic variance bar in Figure 3.5 can provide useful information of KI upper bound for further knock borderline control.

3.2 Experimental Configuration

The above discussions are associated with the simulation-based studies. To validate the modified Bayesian learning algorithm and the future online updating process for borderline knock prediction and control, an engine test bench is needed with the necessary software and hardware configurations. An 1.5L four-cylinder turbocharged gasoline engine was installed on an MSU dynamometer for

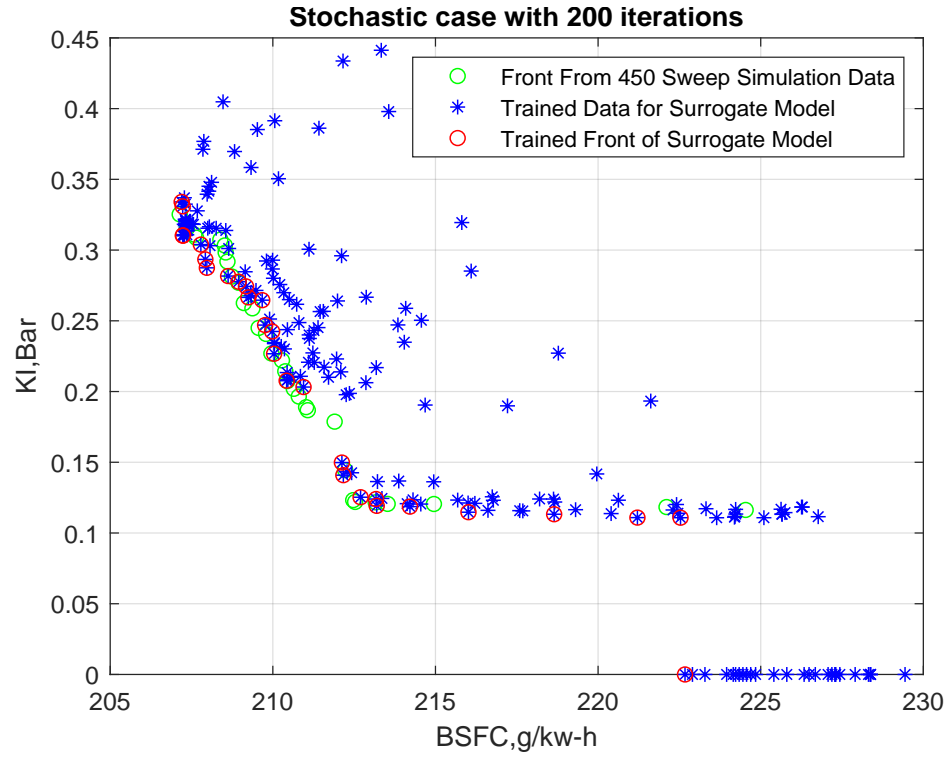


Figure 3.4 Simulation results of reaction-based model with noise

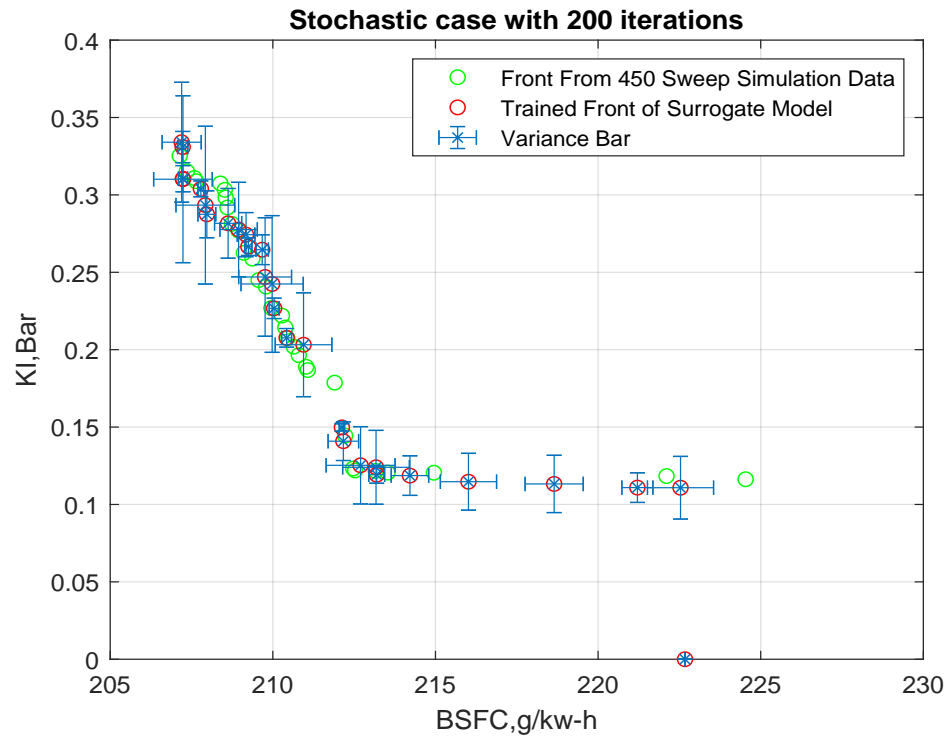


Figure 3.5 Simulation results of reaction-based model with variance bar

validating the learning algorithm of borderline knock limit. Note that a sweeping test was made firstly for comparison purpose before implementing the learning algorithm under two different operational conditions. The specification of the test engine is shown below in Table 3.1, and the main hardware and sensors used are listed in Table 3.2.

Table 3.1 Engine spec list

Item	Value
Engine Type	1.5L, 4cycle, Gasoline
Configuration	L4
Displacement	>0.95
Bore	75mm
Stroke	84.8mm
Compression Ratio	9.5
Fire Order	1-3-4-2
Fuel Injection	Port

Table 3.2 Configuration table for dyno test

Item Number	Name	Type or Model Number	Producer	Usage
1	Dyno System	Sakor	Sakor	Speed control and torque measurement
2	Controller	ECM 5554/0904 Prod	MotoTron	Controlling the following key variables: dwell, short, synchronization, coolant valve, pedal position, safety control, CAN definition
3	PCM	32 bits controller	FIT	Controlling all other engine variables
4	Coolant System	NA	MSU	Controlling the coolant temperature
5	CAN Box	NA	FIT	Communication between host PC and PCM
6	Data Acquisition	USB 6361	NI	Data sampling and recording at 100kHz
7	Ion Circuit Board	NA	MSU	Generating dwell, short and ion signals
8	Pressure Sensor and Amplifier	6118CF-5CQ02-4-1	Kistler	Measuring in-cylinder pressure
9	Current Sensor	NA	MSU	Monitoring dwell currents (all cylinders)
10	AFR Recorder	1200A	AFrecorder	Monitoring downstream air-to-fuel ratio
11	Hall Sensor and Encoder	NA	MSU	Measuring cam and crank positions

The entire engine test bench configuration is shown below in Figures 3.6 and 3.7. From the engine side, an alternating current (AC) dynamometer (dyno) coupled with an engine crankshaft, is used to supply and absorb engine power and maintain the engine speed. Note that the dyno is

operated under speed control mode. Necessary harnesses are made for the signal communication among the controllers, sensors, and actuators. From the control side, two controllers are used. One is an open-source engine control unit (ECU), MotoTron, which is used to send control signals to peripheral devices, such as cooling valve, fuel pump, ion detection circuit boards. MotoTron module is also responsible to send the commanded pedal position signal to the second controller, engine control module (ECM), for the desired engine load. The ECM also sends out the command for fuel injection quantity, injection timing, intake valve timing and other actuators based on current engine speed and pedal position. The calibration of production ECM can be accessed by its calibration software similar to ETAS INCA [62]. A few sensors are attached to the engine for different functions. In Table 3.2, key sensors with the corresponding type and model numbers are listed. For checking the work status of the engine during test, fuel pressure and level, oil pressure, ion signal, dwell current, air fuel ratio, coolant temperature and engine torque delivered are monitored to make sure the engine is ignited in the correct phase and run smoothly. On the other hand, the in-cylinder pressure, exhaust temperature and knock signal are recorded for further analysis and control purposes. Note that, only the 3rd cylinder pressure is measured by a Kistler spark-plug pressure transducer. A bare-wire thermal couple with a short time constant is installed on the exhaust manifold close to the exhaust port of the 3rd cylinder. The in-cylinder pressure signal passes through the Kistler charge amplifier and is sent to an NI USB6361 data acquisition system. Crank synchronization is programmed in MotoTron using the signal from the installed hall effect sensor and 60-2 teeth encoder. A test GUI is designed in LabView for data visualization and communication among MotoTron controller, engine PCM and host PC graphic user interface. The recorded data and control command is across multiple devices, therefore CAN communication is implemented among the MotoTron module, NI DAQ system, and ECM computer. In this way, data can be shared on the CAN bus; and thus NI LabView interface can be used for data communication, control and calibration.

Regarding the implementation of machine-learning algorithm into dyno control system, the entire data communication flow is shown in Figure 3.8. A Kistler piezoelectric in-cylinder pressure

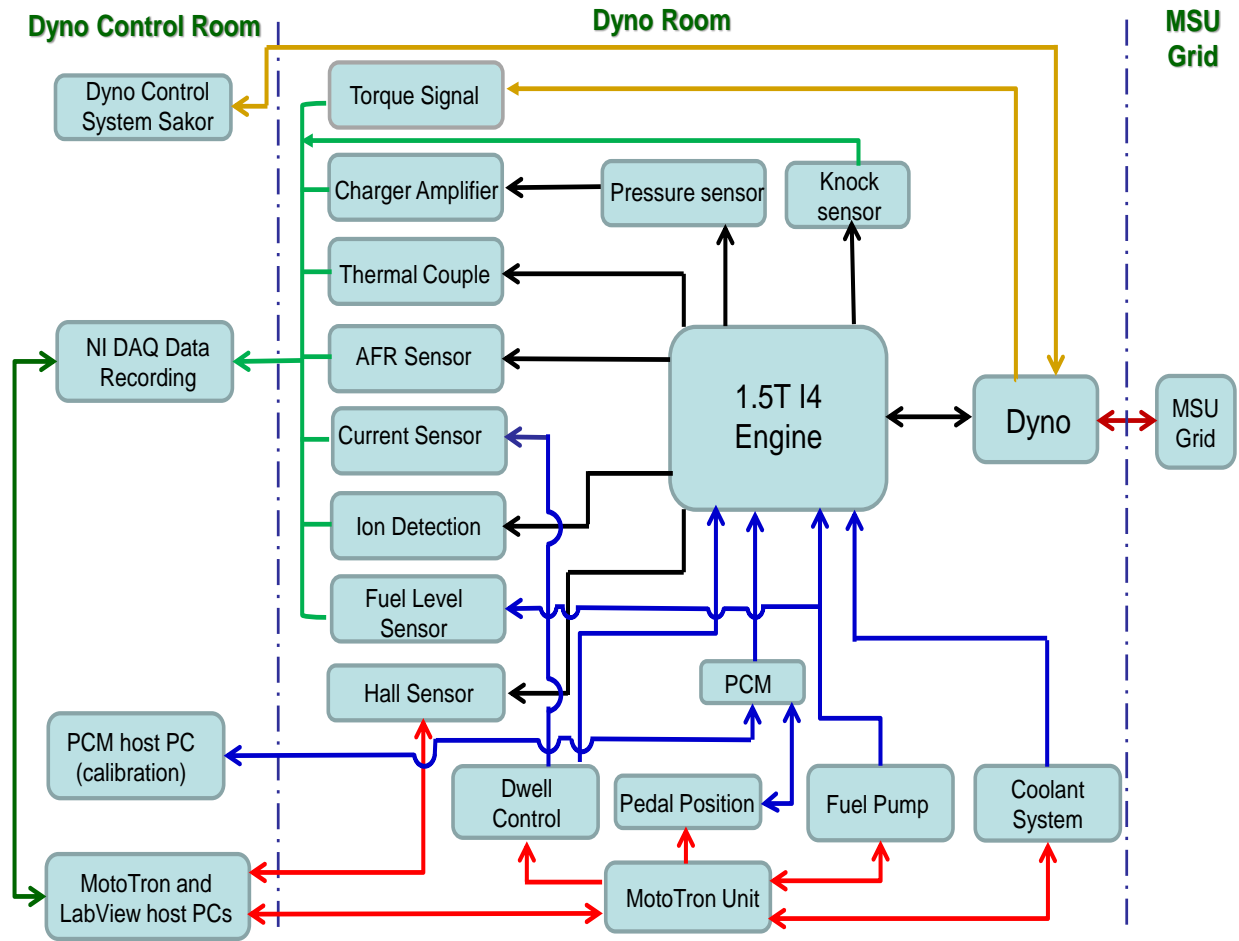


Figure 3.6 Engine test bench configuration

sensor was installed in the 3rd cylinder to monitor the combustion process and its signal is recorded by NI USB6361 at a sample rate of 100 kHz. The pressure signal is sent to the learning algorithm in Matlab for further analysis. A bare-wire thermal couple with a 0.003s response time is installed close to the exhaust valve of the 3rd cylinder. The exhaust temperature data will be investigated in Chapter 5. An open-source MotoTron engine controller is utilized to accept ignition dwell timing from the algorithm and send it to the engine; and all other actuators are controlled by the original engine control module (ECM), for example, closed-loop control of stoichiometric air-to-fuel ratio, variable valve timing (VVT), fuel injection, etc. The engine was running under two different operation conditions summarized in Table 3.3. For all tests discussed in this chapter, the engine speed was kept constant using regular E10 gasoline.



Figure 3.7 Engine on the MSU dyno

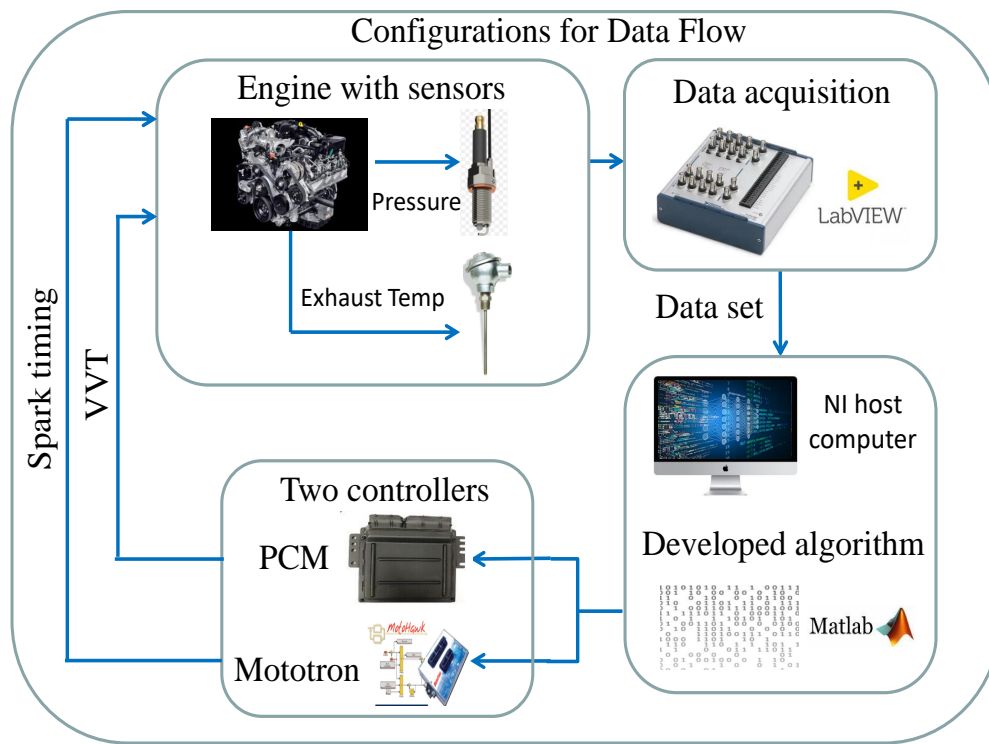


Figure 3.8 Signal flow of engine testing system

Table 3.3 Engine operational conditions for the learning algorithm

Test #	Speed (rpm)	Load (Nm)	VVT (°)	Spark Timing, deg BTDC
T1	1200	90	[20,30]	[0, 7]
T2	1500	80	[20,30]	[0, 8]

3.3 Sweeping Test

In order to verify the functionality of the proposed algorithm for knock borderline prediction, the engine needs to be operated at a region where the MBT (maximum brake torque) timing is limited by the knock borderline. Early research [63] indicated that when 50% mass fraction burned (MFB) crank location, denoted by CA50, is around 9 deg after TDC (DATDC), the MBT timing is achieved [63]. A fast real-time CA50 calculation method [64] was used for all engine tests. For example, when the engine is run at 1200 rpm (T1 in Table 3.3), a sweep spark timing test with a fixed VVT was conducted; see Figure 3.9 for detailed results. Note that at the most advanced spark timing of 6 deg before TDC (DBTDC), CA50 (blue circle) is about 12.5 DATDC, a few degrees away from the MBT CA50 location around 9 DATDC; however, the knock intensity (KI), based on maximum peak-to-peak bandpass-filtered pressure [65, 66] (KI-DP shown in red-star of Figure 3.9) is greater than 1 bar, indicating that under this operational condition, MBT timing is not reachable due to borderline knock limit. The bandpass filter frequencies used are 4 and 20 kHz. For the rest of this dissertation, the KI-DP denotes the knock intensity index calculated using the maximum peak-to-peak bandpass-filtered pressure. KI (or KI-DP) and ISFC are selected to be two competing objectives in our study. For multi-objective optimization based on meta-modeling, a taxonomy was summarized in reference [67]. Out of all the available methods, the "M1" method is utilized in this dissertation so that the mean and variance of each objective can be predicted by its own surrogate model, respectively.

Spark sweeping tests under three different fixed VVT positions were implemented, respectively, for comparison purposes under the T1 condition. For the sweep test, spark timing was swept from 0 to 7 deg BTDC and VVT was fixed at 20, 25 and 30 deg BTDC, respectively. Note that when the spark timing is between 0 and 4 degrees BTDC, the spark timing sample interval is 0.5 degree,

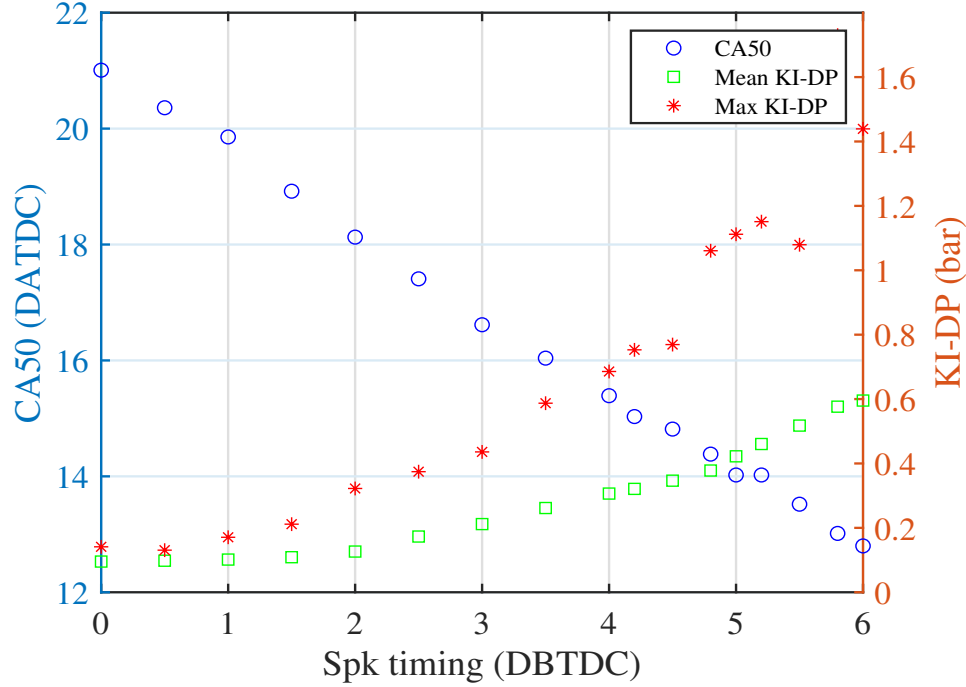


Figure 3.9 CA50 calculation

and when it is between 4 and 7 degrees BTDC, the sample interval is 0.25 degree. Since the engine controller has a spark timing control resolution of 0.1 degree, these four samples within one degree are selected as 0.0, 0.2, 0.5, and 0.8 degree. The total test budget is $21 \times 3 = 63$, where 21 represents total different spark timings within the predefined VVT range and 3 represents total different VVT timings. These input-output relationships are plotted in Figure 3.10 that provides information regarding the spatial trend of the response surface.

As shown in Figure 3.10, both ISFC and KI-DP are nonlinear functions of spark timing and VVT. Regression surfaces using the first- and second-order polynomials were compared as well. It shows that the Kriging model using the second order regression function captures the mean trend better than that using the first-order one. Especially, for the objective of KI-DP, the response of the 1st order regression shows a multi-modal curve which is not the case of our gasoline engine. A Pareto front, which is the tradeoff relationship between KI and ISFC, is obtained by performing non-dominated sorting on all evaluated points; see red stars in Figure 3.11a. The competing relationship of these two objectives is clear as expected.

However it is not clear if these obtained results are optimum under the influence of two pre-

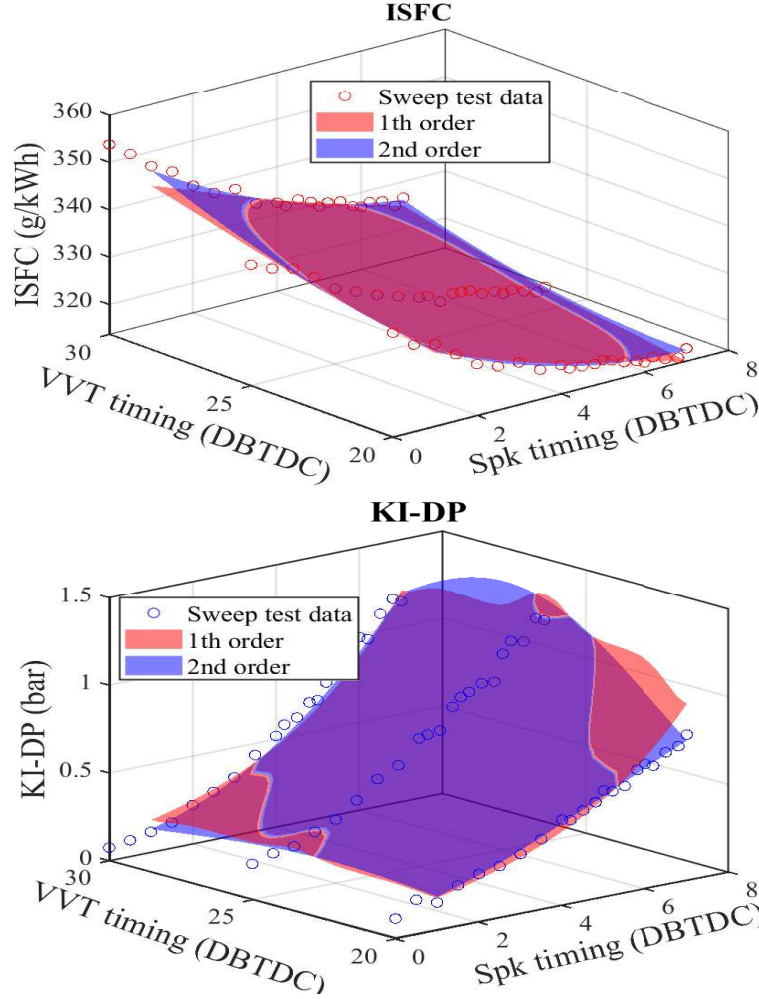
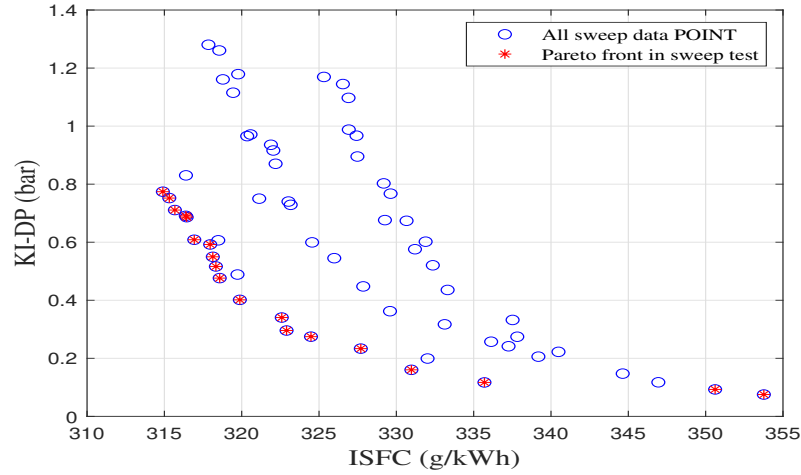


Figure 3.10 Response surface based on sweep-test data under T1 condition

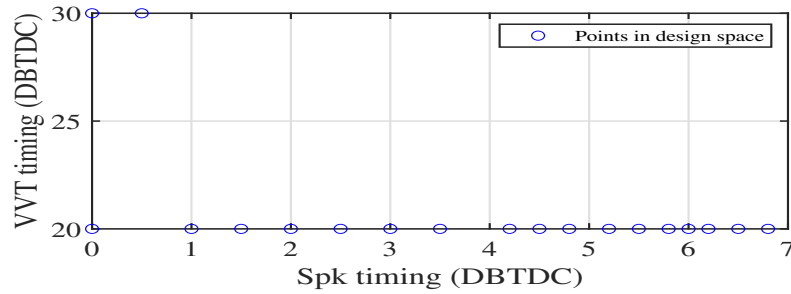
defined factors. As shown in Figure 3.11b, most of the VVT locations tested along the Pareto front are at 20 deg BTDC. Thus, under this operational condition, the optimal VVT could be within the region $[20, 25)$ or $(25, 30]$ through learning process. If a DOE test matrix (VVT and spark timing) was used to find the global optimal region(s), test budget would be very high obviously. For example, if one deg step size was used for VVT timing, the required total number of tests is $21 \times 11 = 231$. The efficient learning algorithm proposed in this paper is expected to significantly reduce the required number of tests.

3.4 Implementation of Learning Algorithm onto Engine Bench

Engine tests are expensive in terms of cost and human efforts. In order to make full use of the limited test resource, the Bayesian optimization algorithm with proposed modifications is utilized



(a) Results in objective space



(b) Design sites of points along Pareto front

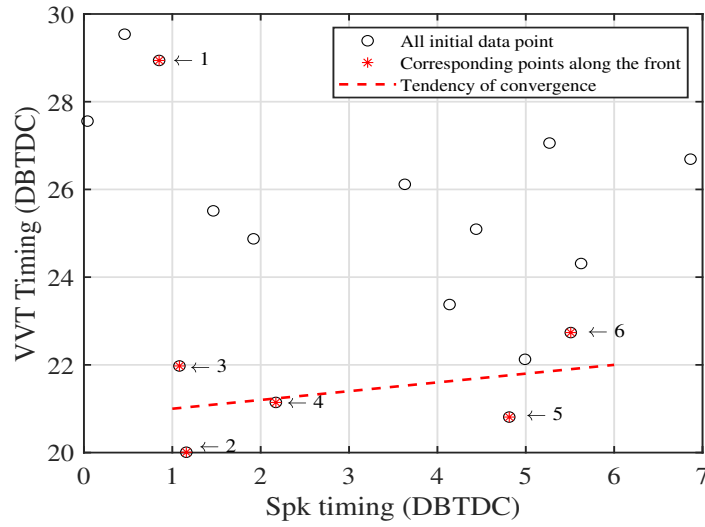
Figure 3.11 Sweep test results

under the same operational conditions as shown in Table 3.3. An improved Pareto front is expected with fewer test iterations consumed.

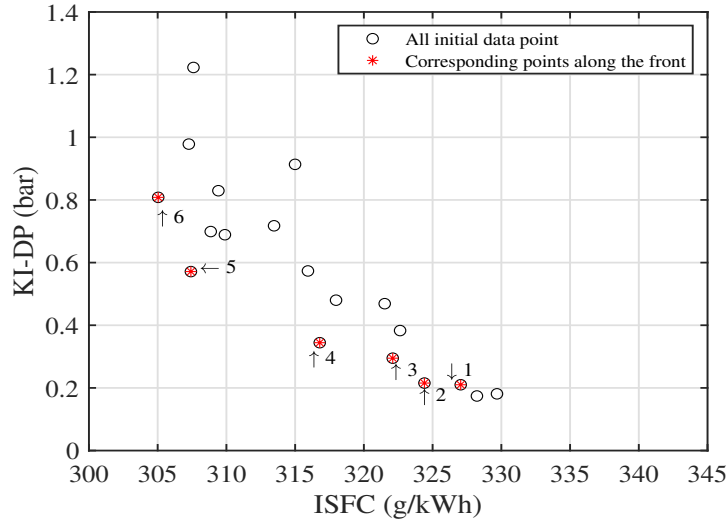
3.4.1 Test Results under T1 Condition

Using Bayesian optimization process sequentially, the learning algorithm makes a decision about updating the design variables. As shown in Figure 2.3, both spark timing and VVT candidates are determined based on the optimized acquisition function at each iteration. As described in Chapter 2, the algorithm starts by creating an initial input data set using any method available in literature. It is an important step of forming the initial response surface, which could very well direct the optimal search process based on the developed initial surrogate-model. Therefore, it is important to have a good initial model, spanning the points over the entire design space. Latin Hypercube Sampling (LHS) is a widely used method for creating uniformly distributed points throughout the design space. Under T1 condition, at the beginning of the test (step S1 in Figure 2.3), the initial

training input data set is generated by the LHS method and evaluated by running the engine to form a complete initial training data set including both inputs in design space and outputs in objective space shown in Figures 3.12a and 3.12b, respectively, where red points marked with sequential numbers are these points along the initial Pareto front. It is found that for the initial data set, there are 6 points along the final Pareto front (see Figures 3.13 and 3.14) and after completing the learning process, the Pareto front was formed by 11 test points. It can be observed that the VVT along the Pareto front is between 20 and 22 deg BTDC under operational condition T1.



(a) Design space of initial training data set



(b) Two objectives of initial training data set

Figure 3.12 Test results using initial data set under T1 condition

After Step S1, continuing running the engine at each new test point generated by carrying out the procedure from Steps S2 to S4 in the learning algorithm, more points along the Pareto front are expected to be found through exploration and exploitation processes. After consuming all predefined test budget of 40, the resulting input and output points in design and objective spaces, respectively, are shown in Figures 3.13 and 3.14, where blue squares represent the infilled points by optimizing acquisition function during iterations. Among them, these green stars are the new points along the Pareto front and their tendency is consistent with the initial data set but provides a better spread or diversity of the Pareto front. We can conclude that under this engine operational condition, the design space parameters along the Pareto front mainly depends on the spark timing where the VVT timing is less sensitive.

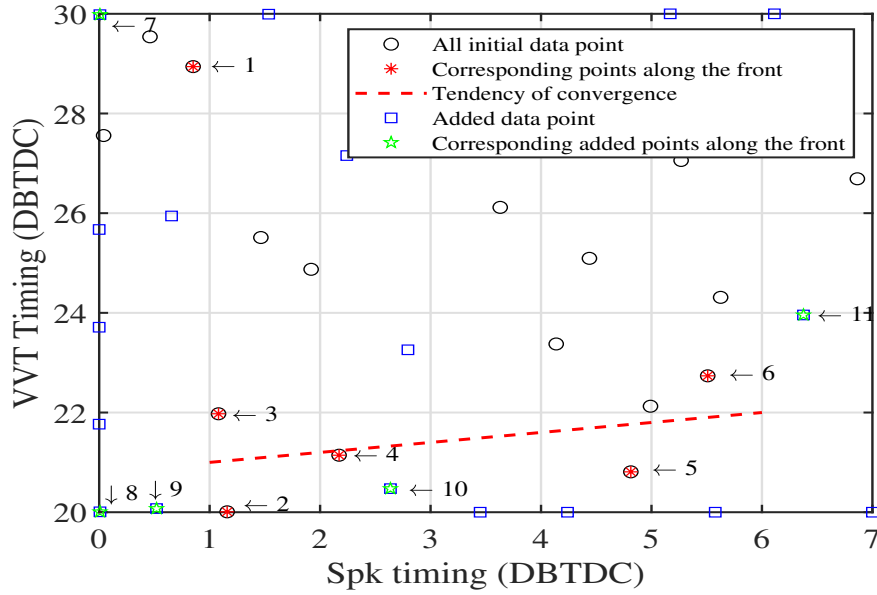


Figure 3.13 Design space of final training data set under T1 condition

After a total of 20 iterations (40 total test point budget), not only scatter points along the Pareto front can be obtained, but also the surrogate Kriging model can provide an approximated Pareto front. Two smooth response surfaces are shown in Figure 3.15 and Figure 3.16, where x_1 is spark timing in deg BTDC, x_2 is VVT in deg BTDC, y_1 is ISFC in g/kWh , and y_2 is KI-DP in bar .

Accordingly, the Pareto front of the surrogate-model and its corresponding design space are shown in Figure 3.17 and Figure 3.18, respectively.

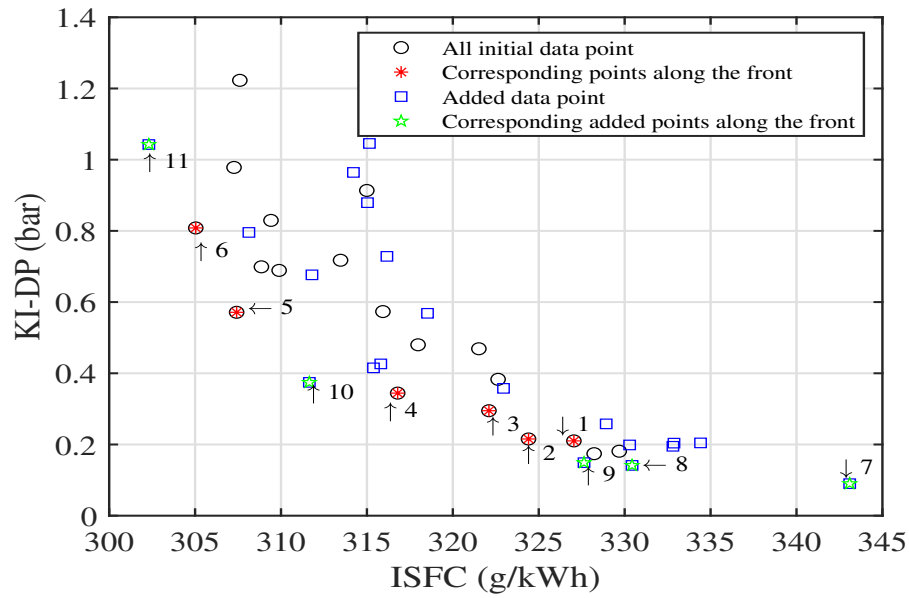


Figure 3.14 Two objectives of final training data set under T1 condition

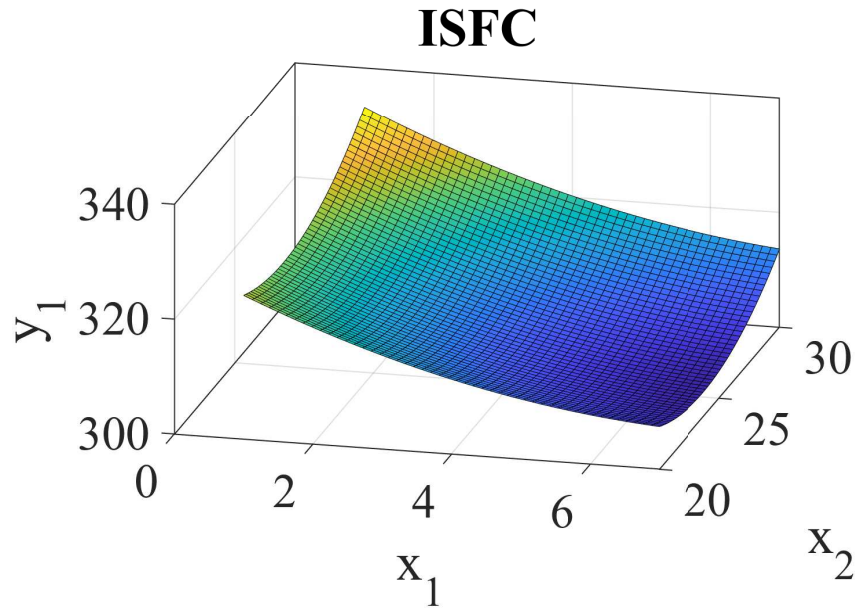


Figure 3.15 Response surface (ISFC) of trained surrogate model under T1 condition

In the design space (Figure 3.17), a similar optimal combination of spark timing and VVT can be found, where these red stars are the same points as those in Figure 3.13 and the blue circles are generated by the surrogate model. On the other hand, the Pareto front of the surrogate model is converging to the lower left corner during the iteration process, guided by the learning algorithm (see direction pointed by arrows in Figure 3.18) with three different regions circled. When spark

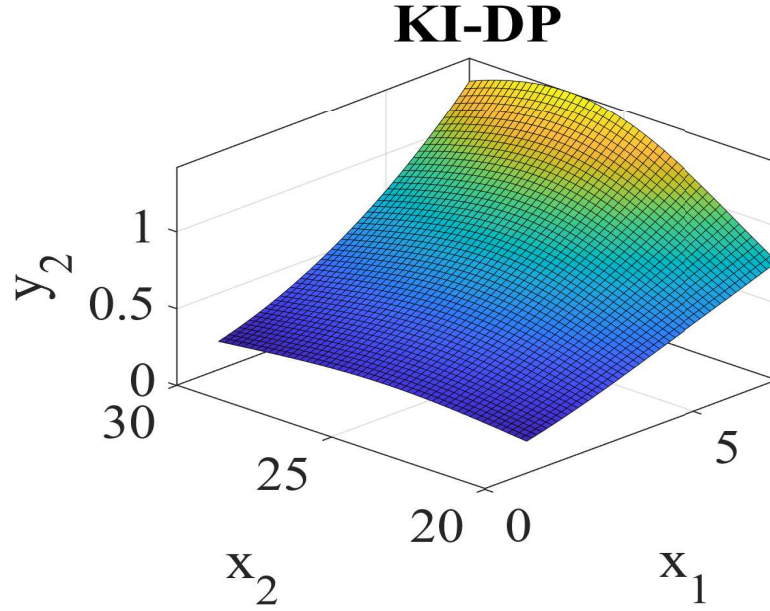


Figure 3.16 Response surface (KI) of trained surrogate model under T1 condition

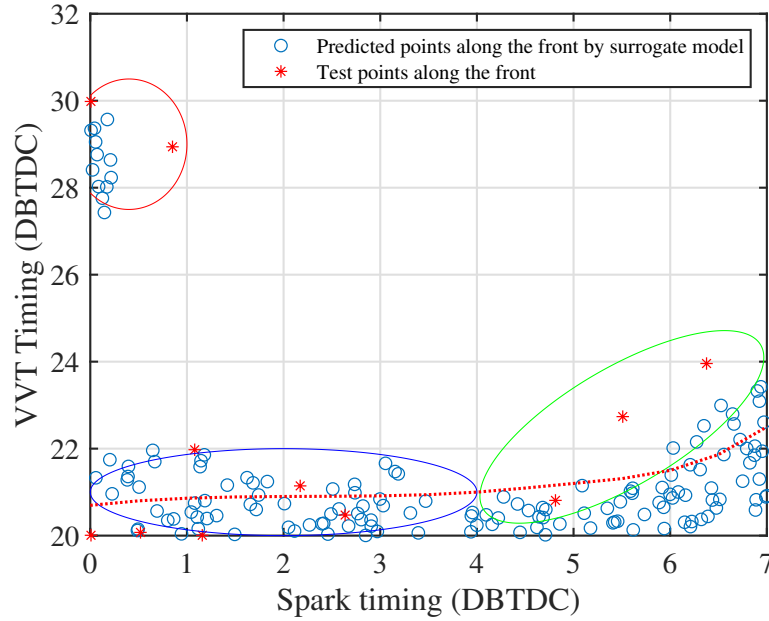


Figure 3.17 Design space of surrogate model

timing is relatively less advanced or more retarded, the VVT plays a main role in affecting the knock intensity with relatively high ISFC. As the spark timing advances, knock intensity is mainly affected by spark timing and is less sensitive to VVT with relatively small ISFC.

The comparison of sweep test and learning results is shown in Figure 3.19. An improved final Pareto front (shown as green crosses) can be observed with only a total of 40 test points (pink

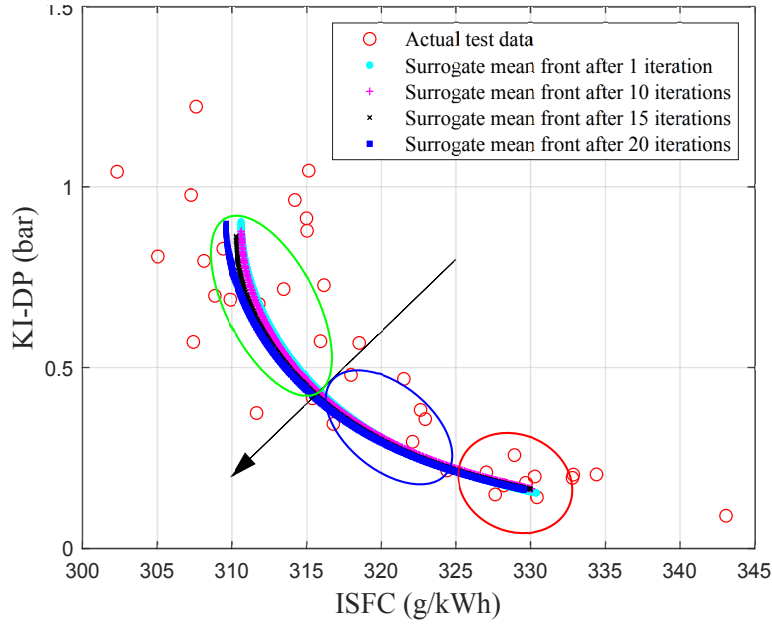


Figure 3.18 Objective space of surrogate model

squares) during the learning process, indicating the efficiency of proposed algorithm.

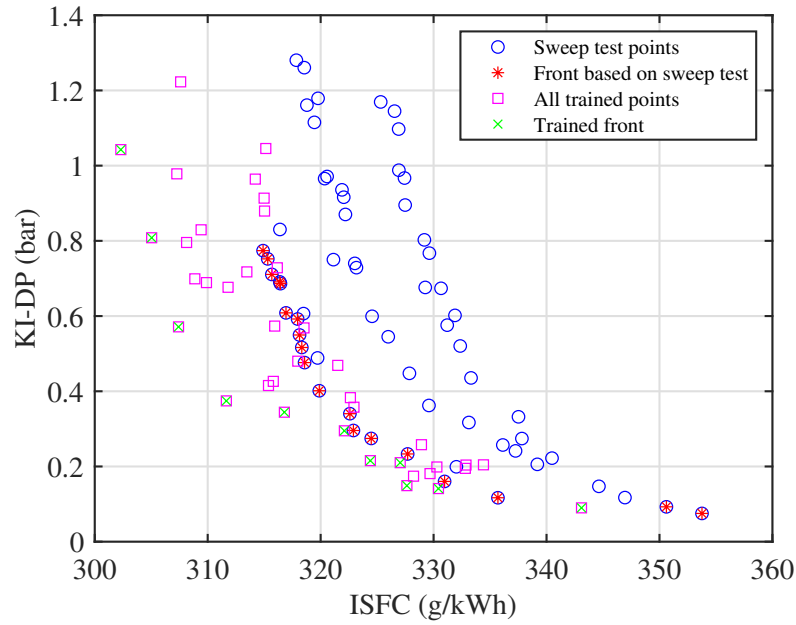


Figure 3.19 Comparison of sweep test and learning test results

After the learning process, the obtained surrogate Kriging models can provide both mean front (green curve) with upper and lower limits (pink curve) generated using the variance information from the Kriging model; see Figure 3.20.

The purple lines, corresponding to the mean value plus and minus three times of standard

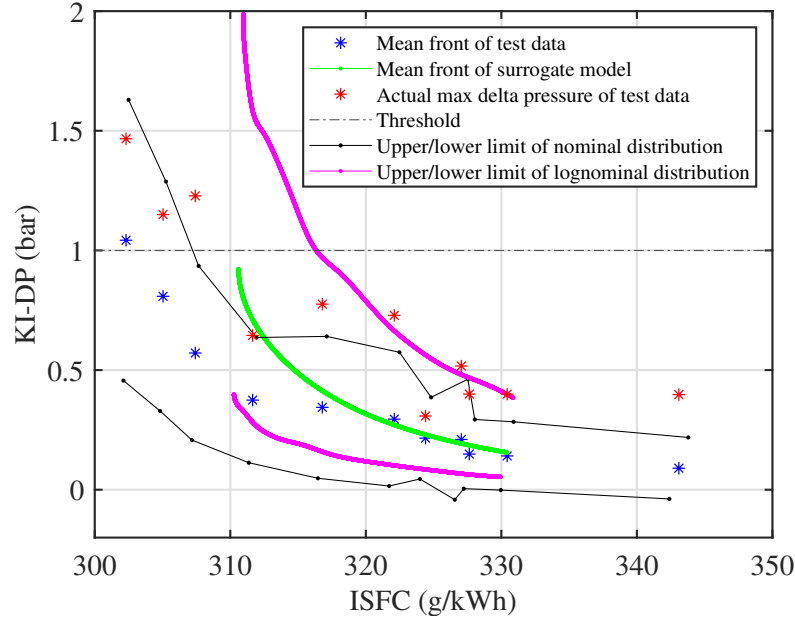


Figure 3.20 Train results with log-normal distribution under T1 condition

deviation for each point along the Pareto front, represent the upper and lower bounds with 99.7% confidence statistically. Without considering the desired KI limit, all the points along the Pareto front are the mean optimal responses with a tradeoff relationship between ISFC and KI. For a given KI upper bound, an intersection of the given KI bound and estimated upper bound can be found, and this point is projected back to the design space to find the optimal control parameters (spark timing and VVT), guaranteeing that the actual engine KI stays below the desired upper bound (1 bar in this case). As shown in Figure 3.15 and Figure 3.16, the smooth response surface of each objective can be approximated which is obviously not a multi-modal surface. The property of non-multimodal guarantees the existence of a unique mapping process. In other words, the design and objective spaces are one-to-one projections, therefore unique solutions of control parameters (spark timing and VVT) can be obtained for any given pair of ISFC and KI. Note that the upper limit from the log-nominal distribution predicts the maximum possible knock intensity (red star), which guarantees that using the corresponding control parameters the engine will be operated below the knock limit with the best fuel economy possible. However, if a Gaussian distribution model were used for Kriging model (shown in black curves), maximum KI prediction would be far below the actual one, indicating that using normal distribution would put the engine at a high risk due to

failed knock control.

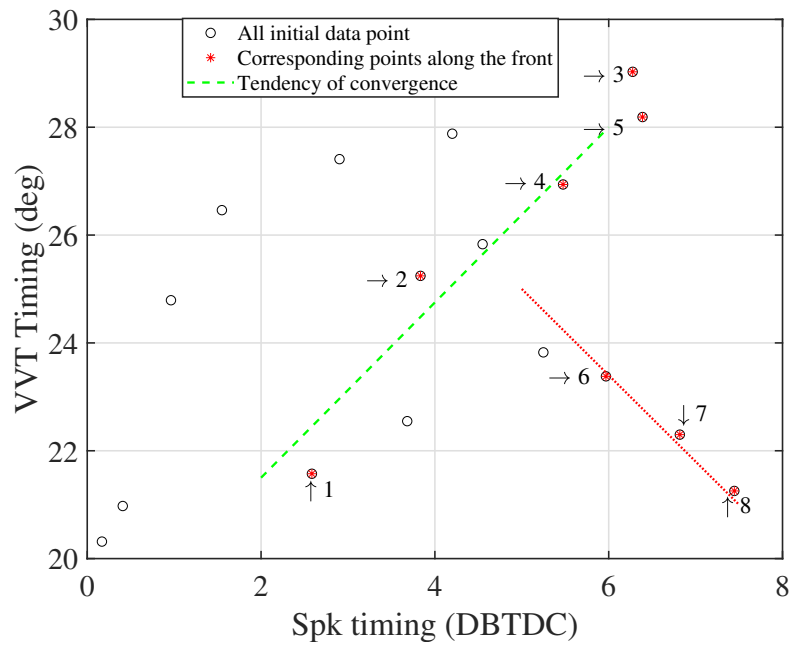
3.4.2 Test Results under T2 Condition

The proposed algorithm was also used for the engine operated under T2 condition for further verification. The proposed algorithm was implemented on the engine starting from an initial data set. The initial training data set is shown in design space and objective space in Figures 3.21a and 3.21b, respectively. Within these two plots, red points marked with sequential numbers are points along the Pareto front. It is found that, within this initial data set, there are 8 points located on the Pareto front out of 14 total points along the final Pareto front obtained after the learning process. Before starting the training process, the initial data set has provided a pattern for design variables and responses along the Pareto front. A linear combination of spark timing and VVT is noticed in the region formed by points 1 to 5, where the mean KI is lower than 0.1 bar. Beyond this point, points 6 to 8 form a new region.

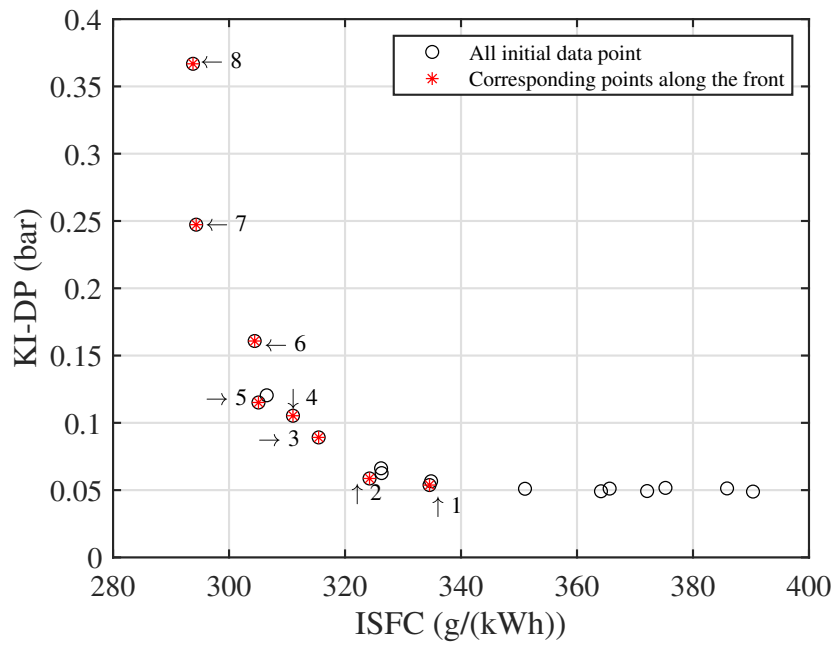
By using the learning algorithm, more points along the Pareto front are expected to be found through exploration and exploitation processes and final results are shown in Figures 3.22a and 3.22b. Note that, all legends in the two figures are the same, but are only presented in Figure 3.22b. The same total iterative test budget of 40 was used for T2 condition. There are six more points found along the Pareto front through the iterative learning process, where blue squares are the infilled points by optimizing acquisition function at each iteration, and green stars are new points found along the final Pareto front after 20 iterations. Under this test condition, the overall KI is lower than that under the T1 condition. It is also found that less exploration is guided in the high KI region by the acquisition function.

Under the T2 condition, after completing the learning process, the same approximation process is used to acquire smooth response surfaces for KI and ISFC; see Figure 3.23 and Figure 3.24, respectively.

Accordingly, the Pareto front of the surrogate model and corresponding design space variables are shown in Figures 3.25a and 3.25b. As shown in these circled areas, under this operational condition, the effect of spark and intake valve timings to engine knock is obvious. Especially, on

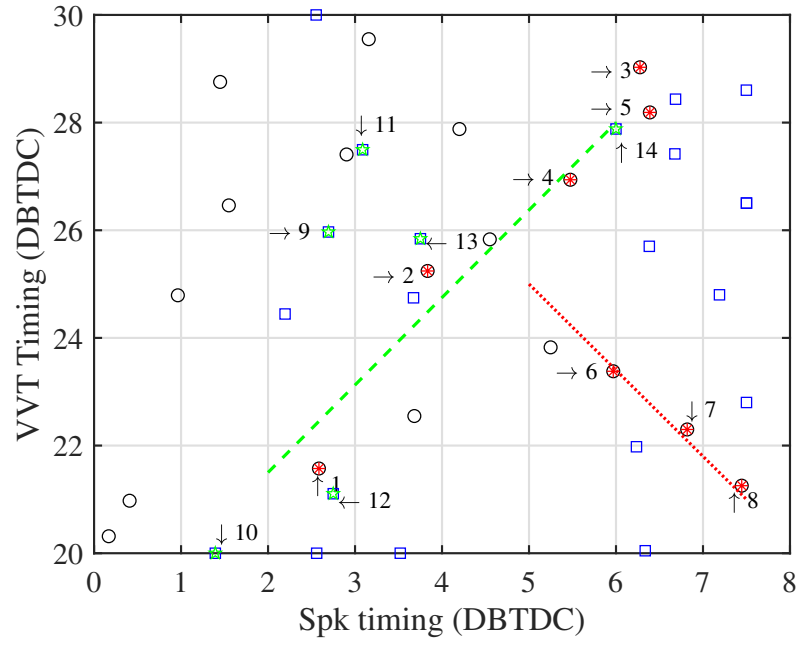


(a) Design space of initial training data set (T2)

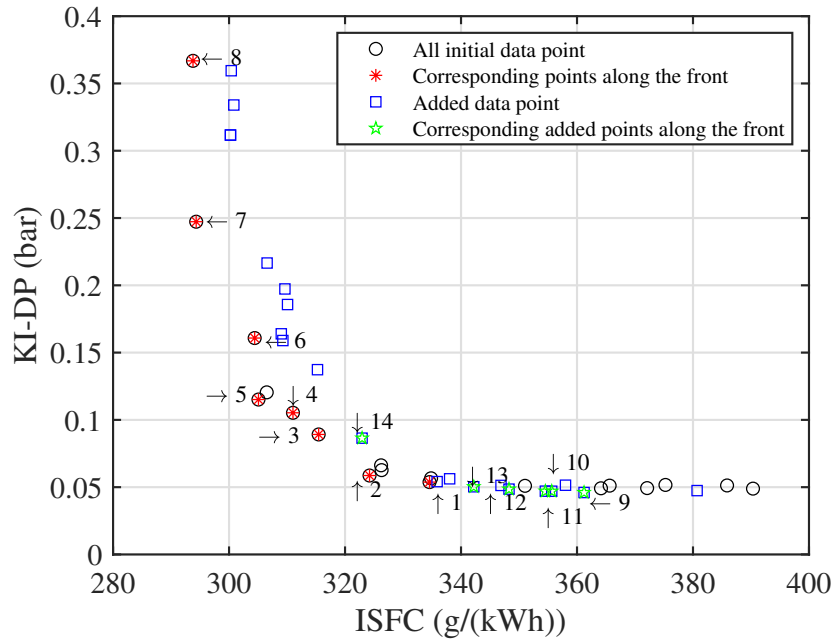


(b) Two objectives of initial training data set (T2)

Figure 3.21 Test results using initial training data set (T2)



(a) Design space of final training data set



(b) Two objectives of final training data set

Figure 3.22 Test results ($CoV < 1.9\%$) using final training data set under T2 condition

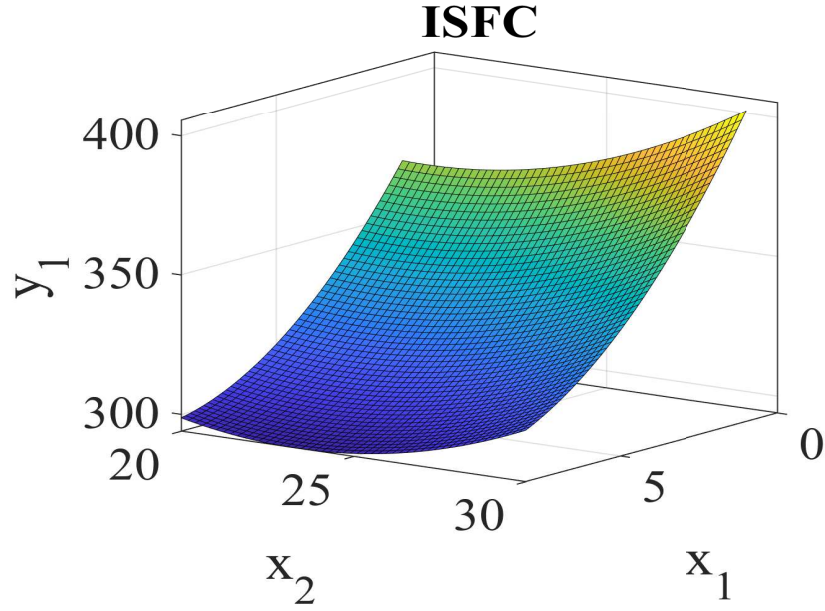


Figure 3.23 Response surface (ISFC) of trained surrogate model under T2 condition

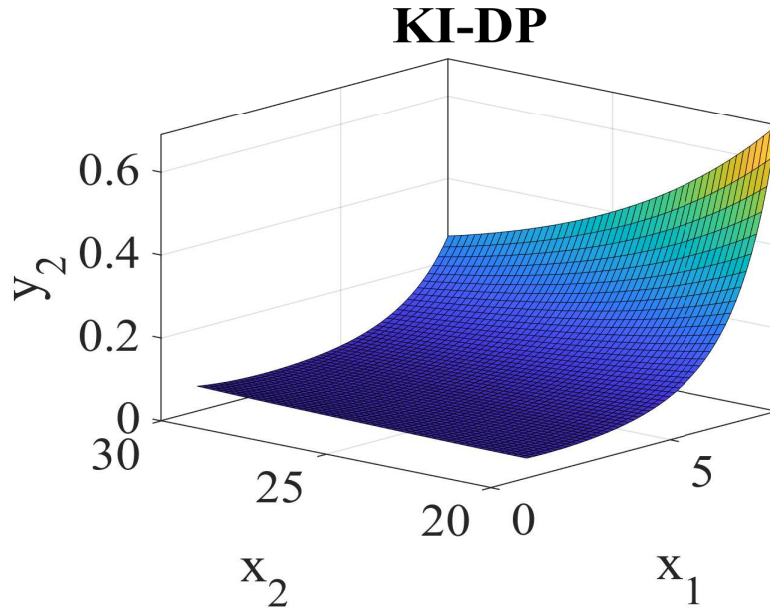
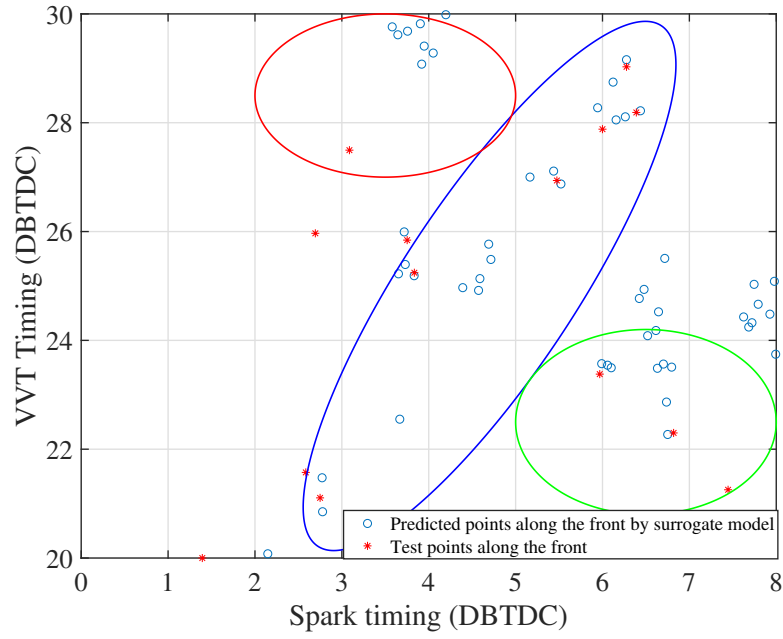


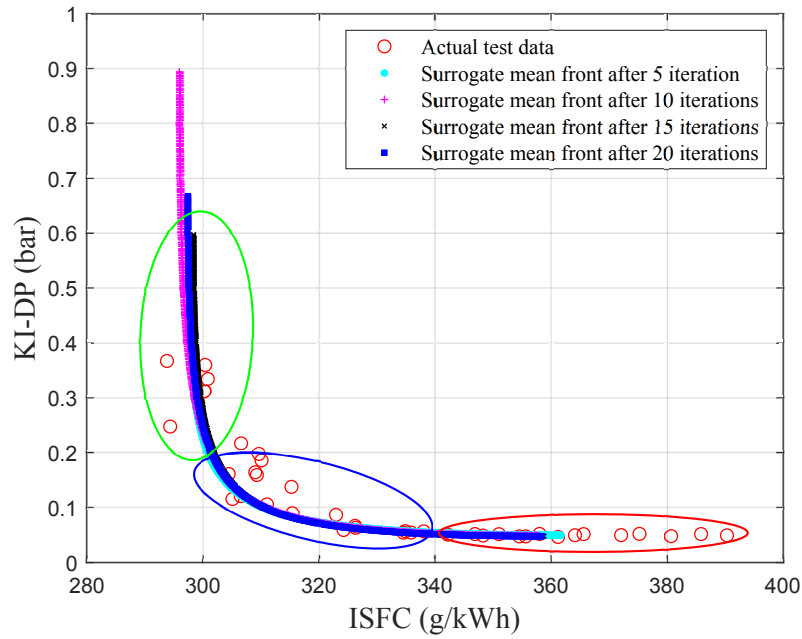
Figure 3.24 Response surface (KI) of trained surrogate model under T2 condition

the right side of the green circle, spark timing starts dominating the effect of engine knock. Since the overall KI is low under this operational condition, the change of Pareto front is small during the iteration process, but the converging trend is still noticeable.

Similar to the case under T1 condition, the obtained surrogate Kriging models can provide both mean front and upper and lower KI limits; see Figure 3.26, and the design space control variables



(a) Design space of surrogate model



(b) Objective space of surrogate model

Figure 3.25 Pareto front and corresponding design space of surrogate model under T2 condition

can be found by locating the intersection of desired knock limit (1 bar) and Kriging upper KI limit.

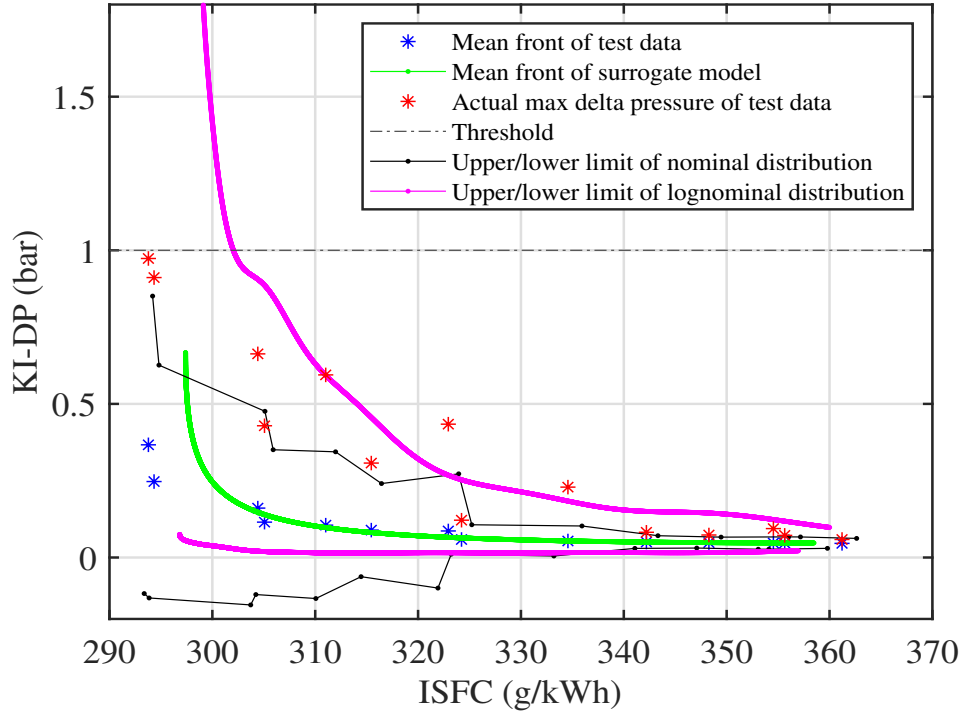


Figure 3.26 Train results with log-normal distribution at T2 condition

3.5 Test Verification with Predicted Knock Borderline

Both Kriging models obtained under two test conditions show that this algorithm is capable of predicting the knock upper limit relative to a given level of KI allowed for the best fuel economy possible with very low test budget. Additionally, the surrogate-model with predicted variance can be used to generate a safe operational region in the design space summarized in Table 3.4. The 100-cycle KI values when the engine operated under both recommended conditions are shown in Figure 3.27.

Table 3.4 Engine operation condition recommended by the learning algorithm

Test #	T1 condition	T2 condition
Speed (rpm)	1200	1500
Load (Nm)	90	80
Recommended VVT (DBTDC)	21	25.8
Recommended Spark Timing (DBTDC)	4	7

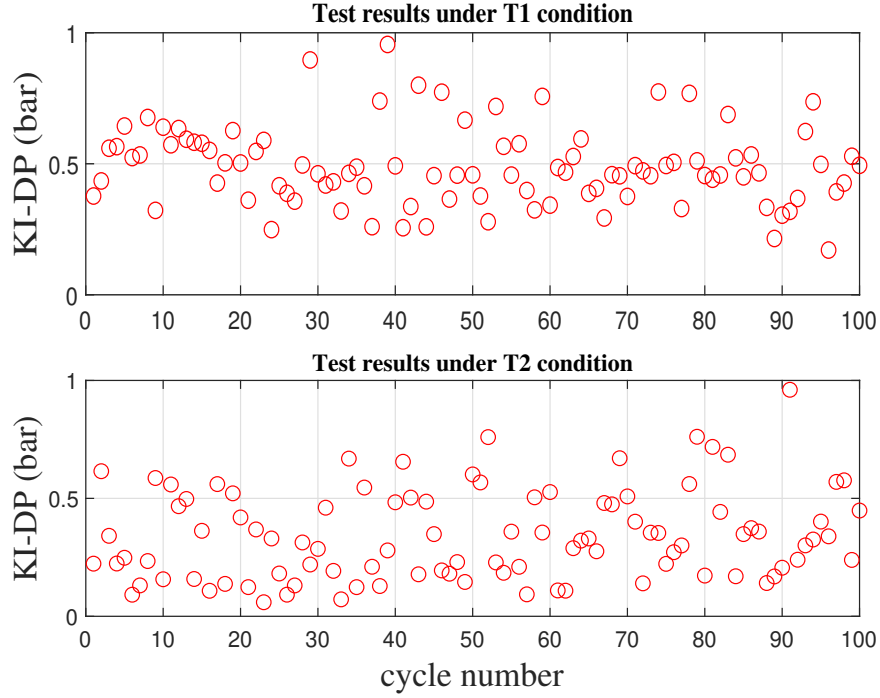


Figure 3.27 Cycle-to-cycle KI results with recommended design parameters

These test results show that the recommended design control variables make engine KI close to the knock borderline limit but strictly below the desired knock limit. If a Gaussian distribution of KI were used, the selected design control variables would be too aggressive to cause engine damage. On the other hand, using the current selected design control variables as shown in Table 3.4, under Gaussian distribution assumption, the upper bound of KI predicted by the Kriging model would be only 0.6 and 0.5 bar under T1 and T2 conditions, respectively, which are much smaller than actual test data of 1 bar.

3.6 Conclusions

In this chapter, the Bayesian optimization algorithm with the dual-surrogate-model was implemented by running simulation studies on both test problem and combustion model firstly. All the simulation results show the efficiency of optimization algorithm and the effectiveness of these proposed modifications for the stochastic cases. The significant cost reduction for performing the multi-variable optimization under a stochastic environment is confirmed. The concept of using the predicted statistic information for knock prediction is demonstrated clearly. Further verification of the proposed learning algorithm in real-time is successfully conducted on a practical engine

bench test for predicting engine knock borderline. Note that some modifications are made in the learning algorithm to fit the knock borderline application, for example, dual-surrogate-model structure and log-normal to Gaussian distribution mapping. A dual-surrogate-model structure is proposed to make it possible for solving stochastic information of Kriging model with unknown noise characteristics; and a distribution mapping is proposed to convert the actual log-normal KI distribution to Gaussian one so that Bayesian optimization algorithm can be used. For performing experimental optimization, inspired by the multi-variable Bayesian optimization, two objectives are chosen for optimization, namely knock intensity (KI) and ISFC (indicated specific fuel consumption). Experimental results indicate that the active learning logic embedded in this algorithm efficiently predicts the optimum operational conditions with a significantly reduced test budget of 40 points, compared with more than 200 points using the conventional sweeping process. Using the Bayesian learning algorithm, a mean Pareto front can be generated along with its upper and lower limits obtained based on estimated variances. By using the final trained surrogate model, a smooth Pareto front can be found. Compared the predicted statistic features with a predefined desired knock limit, the associated optimum control variables in the design space can be found with the best engine efficiency possible. Since the approximated surrogate model is cheap to evaluate, it has the potential to be updated online for compensating engine aging and operational condition variations.

CHAPTER 4

ONLINE UPDATING OF SURROGATE MODEL

After a few verification simulations, the learning algorithm was finally implemented onto the engine dyno bench in the previous chapter. Through the iterative intelligent search, the algorithm is converged to the Pareto front formed by two competing objectives (KI and ISFC). The approximated surrogate models provide a promising smooth response surface. This offline training process helps to obtain optimum design parameters. These parameters need to be adaptive to the environmental changes and engine aging.

The offline trained Kriging model is a Gaussian process model providing both mean and variance. Due to the simplicity of data-driven model, it can be updated in real-time to compensate for engine aging and operational environmental changes such as fuel type, temperature, humidity, etc. The main content of this chapter is to design an online updating structure utilizing the offline learned surrogate model. The simulation study will be implemented under both LabView and Matlab environments, which is the same software platform used for real-time tests. The updating process can be simplified by conducting the principal component analysis (PCA) to find dominated control variable. The results of PCA indicated that the spark timing is the most sensitive factor affecting the Pareto front used to determine the borderline knock limit. A likelihood ratio controller is adopted to capture knock variations caused by engine aging and other factors. The goal is to update the upbound of Pareto front according to the calculated likelihood ratios. Thus after mapping the predicted values from Gaussian back to log-nominal distribution, the knock borderline limit can be located on the intersection between predicted KI upper bound and given KI limit.

This chapter initially talks about the concept of likelihood ratio control followed by the proposed online updating scheme. Sections 2 and 3 present configurations of co-simulation environment and the simulation results utilizing a stochastic SI engine knock intensity model with the log-nominal distributed mean and variance obtained experimentally. Physical engine dynamometer tests under different scenarios are presented in Section 4. Finally, conclusions are drawn in Section 5.

4.1 Likelihood Ratio

As described in the previous chapter, a desired KI limit is defined. When the pressure oscillation magnitude is above 1 bar, it is assumed to be an undesired knock event. Therefore the knock events have a binomial distribution which is a type of distribution that has two possible outcomes, knock or no knock. Given a target knock probability, p , the probability of obtaining k knock events within n cycles of observations can be calculated using Equation (4.1) below. For our project, the target probability is $p = 0.3\%$ which corresponds to the knock probability on the basis of log-normal distribution and 3-sigma rules. However, this probability is not a direct tool for knock control or the online updating process [68].

$$P_n(k) = \binom{n}{k} p^k (1-p)^{(n-k)} \quad (4.1)$$

The concept of likelihood ratio, $L_n(k)$, was first introduced for knock control in 2010 [68]. The likelihood ratio delegates the ratio between likelihood level evaluated using the target knock rate and the maximum likelihood estimation of the current knock probability. To be specific, it is used to evaluate the ratio between expected knock probability ($p = 3\%$) obtained from the offline learning algorithm and the observed knock probability ($p_{obs} = k/n$) obtained from the actual knock rate, which maximizes the likelihood. It can be expressed as below.

$$L_h = \frac{L(p | N_k)}{L(p_{obs} | N_k)} = \frac{\binom{n}{k} p^k (1-p)^{(n-k)}}{\binom{n}{k} p_{obs}^k (1-p_{obs})^{(n-k)}} \quad (4.2)$$

where N_k is the observed knock event sequence; k is the number of knock events in N_k ; and n is the number of engine cycles. Later, two buffers will be defined and n would be the size of buffer. There are several likelihood ratio applications; see [68, 69, 70, 71]. Most of them only used one buffer that is directly used to decide the spark timing direction without considering the statistic characteristic of the knock combustion. Meanwhile the effect of short term environmental

changes and long term engine aging cannot be predicted in the Kriging model. We propose to use the concept of likelihood ratio to update the predicted surrogate models. A plot of the likelihood ratio as a function of engine cycle number is shown in Figure 4.1 using our target knock rate. The plot shows relationships between the likelihood ratio and cycle numbers when a pre-defined knock occurs. As expected, the likelihood ratio peaks when the observations match with the assumed knock rate. For example, the likelihood ratio is almost one at 300 cycles in the one knock event curve; when the cycle number is different from 300 cycles, the likelihood ratio would be lower than one. However, if the cycle number is smaller than 300 cycles, the knock rate or the observed knock probability would be larger than the target rate. Inversely, the knock rate would be lower. Thus, this observation inspires us to use both knock rate and likelihood ratio to design updating scheme for short term compensation using a dynamic-sized buffer, which will be discussed in the next section.

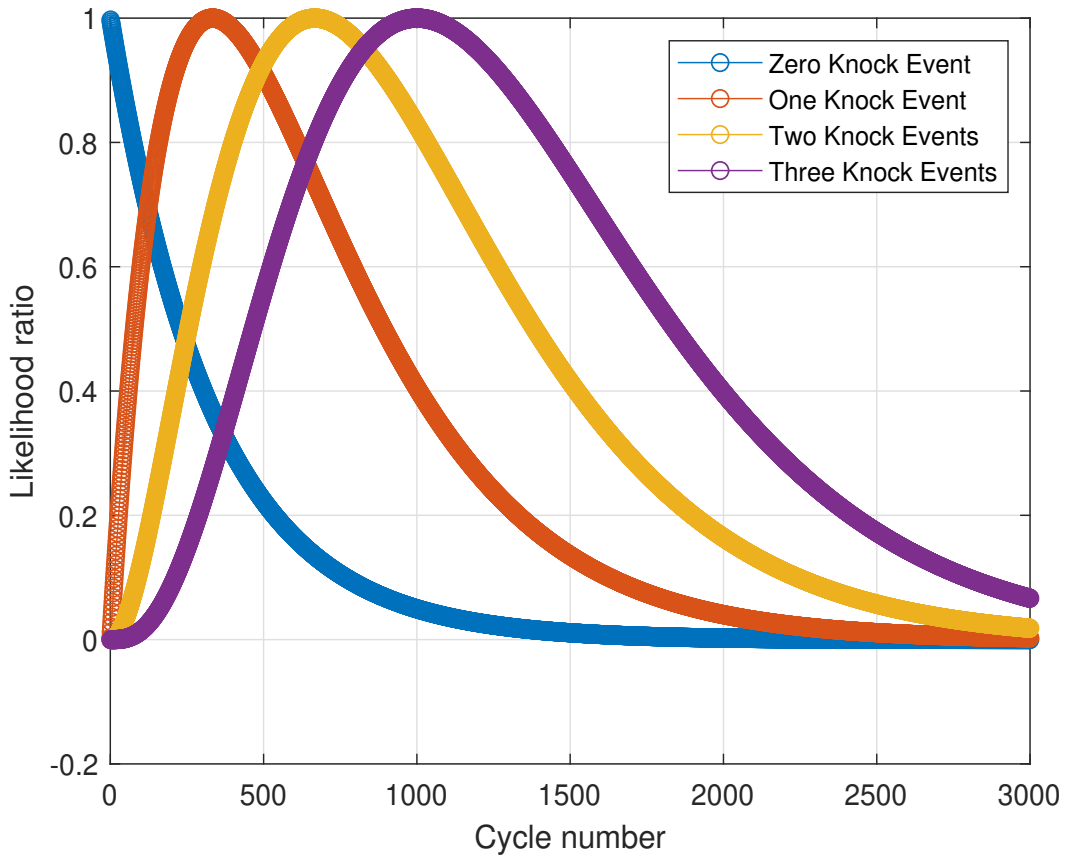


Figure 4.1 Likelihood ratio vs cycle numbers ($p = 0.3\%$)

A two-buffer design is proposed in this research. In order to better understand the effect of buffer

size, a plot of the likelihood ratio as a function of knock event is shown in Figure 4.2 regarding different buffer sizes.

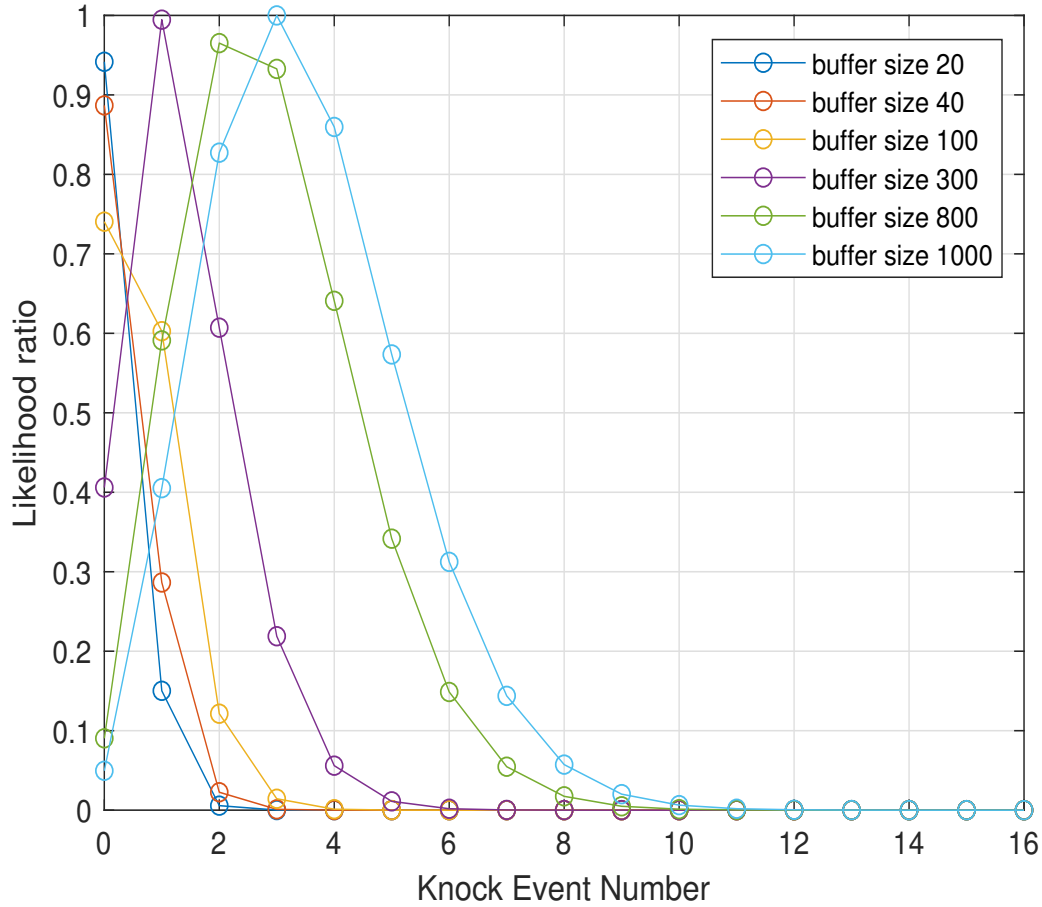


Figure 4.2 Likelihood ratio vs number of knock events ($p = 0.3\%$)

As indicated in the plot, when buffer size is small, the likelihood ratio decreases monotonically as the number of knock events increases. Note that, if the buffer size is too small (less than one hundred), the change of amplitude of likelihood ratio would be too small to identify. On the other hand, when buffer size is large, the monotonic relationship disappears, but the likelihood ratio is distinguishable. For our application, we want to have a relative quick response for short term effects and slow adjustment for long term factors. Therefore, two different buffer sizes are proposed for this purpose. One buffer will have a flexible or dynamic size which really depends on the time when one knock event occurs. As described before, if the knock occurs within a short period, the likelihood ratio would monotonically decrease to a relatively low value, which requires a quick

and large compensation action. The other buffer will have a relatively large buffer size (e.g., 6000 cycles) for a long-term consideration. To save the memory on the micro-controller, a special design would be used for this buffer. Since the target ratio is 0.3% and the turning point is clear when the buffer size is 300 cycles, thus a 300-cycle buffer is chosen as the base bin for the long buffer. Thus 20 bins together will be extended to a total 6000 cycles buffer. The detailed buffering and control strategies will be introduced in the next section.

4.2 Principal Component Analysis and Proposed Online Updating Scheme

4.2.1 Principal Component Analysis

After the offline training process, an optimum design parameter set was detected. The trained surrogate models present smooth response surfaces and have the potential to be used in real-time applications. Note that the upper bound of Pareto front is affected by several factors such as environmental conditions and fuel type. To accurately control the combustion process, the Pareto front and its upper boundary of surrogate model needs to be updated to adapt to those changes. To reduce computational load of real-time model updating, a further data analysis is required to reduce number of control parameters so that the key factor in the model is preserved. Principal component analysis (PCA) [72] is one of the widely used method. It computes the principal components and uses them to reduce the input dimension while preserving dominated statistical function, where only the first few principal components are going to be kept forming a lower-dimensional model.

In this research, there are two control inputs x_1 (spark timing) and x_2 (VVT). Before applying PCA, it's a standard practice to perform mean normalization on each dimension so that the control inputs should have zero mean, and comparable ranges of values using the equation below.

$$\tilde{x}_i = \frac{x_i - \frac{1}{n} \sum_{j=1}^n x_i^j}{\max(x_i) - \min(x_i)} \quad (4.3)$$

where x_i^j is the j^{th} point of the i^{th} dimension; and n is the total number of evaluation points. After this sort of data pre-processing, the PCA algorithm can be applied. First, compute covariance matrix Σ of design parameters, which is a symmetric positive definite matrix.

$$\Sigma = \frac{1}{m} \tilde{\mathbf{x}}^T \tilde{\mathbf{x}} \quad (4.4)$$

where $\tilde{\mathbf{x}}$ is an $n \times l$ matrix; n is the number of data points; and l is the number of input dimensions. Singular value decomposition (SVD) is used to obtain eigenvector matrix U and eigenvalue matrix

$$[U, S, U^*] = \text{svd}(\Sigma) \quad (4.5)$$

To reduce the input data dimension from l down to k , the first k column vectors of U are selected. The total variation, which is computed by equation below, is used to determine value of k .

$$\text{Var} = \frac{\sum_{i=1}^k S_{ii}}{\sum_{i=1}^l S_{ii}} \quad (4.6)$$

where Var denotes how much of the variance is retained or at most $1 - Var$ average squared projection error after dimension reduction. Normally, the error below 5% is acceptable for dimension reduction. For our case, PCA is implemented for these points along the Pareto front. When the dimension of spark timing is chosen, the total variations is 97.3% with a 2.7% projection error. Thus, only spark timing is the leading factor for online updating the surrogate Kriging model that is used to determine the borderline knock limit along the Pareto front in real-time.

4.2.2 Online Updating Strategy

The scheme used for the online updating process is going to be fully discussed in this section. Equation (4.7) is proposed to update the upbound of Pareto front estimated by the offline machine-learned surrogate models.

$$y = \beta (\alpha y_{\text{prem}} + (1 - \alpha) y_{\text{reg}}) \quad (4.7)$$

where y_{prem} and y_{reg} are outputs of offline trained Kriging models when the engine runs with regular (worst KI case) and premium fuel (best KI case), respectively; α and β delegate the outputs of each buffer and are updated based on the likelihood ratios. In our structure, two different buffer sizes are used to assist in updating the surrogate model under different rates. The fast buffer with a flexible size is used to interpret the surrogate models between the premium and regular fuel, and changes brought by other environmental changes. For long term consideration, such as engine aging or other slow changing phenomena, a slow buffer with a fixed size (relatively long) is used to update these outputs of surrogate-models.

As shown in Equation (4.7), output α is mainly dominated by the short-term effects such as fuel Octane level and temperature. Other factor or variation could cause α to fluctuate around certain point; and β is mainly caused by long-time effects such as engine aging. Without considering the possible engine aging, two boundary conditions (best and worst) need to be considered. $\alpha = 1$ means that the output is fully affected by the best surrogate-model (y_{best}), for instance, using fuel with a very high Octane level at low engine temperature; and output with $\alpha = 0$ is dominated by the worst model (y_{worst}), for example, using the extremely low Octane fuel at high engine temperature. When α is between 1 and 0, the engine is operated in the middle of best and worse knock conditions. For example, the vehicle is fueled with high Octane fuel mixed with low Octane fuel remained in tank so that the engine is in a transition from low Octane fuel to high one. Other short-term factors will cause α fluctuating within a narrow range. We need to extend our updating structure (see Equation (4.8)) to cover all scenarios, especially for the two boundary conditions. This extension has an advantage when we define the detailed updating strategies according to the likelihood ratios. That is, when α is changing (increasing or decreasing), output y would remain in the same updating direction (aggressive or conservative).

$$\begin{aligned}
y &= \beta \left(\text{Min}((2 - \alpha), 1.2)y_{\text{prem}} \right) & \alpha \geq 1 \\
y &= \beta \left(\alpha y_{\text{prem}} + (1 - \alpha)y_{\text{req}} \right) & 0 < \alpha < 1 \\
y &= \beta \left(\text{Max}((1 - \alpha), -1.2)y_{\text{reg}} \right) & \alpha \leq 0
\end{aligned} \tag{4.8}$$

After the KI upper bound along the Pareto front is updated, the new intersection point with the desire knock limit can be located which will be directly mapping back to the design space for selecting the corresponding spark timing. The updated design parameter will be forward to the engine controller.

4.2.2.1 Updating Logic for α

Based on the definition of likelihood ratio and proposed updating structure, the logics for updating α and β using the likelihood ratio information is presented below, respectively.

For short-term compensation coefficient α , a safety region is defined with a range of [200,500] cycles. Within this range, one light knock event is tolerable. Instead of a fixed size buffer, a dynamic

buffer is used, which can be assumed as a counter. The counter will accumulate the number of no-knock cycles. Once a knock is detected, α value will be updated based on the calculated likelihood ratio and knock rate from counted cycles. After this, the counter will be reset. The basic rule is that if likelihood ratio approaches unity, it indicates a relatively good match between the target knock rate predicted by the surrogate model and the observed knock events, so no control action (model-update) is required; if the ratio is small, a large mismatch exists, and therefore, a correction (model update) is required. The detailed updating scheme for α is shown in Figure 4.3.

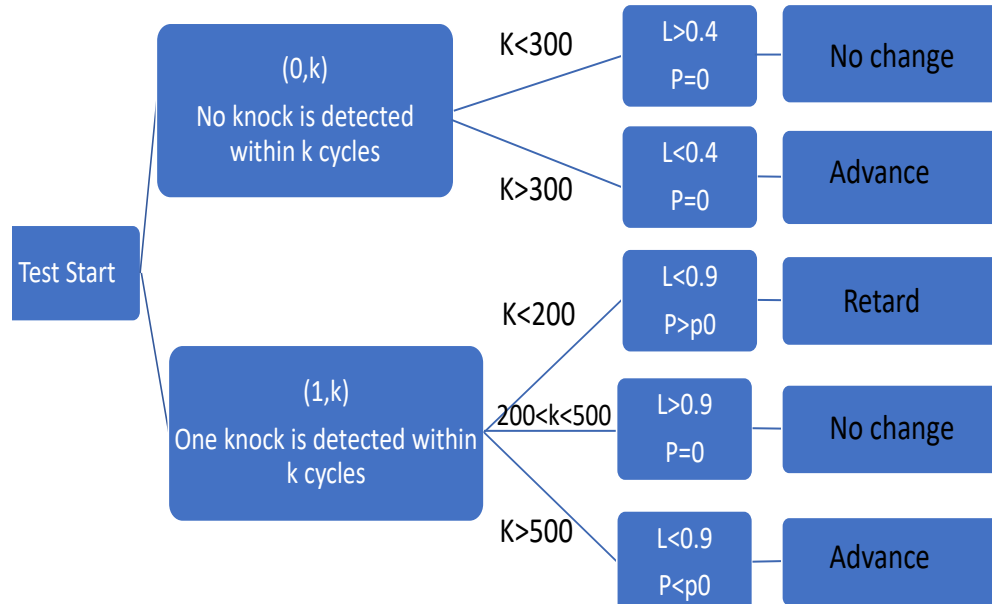


Figure 4.3 Updating scheme for α

For the counter output, it will only have two categories regarding the number of knock events: one knock or no knock. For no knock case, 300-cycle location is the dividing line. When the counted cycle is less than 300 cycles, the likelihood ratio is larger than 0.4 (see the monotonically decreasing tendency of likelihood ratio in Figure 4.1), but this does not give enough information about the possibility of knock in the future, so no action is required; on the other hand, if the cycle number is larger than 300 cycles, the likelihood ratio would be smaller than 0.4, which means that there is certain room to advance. The step size is controlled by the amplitude of likelihood ratio naturally. For one knock case, it includes three branches divided by the safety zone. If the counted number is within the safety region, there is no updating needed. Out of this region, it will

be divided into two scenarios. When the cycle number is less than 200 cycles, the likelihood ratio would be less than 0.9, but knock rate p will be larger than the target one p_0 . Inversely, when the cycle number is larger than 500 cycles, the knock rate would be less while the likelihood ratio is also less than 0.9. Therefore, the sign of difference between the actual knock and target rates can be used to control the updating direction (best or worst) which can also be called aggressive or conservative. To realize this updating logic, the following equation (4.9) is used to update the α as required. Otherwise, α will remain unchanged.

$$\alpha_i = \alpha_{i-1} + \text{sgn}(p_0 - p)(1 - L)K_{\text{gain}}\Delta KI \quad (4.9)$$

where i and $i - 1$ are the current and previous steps; p_0 is target knock rate and p is the actual knock rate; L is the counter likelihood ratio; K_{gain} is a calibration parameter; sgn represents the sign function used to determine updating direction; and ΔKI is the difference between measured and estimated means.

4.2.2.2 Updating Logic for β

The updating logic for β is different from that for α . In order to efficiently record history data for the long-term engine aging effect, a 300-cycle buffer with a 20 first-in-first-out (FIFO) recorder is shown in Figure 4.4.

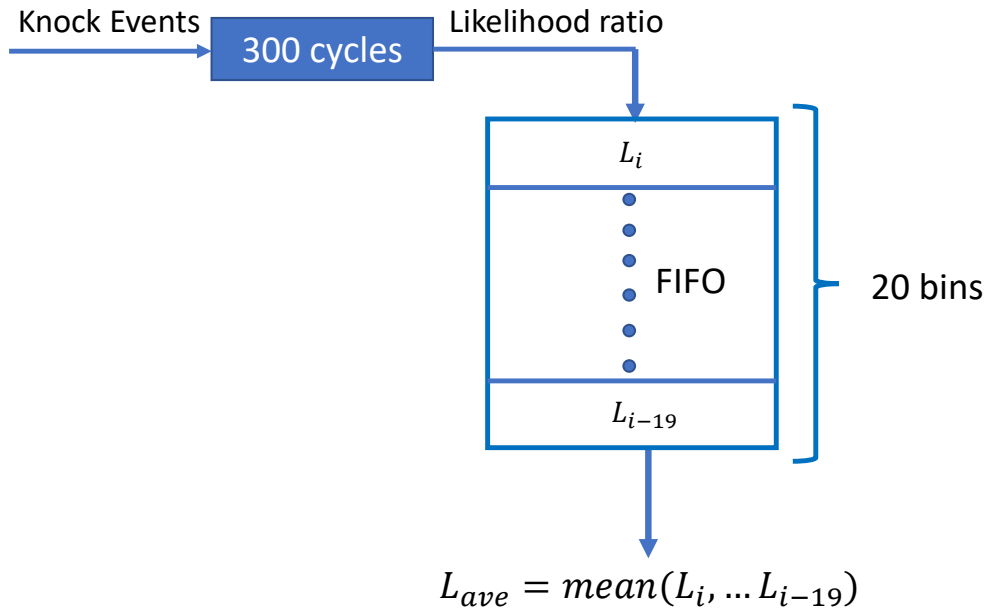


Figure 4.4 Updating scheme for β

The fixed size (300 cycles) buffer is used to calculate the likelihood ratio every 300 cycles. The calculated likelihood ratio is stored in the short FIFO recorder updated every 300 cycles. The average of total 20 values will be calculated. Note that engine aging and other slow changing factors could change the knock tendency. For example, deteriorate piston rings reduce the effective engine compression ratio, and/or increased Carbon deposits on piston could also increase the effective compression ratio. Therefore the direction of β can be decided by the difference between actual mean and baseline mean of KI, which can also drive the Pareto front to the related direction. The following updating equation is proposed and β is updated based on the averaged likelihood ratio and likelihood threshold.

$$\begin{aligned}\beta_i &= \beta_{i-1} + \text{sgn}(KI_{mean} - KI_{base}) \\ &\quad (L_{thr} - L_{ave})K_{gain2} \quad L_{ave} < L_{threshold} \\ \beta_i &= \beta_{i-1} \quad L_{ave} \geq L_{threshold}\end{aligned}\tag{4.10}$$

where i and $i - 1$ are the current and previous steps; KI_{mean} is the actual KI mean; KI_{base} is the baseline KI mean; L_{ave} is the averaged likelihood ratio of the FIFO; K_{gain2} is a calibration parameter; and L_{thr} is a threshold.

4.3 Simulations Study in LabVIEW and Simulink Using Proposed KI Statistic Model

The co-simulation configuration in LabView and Simulink provides an effective approach to validate the proposed algorithm before its implementation for real-time application. Note that the co-simulation structure, in terms of signal communication and execution of updating scheme, is the same as that for real-time engine tests.

4.3.1 Configurations for Co-simulation Environments

Instead of using a complex combustion model to generate engine knock, a statistic knock simulator is used to generate the KI-DP. This simulator uses a lookup table to generate random knock intensities with the log-normal distribution for given mean and variance. Inputs to the lookup table are the spark timing and intake valve timing (IVT) and outputs of the table are mean and variance of the KI-DP that matches with the trained model based on experimental data. Both mean and variance

will be sent to the knock generator to shape the log-nominal distribution. Even though this is not a true engine combustion model, it can delegate the interested knock variations from the stochastic perspective. A block diagram of this arrangement is shown in Figure 4.5.

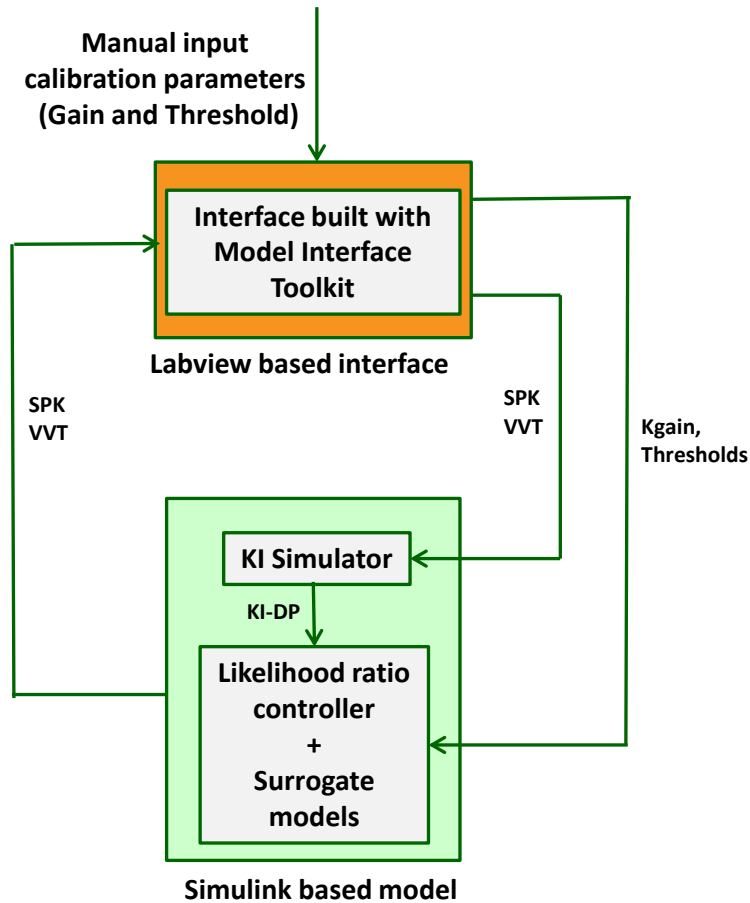


Figure 4.5 Configuration of co-simulation in LabView and Simulink

As shown in Figure 4.5, the updating logics for α and β , surrogate models and knock simulator are modeled in Simulink. The NI LabView provides the communication interface. A model interface toolkit is used to compile the model, and the generated dynamic link library (dll) file provides the communication between LabView and Simulink. The Simulink based model accepts the control commands from LabView interface. Based on the spark timing and IVT, the knock simulator generates the corresponding random KI-DP. Calculated model coefficients α and β updates the upper boundary of surrogate in the aggressive or conservative direction so that both spark timing can be updated accordingly. This co-simulation process provides a good foundation

for the future tests since the proposed updating strategy is programmed in the same NI LabView system.

4.3.2 Simulation Results

The simulation study starts with the knock simulator adopting the mean and variance from regular fuel (Octane = 87) to simulate the worst knock case. Now the short-term variation factor α would be updated in the simulation environment when the mean and variance used in the knock simulator changes. In order to activate the updating process early, a low predefined KI upper bound (0.95 bar) for the likelihood ratio control is set in the LabView. A 3000-cycle simulation is conducted and the associated results are presented in Figures 4.6, 4.7, and 4.8, respectively.

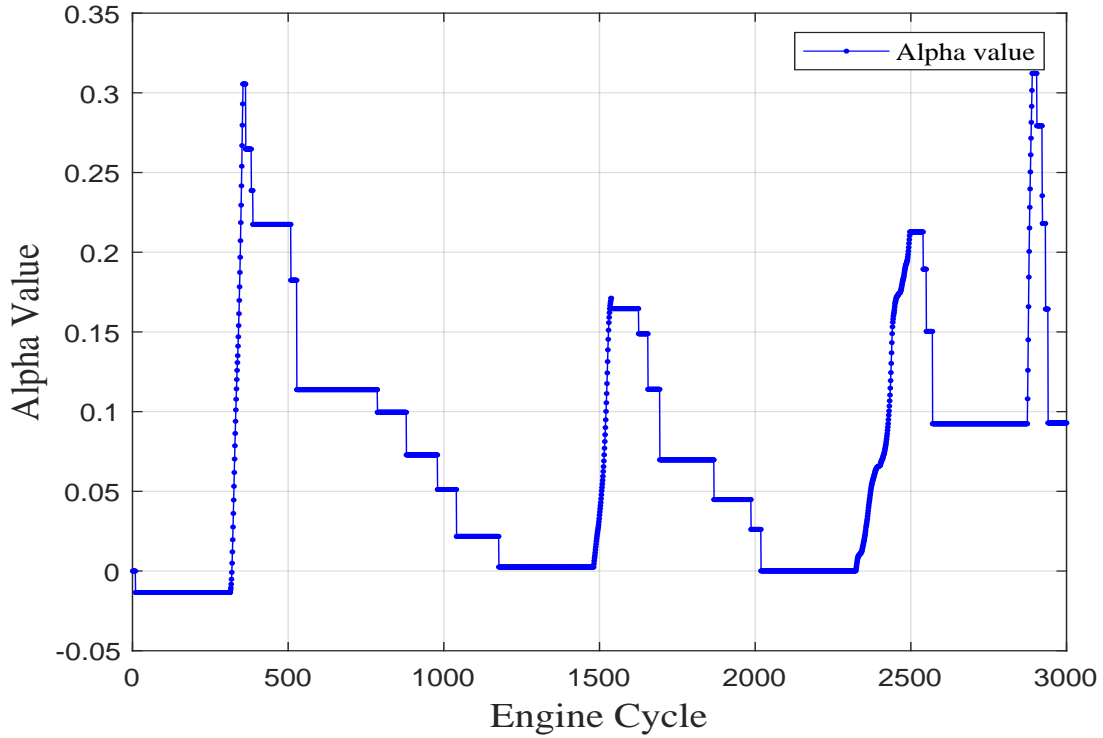


Figure 4.6 α value during the simulation with regular fuel

The updated α adjusts the surrogate-model (4.7) due to change of knock occurrences with a threshold set at 0.95 bar for the likelihood controller. Spark timing is updated accordingly and has a consistent tendency as α . It shows that most of time, spark timing locates at a relative advanced position. Although, α fluctuates during the simulation since KI-DP passed the predefined lower bound, most of actual KI-DP values stay within our desired KI limit (one bar). There are about 11

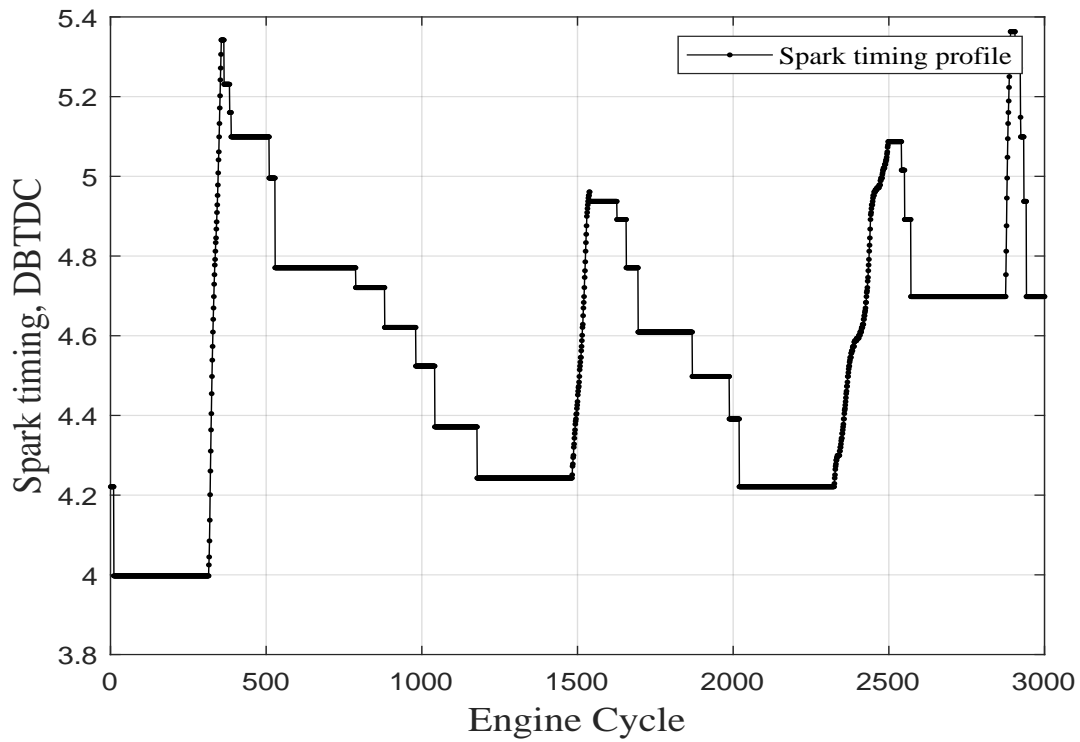


Figure 4.7 Spark timing during the simulation with regular fuel

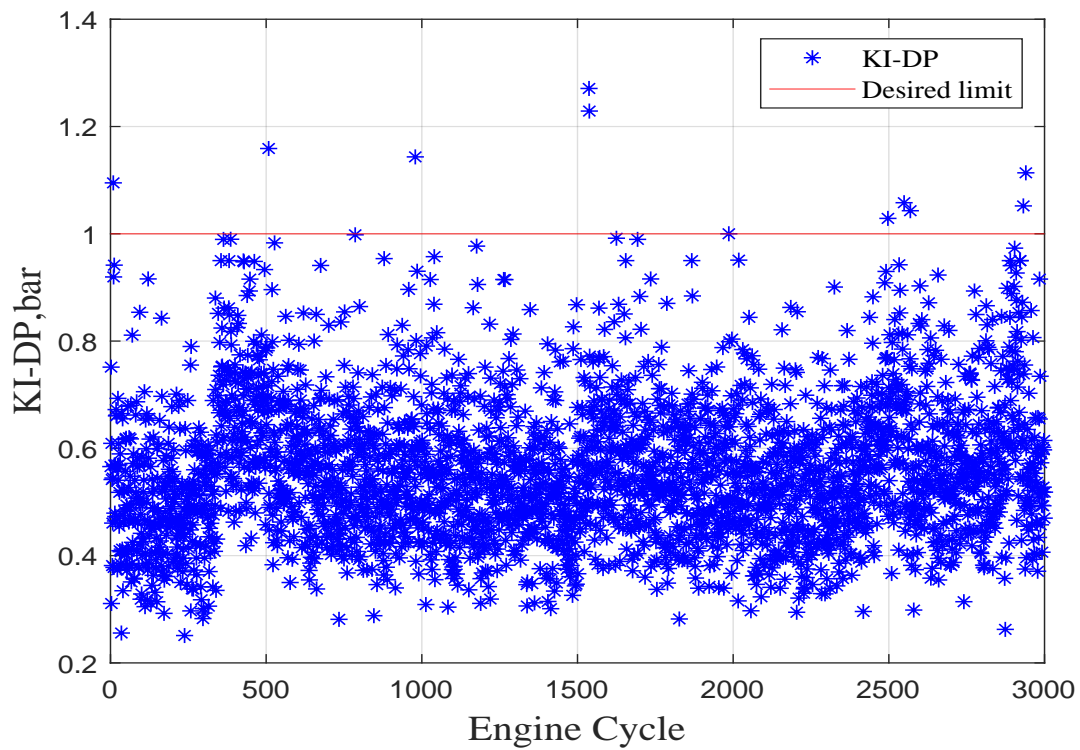


Figure 4.8 Delta pressure (KI) during the simulation with regular fuel

points slightly beyond 1 bar limit over 3000 cycles, which is very close to our target tolerance of 1/300.

The same simulation study was also conducted with the table of knock simulator using mean and variance from high Octane fuel to simulate the best knock case. Note that the initial α is set to 0, which corresponding to regular fuel (the worst case). The simulation results are shown in Figures 4.9, 4.10, and 4.11, respectively. It shows that the spark timing gradually converges to a relative advanced position and the knock rate is also within the tolerable region.

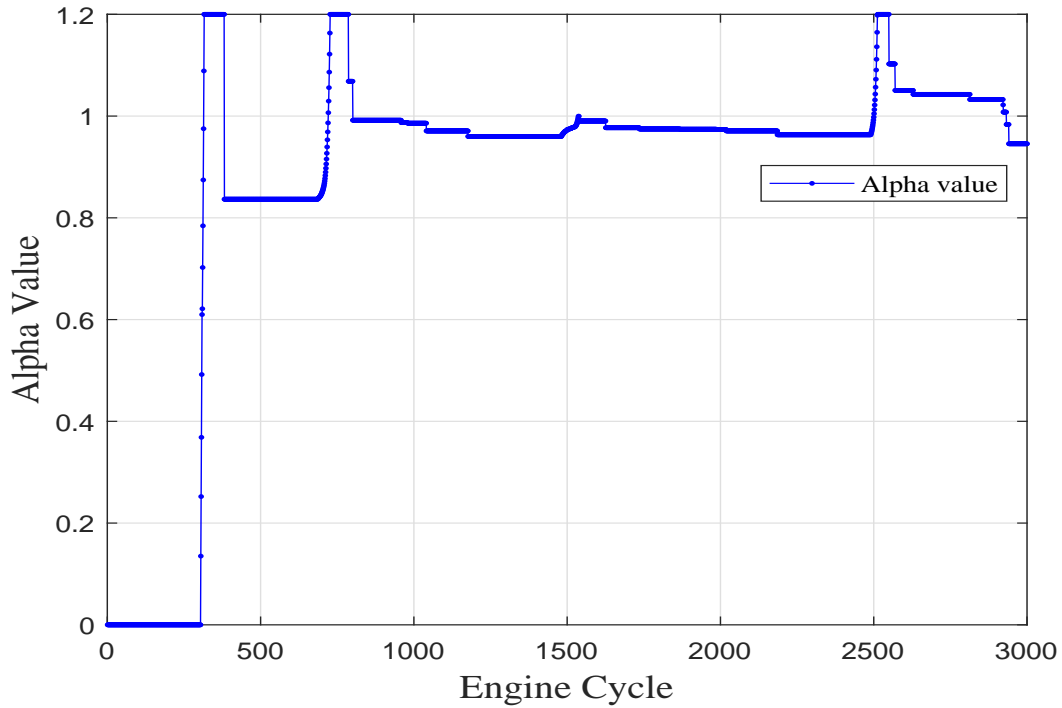


Figure 4.9 α value during the simulation with premium fuel

These simulation results show a successful updating process for α using the proposed updating logic. The obtained profile shows that the updated surrogate-model is able to operate the engine at a more advanced spark timing position. Meanwhile, the resulting knock delta pressure (KI-DP) is within the range of desired distribution and tolerance with only light knock events occurred. For long-term effect, a threshold of β is used to trigger the long-term compensation. A 4% increment of KI is added to the knock simulator when the combustion cycle reaches 7000 cycles to simulate a slight increment of engine compression ratio due to heavy carbon deposits. The simulation results are shown below in Figures 4.12 and 4.13. It shows that β value gradually converges to a new value

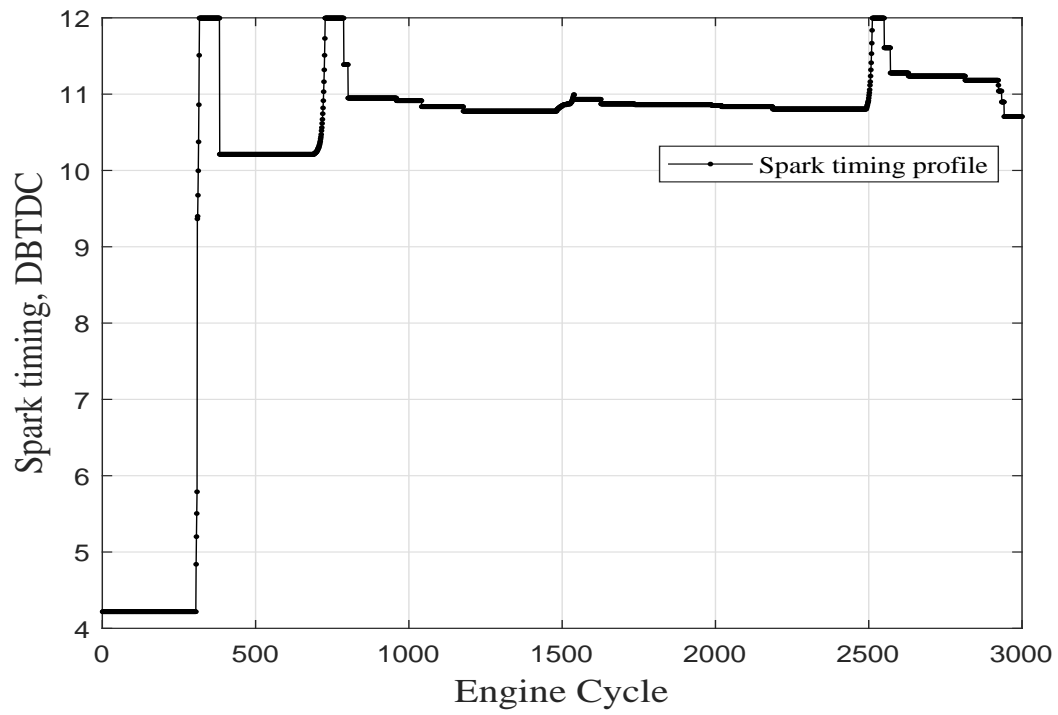


Figure 4.10 Spark timing during the simulation with premium fuel

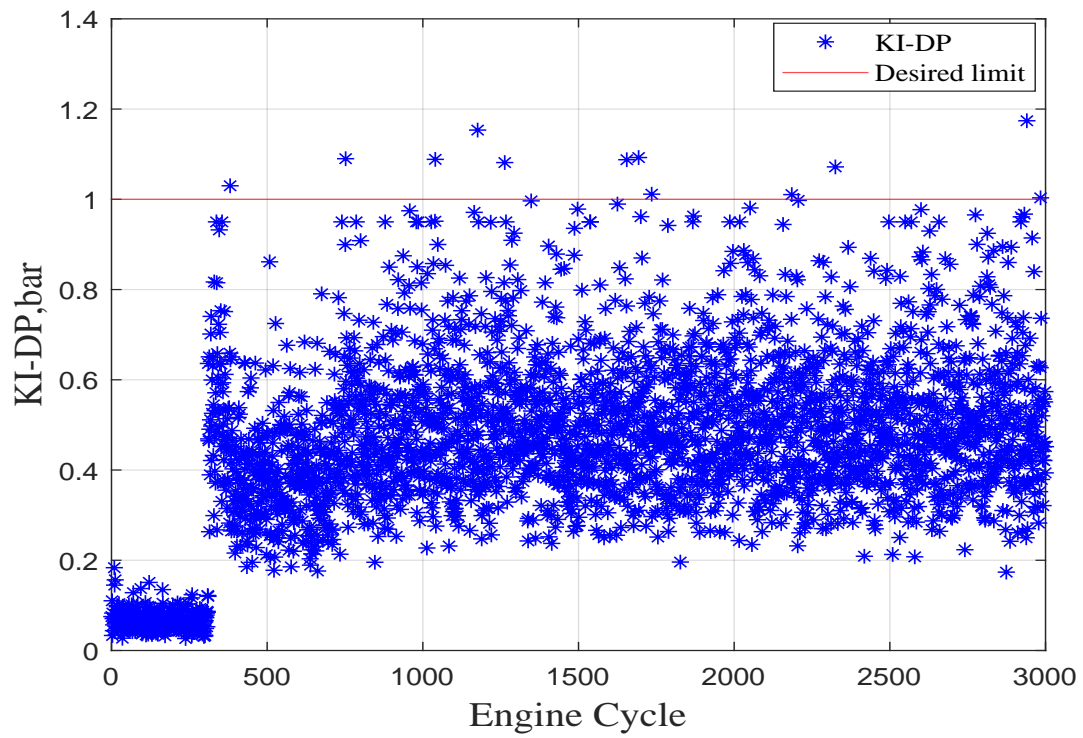


Figure 4.11 KI-DP during the simulation with premium fuel

after the 7000 cycles. After that, spark timing gradually converges to a lightly retarded region due to increased knock level caused by increased compression ratio.

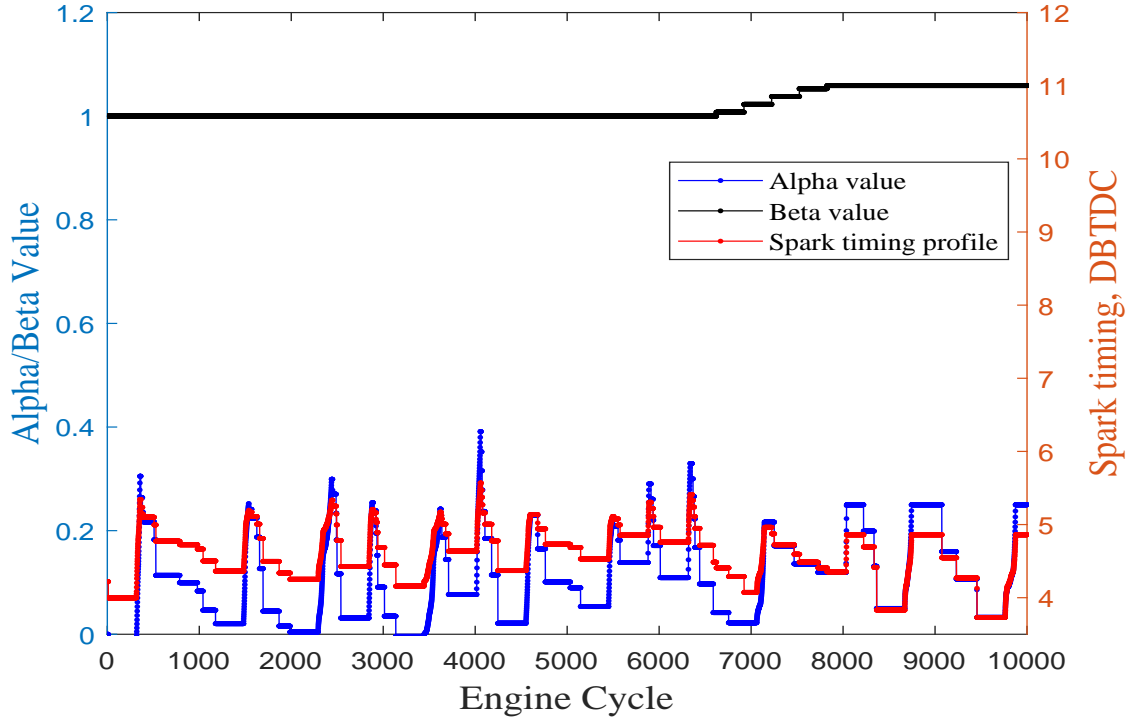


Figure 4.12 Simulated spark timing with long-term affects

4.4 Engine Bench Test Results

The proposed online updating algorithm is also verified on MSU engine test bench. Note that the overall test configuration is almost the same as the co-simulation one except that the KI simulator is replaced by the physical engine. The KI-DP can be obtained by bandpass filtering the in-cylinder pressure signal in real-time. The overall architecture is the same as the configurations shown in Figure 3.8. The engine was running under different scenarios at a fixed speed and load condition (1200 rpm and 90Nm) summarized in Table 4.1.

Table 4.1 Test scenarios for online updating at fixed speed and load condition

Case Number	Fuel type	KI Threshold for likelihood ratio calculation, bar	Coolant Temperature deg C	Test Purpose
1	regular	0.95	90	short term effect
2	regular	1	90	short term effect
3	premium	1	90	short term effect
4	mixed	0.95	90	transient case
5	premium	0.95	90 to 95	long term effect

As listed in Table 4.1, a fixed fuel type is used to verify the short-term compensation from Cases

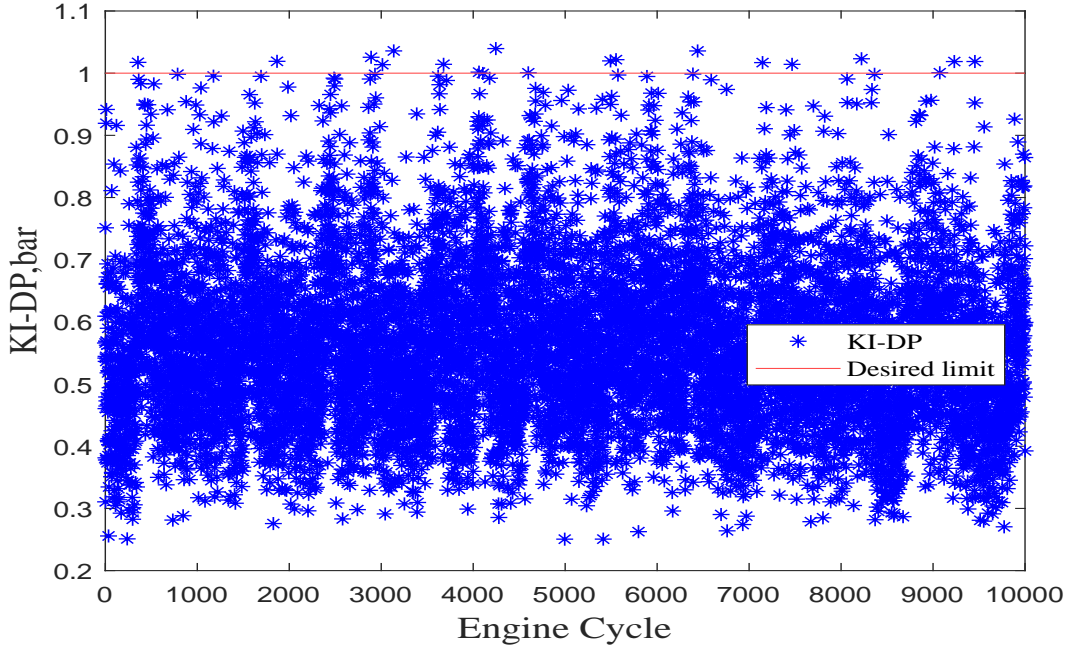


Figure 4.13 Simulated KI-DP with long-term affects

1 to 3, where the change of system operational environment causes the deviation from the offline trained surrogate model (baseline). Among them, a different KI threshold is tuned and verified when these tests were running using regular fuel in Cases 1 and 2. A smaller value means an earlier updating action is expected. The results are shown below in Figures 4.14 and 4.15.

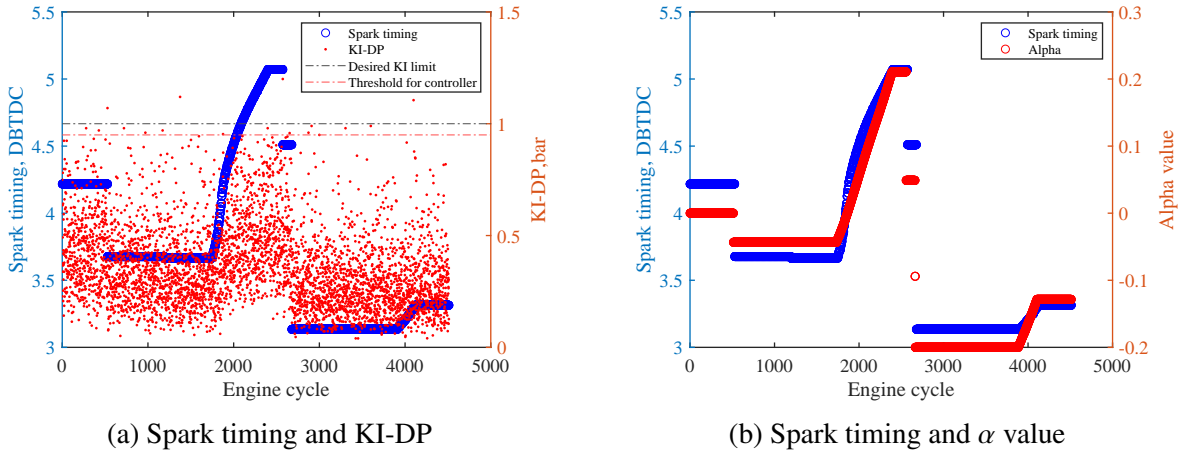
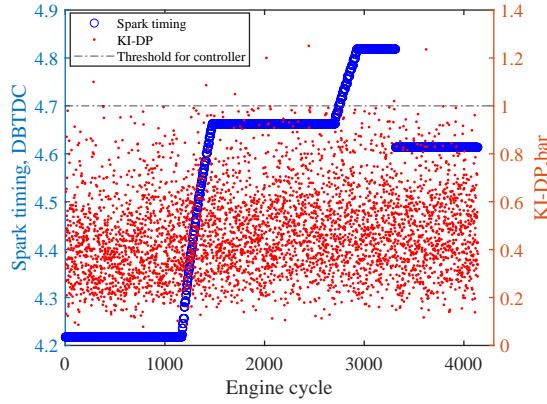
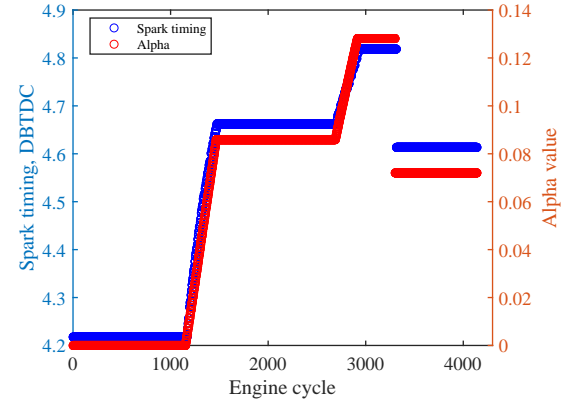


Figure 4.14 Engine test using regular fuel with small control threshold

When the KI threshold is set to a small value, the model updating takes action early. Accordingly, this adjusts the spark timing to a relative retard (conservative) location mostly. In general, the KI is controlled within the desire knock limit. As observed, the spark timing updates with the calculated



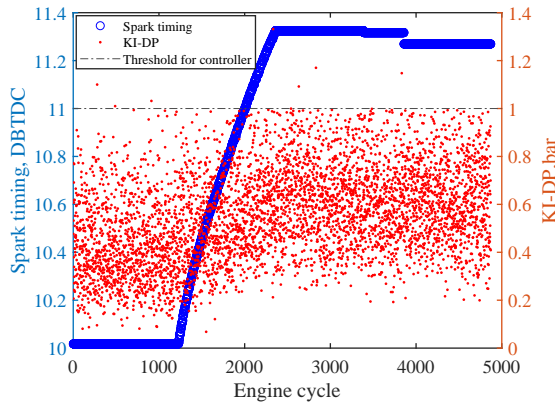
(a) Spark timing and KI-DP



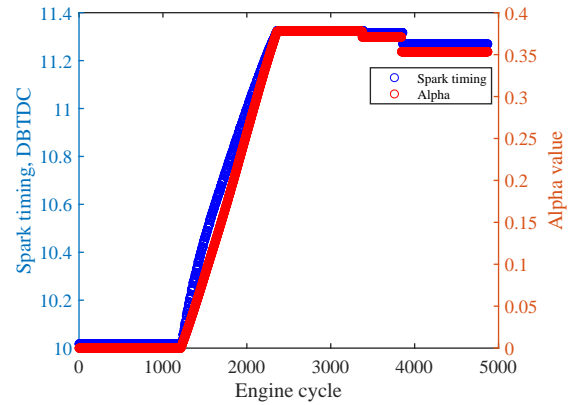
(b) Spark timing and α value

Figure 4.15 Engine test using regular fuel with large control threshold

α . Increase the threshold close to one bar, the overall KI is higher than before. Accordingly, the controller is expected to update the model in a relative aggressive way. The test results show that the spark timing is at an advanced position most of the time, and the KI is still under the desired limit with a few events above the threshold. While using the premium fuel, the threshold was kept at 1 bar for the likelihood ratio controller. The test results are shown below in Figure 4.16. It is obviously that engine has stable combustion status and the spark timing finally converges to a stable position. The α also varies less after it converges.



(a) Spark timing and KI-DP



(b) Spark timing and α

Figure 4.16 Engine test using premium fuel with large control threshold

The proposed updating scheme is able to not only adapt to environment changes with a fixed fuel, but also help to updating control parameters when fuel transmits from one to another. As shown in Case 4, the premium fuel is added when the fuel tank is about 10% full during the test.

As shown in Figure 4.17, the gradual decreasing process of KI is noticeable. It took about 1500 cycles (2.5 minutes) to advance the spark timing, and α is updated during the transition process from regular to premium fuel. Note that the KI threshold was set as 0.95 bar for this test.

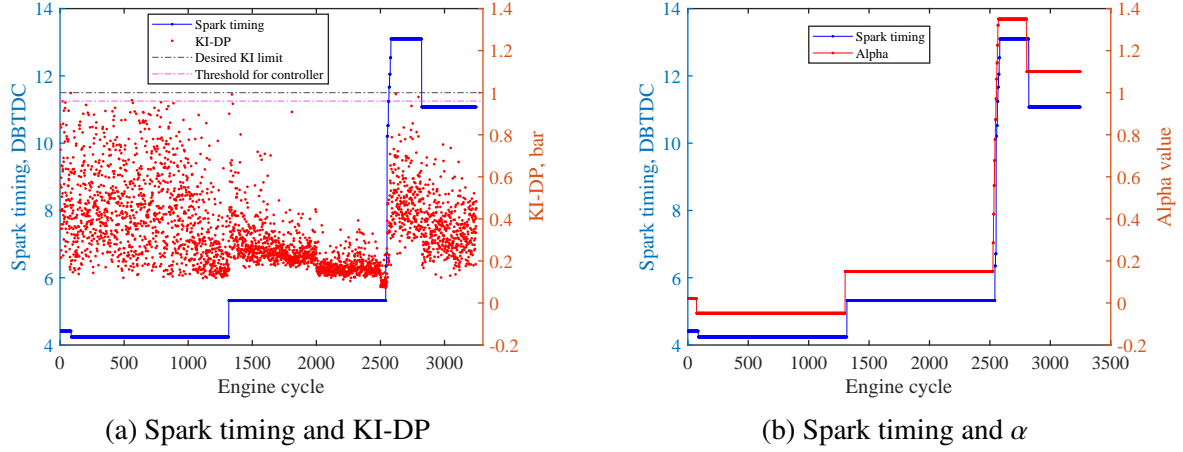


Figure 4.17 Engine test with transient fuel mixed

In order to verify the long-term compensation strategy, the control valve of coolant system was adjusted so that the coolant temperature is increased from 90 to 95 deg C during the test. This is used to simulate the engine aging process with increased knock intensity. The test results are shown below in Figure 4.18, indicating that when long-term compensation starts work, the spark timing converges to a relative retarded position. Both α and β values converge, and spark timing is updated accordingly.

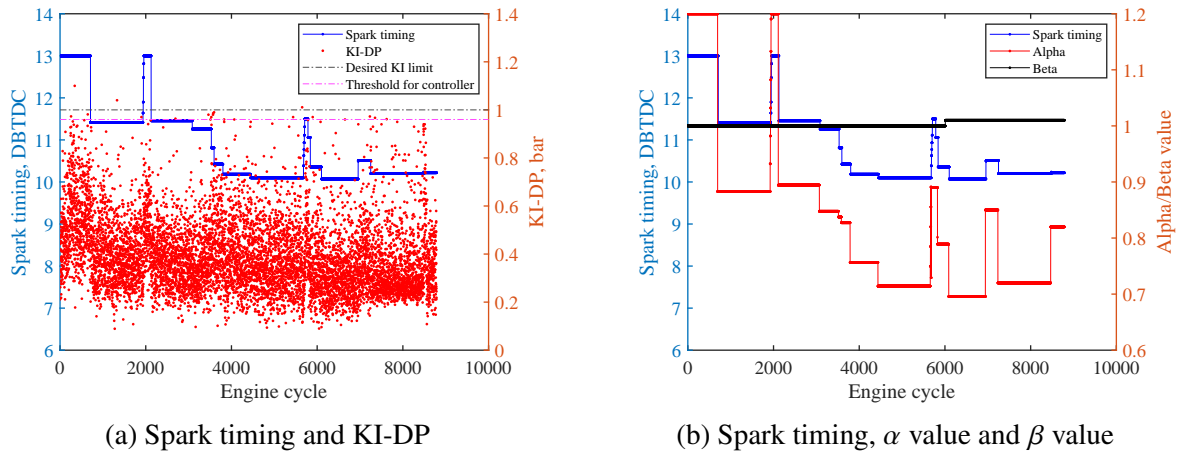


Figure 4.18 Engine test with high coolant temperature

The proposed online updating strategy for knock feedforward control is successfully validated

on a physical engine under five different predefined test scenarios in real-time assisted by CAN communication between original ECM and knock control module. By changing the fuel type, coolant temperature and KI threshold, the proposed online knock control architecture demonstrated its effectiveness for both short- and long-term compensation at quasisteady- and transient-status. The provided calibration parameters in the updating scheme could also be used for other engine applications since they contain information such as engine aging and its current operational condition.

4.5 Conclusions

This chapter proposes an online updated feedforward knock control architecture to adapt to engine aging and varying operational environment, where the worst and best surrogate models are trained offline. During the offline training process, a multi-variable Bayesian learning algorithm is modified for knock borderline application with a dual-surrogate-model structure and log-normal to Gaussian distribution mapping. The proposed offline training algorithm is implemented into a physical engine test bench for predicting engine knock borderline limit along with associated control parameters. After the offline training, well-trained best-and-worst-case Kriging models are obtained with smooth response surfaces. Utilizing the potential of Kriging model, an online updating scheme is designed with the assistance of principle component analysis and updated likelihood ratio controller. Both short- and long-term coefficients are designed to compensate fast environmental change and slow engine aging, respectively. A statistic knock simulator is implemented for co-simulation in LabView-Matlab for verifying proposed online updating scheme. The same configurations are used for performing engine experiments. Five different typical scenarios are tested to verify the proposed engine knock borderline control structure and online updating scheme. The simulation and test results indicate the ability of updating the Kriging model in real-time to adapt to engine aging and its current operational condition. This online updating control scheme extends the potential usage of offline trained Kriging models. The recommended baseline control parameters provide an improved knock borderline limit for real-time feedforward control. It is easy to implement and can be further integrated with other closed-loop knock control algorithms.

CHAPTER 5

CYCLE-BASED LQG KNOCK REGULATION USING IDENTIFIED EXHAUST TEMPERATURE MODEL

In the previous two chapters, control algorithms are developed based on statistic properties of the history data. They do not provide any immediate control action to reduce cyclic variations caused by the inherent combustion dynamics. In this chapter, the designed control strategy will reduce the knock intensity variations while maintaining the knock intensity (KI) below a desired level for spark ignition engines when it is operated close to or at knock borderline. The exhaust temperature data is studied for analysing the knock combustion variations. Several model-based prediction methods have been conducted for complex exhaust temperature dynamics, such as estimation of temperature drop in the manifold using mean value model [73, 74] and estimation of transient gas temperature [75]. However, these models have two main limitations: a) mean-value model without cycle-to-cycle variations, making it not suitable for cycle-based knock compensation and b) high-complex and nonlinear physics-based models that cannot be used for model-based real-time control. To simplify the physics-based modeling process, several data-driven system identification methodologies [76, 48] came to the rescue in these scenarios, which generate black-box linear models based on a set of input-output data, where deviation (δ) of spark timing, associated deviation (δ) of KI and exhaust temperature are used for our application. The exhaust temperature measurement is important since its fluctuation represents the current-cycle KI variations and will be utilized as control feedback. A detailed study was conducted for measuring exhaust temperature using three different thermal couples [77, 78], which shows that the sealed thermal couple can only obtained averaged temperature due to its large time constant, and only bare-wire thermal couple is fast enough to catch accurate pulsations of exhaust temperature. In our research, a bare-wire thermal couple is installed on the exhaust manifold close to exhaust port of interested cylinder.

5.1 Q-Markov COVER and LQG Introduction

This section introduces the main system identification method (PRBS q -Markov COVER) and control scheme (LQG) used in this dissertation, which develops the guidance for system identification

open-loop tests and for closed-loop tests of cycle-based knock compensation.

5.1.1 Q-Markov COVER Algorithm

The q -Markov COVER algorithm, originally developed for model reduction using white noise test signal, which is unrealizable accurately in experiments [79]. A pseudo random binary signal (PRBS) [79] was proposed to replace the white noise as the excitation and the corresponding output responses could be observed. The input-output data set is used to generate the first q Markov parameters, along with covariance matrices, of the original system (for example, engine dynamics in this paper), and a reduced-order linear model with the equivalent Markov parameters can be generated using a series of well-proved formulations. PRBS is a periodic signal with magnitude a and period of $2m$, where m is the length of PRBS defined as $m = 2^n - 1$ and n is the order of PRBS. The exhaust temperature itself have a relatively slow response to the PRBS signal (δ spark timing), so a 7th order PRBS signal is selected so that the duration of PRBS is long enough to cover the temperature variation. The detailed description of PRBS signal can be found in reference [48]. The generated PRBS is then used to evaluate the system outputs, which are then used for system identification.

A brief discussion of the state-space representation for an unknown linear dynamic system is described below. A detailed linear algebra derivation of PRBS q -Markov COVER can be referred to reference [48] for thorough understanding. A discrete asymptotically stable system can be represented below in a standard state-space format:

$$\begin{aligned} x(k+1) &= Ax(k) + Dw(k) \\ y(k) &= Cx(k) + Hw(k) \end{aligned} \tag{5.1}$$

where $x \in R^{n_x}$, $w \in R^{n_w}$ and $y \in R^{n_y}$ are state, input and output vectors, respectively. Considering X to be the solution of the following Lyapunov equation:

$$X = AXA^T + DWD^T \tag{5.2}$$

where W is the covariance matrix of white noise input w . The system Markov parameter matrices ($H_i, i = 1, 2, \dots, q$) and the associated covariance matrices ($R_i, i = 1, 2, \dots, q$) are defined as:

$$H_0 = H \quad , \quad H_i = CA^{i-1}D \quad i = 1, 2, \dots \tag{5.3}$$

$$R_i = CA^iXC^T + H_iWH^T \quad i = 0, 1, 2, \dots \quad (5.4)$$

Note that these parameter matrices are dependent on system matrices A , B , C , and D . They will be calculated based on the input-output data and its auto-correlations. Thus inversely, the results will further be used to generate the system state-space model.

Since PRBS vector w is a periodic signal with period $2m$, its system steady-state responses are also periodic with the same period as defined below in equations (5.5) and (5.6).

$$x_s^j(k+i) = A^{k+i}x_s^j(0) + \sum_{n=0}^{k+i-1} A^{k+i-1-n}De_ju(n) \quad (5.5)$$

where $x_s^j(0)$ is the steady-state periodic response at time zero. The associated system output responses are

$$y_s^j(k+i) = Cx_s^j(k+i) + HDe_jw(k+i) \quad (5.6)$$

for $i = 0, 1, \dots, q-1$, where e_jw selects the j^{th} PRBS input w and e_j is the j^{th} standard base vector.

With all equations above from (5.1) to (5.6), we have

$$Q_q y_q^j(k) = Q_q O_q x_s^j(k) + Q_q H_q w_q^j(k) \quad (5.7)$$

where Q_q is a block diagonal matrix with q blocks of positive definite output weighting matrix Q , and

$$y_q^j(k) = \begin{bmatrix} y_s^j(k) \\ y_s^j(k+1) \\ \vdots \\ y_s^j(k+q-1) \end{bmatrix}, \quad w_q^j(k) = \begin{bmatrix} e_jw(k) \\ e_jw(k+1) \\ \vdots \\ e_jw(k+q-1) \end{bmatrix} \quad (5.8)$$

$$O_q = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{q-1} \end{bmatrix}, \quad H_q = \begin{bmatrix} H_0 & 0 & \cdots & 0 \\ H_1 & H_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ H_{q-1} & H_{q-2} & \cdots & H_0 \end{bmatrix} \quad (5.9)$$

Taking the expectation over one PRBS period from both side of equation (5.7) yields the following equations from (5.10) to (5.15) for any $q > 0$.

$$Q_q R_q Q_q^T = Q_q O_q X O_q^T Q_q^T + Q_q H_q W H_q^T Q_q^T + Q_q \Delta_q Q_q^T \quad (5.10)$$

where Δ_q is the error term caused by finite period number m of the PRBS and based upon Theorem 3.1 in reference [79], the Frobenius norm of Δ_q goes to zero as m goes to ∞ . The rest of matrices are defined below.

$$R_q = \begin{bmatrix} R_{yy}(0) & R_{yy}(1) & \cdots & R_{yy}(q-1) \\ R_{yy}(1) & R_{yy}(0) & \cdots & R_{yy}(q-2) \\ \vdots & \vdots & \ddots & \vdots \\ R_{yy}(q-1) & R_{yy}(q-2) & \cdots & R_{yy}(0) \end{bmatrix} \quad (5.11)$$

$$R_{yy}(\tau) = \sum_{j=0}^{n_w} \frac{1}{2m} \sum_{k=0}^{2n_w m-1} y_s^j(k+\tau) [y_s^j(k)]^T \quad (5.12)$$

$$X = \sum_{i=0}^{n_w} \frac{1}{2m} \sum_{n=0}^{2n_w m-1} x_s^j(n+\tau) [x_s^j(n)]^T \quad (5.13)$$

$$W_q = \begin{bmatrix} W(0) & W(1) & \cdots & W(q-1) \\ W(1) & W(0) & \cdots & W(q-2) \\ \vdots & \vdots & \ddots & \vdots \\ W(q-1) & W(q-2) & \cdots & W(0) \end{bmatrix} \quad (5.14)$$

$$W(\tau) = \begin{cases} \left(n_r - \tau \frac{m-1}{m}\right) \text{diag} [a_1^2, \dots, a_{n_w}^2], & \tau < n_r \\ 0_{n_r \times n_r} & \tau \geq n_r \end{cases} \quad (5.15)$$

where n_r is the sample period ratio of PRBS and system. As a result, for sufficient large m ,

$$Q_q R_q Q_q^T = Q_q O_q X O_q^T Q_q^T + Q_q H_q W_q H_q^T Q_q^T \quad (5.16)$$

which allow the use of PRBS to develop a q-Markov COVER for system identification by obtaining these system matrices in (5.1) following these brief steps of identification algorithm described below. More details can be found in reference [48].

- 1) Determine the PRBS input order m and magnitude a based on knowledge of system dynamics and conduct experiments to obtain the corresponding steady-state input-output responses.

- 2) Calculate auto-correlation and cross-correlation matrices $R_{yy}(i)$ and $R_{yw}(i)$ using following equations

$$R_{yy}(i) = \frac{1}{2m} \sum_{j=0}^{n_w} \sum_{k=0}^{2m-1} y_s^j(k+i) [y_s^j(k)]^T \quad (5.17)$$

$$R_{yw}(i) = \frac{1}{2m} \sum_{j=0}^{n_w} \sum_{k=0}^{2m-1} y_s^j(k+i) w_i(k) [e_j]^T \quad (5.18)$$

where $w_i(k)$ is the PRBS input and $y_s^j(k)$ is the steady-state system responses with only the j^{th} input excited by $w_j(k)$. The system Markov matrices can be then computed by

$$H_i = R_{yw}(i) W^{-1}; W = I a^2 \quad (5.19)$$

- 3) Select q and form matrices R_q , M_q , and H_q , where M_q can be calculated as below.

$$M_q^T = [H_1^T, H_2^T, \dots, H_{q-1}^T] \quad (5.20)$$

4. Form matrix P_q by

- Computing $\mathcal{D}_q = \mathcal{R}_q - \mathcal{H}_q W_q \mathcal{H}_q^T$, where

$$W_q = \text{block-diag}[W, \dots, W]; W = I a^2 \quad (5.21)$$

- and doing the Schur decomposition of D_q to obtain

$$\mathcal{D}_q = [\hat{\mathcal{P}}_q \quad \tilde{\mathcal{P}}_q] \begin{bmatrix} \Lambda & 0 \\ 0 & \tilde{\Lambda} \end{bmatrix} \begin{bmatrix} \hat{\mathcal{P}}^T \\ \tilde{\mathcal{P}}^T \end{bmatrix} \quad (5.22)$$

such that the diagonal elements of block diag $[\Lambda, \tilde{\Lambda}]$ are in a decreasing order, and Λ is

chosen such that $\bar{\sigma} \|\hat{\Lambda}\| < \underline{\sigma} \|\Lambda\|$, where $\Lambda \in R^{r \times r}$

- letting $\mathcal{P}_q = \hat{\mathcal{P}}_q \Lambda^{1/2}$

- 5) Construct P , \bar{P} , and $V_P U V_P^T$ by

- defining

$$\begin{aligned} P_q^T &= [P_0^T, P_1^T, \dots, P_{q-1}^T], \\ P^T &= [P_0^T, P_1^T, \dots, P_{q-1}^T], \\ \bar{P}^T &= [P_0^T, P_1^T, \dots, P_{q-1}^T] \end{aligned} \quad (5.23)$$

- and making the singular value decomposition of P and $[\bar{P}, M_q]$

$$\begin{aligned} P &= \begin{bmatrix} U_a & U_b \end{bmatrix} \begin{bmatrix} \Sigma_a & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_a^T \\ V_b^T \end{bmatrix} \\ [\bar{P}, M_q] &= \begin{bmatrix} U_c & U_d \end{bmatrix} \begin{bmatrix} \Sigma_c & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_c^T \\ V_d^T \end{bmatrix} \end{aligned} \quad (5.24)$$

- If P is full rank, letting $V_p UV_{\bar{P}}^T = 0$, and else,

$$V_p UV_{\bar{P}}^T = V_b UV_d^T \quad (5.25)$$

where U is an arbitrary unitary matrix

- 6) Compute $\hat{A}, \hat{D}, \hat{C}, \hat{H}$ for identified system by letting

$$\begin{aligned} \hat{A} &= P^+ \bar{P} + V_p UV_{\bar{P}}^T I_1, \quad I_1 = [I_r, 0]; \\ \hat{D} &= P^+ M_q + V_p UV_{\bar{P}}^T I_2, \quad I_2 = [0, I_{n_w}]; \\ \hat{C} &= Q^{-1} P_0, \hat{H} = Q^{-1} H_0 \end{aligned} \quad (5.26)$$

Note that reference [79] shows that if the system responses are from a nonlinear system, the identified system model represents a linearized model of that nonlinear system.

5.1.2 LQG Control

The identified linear system model based on PRBS open-loop system responses can be used for model-based control such as LQR (linear quadratic regulator). However, state feedback LQR cannot be implemented since not all system states of identified model can be measured directly. The linear–quadratic–Gaussian (LQG) control, one of the most popular optimal control strategies, is used in this study with unknown states. The LQG control can be used for a class of linear systems with additive white Gaussian measurement noise. The LQG regulator uses the noisy measurements to estimate the system states and then generate a state feedback control signal based on estimated states to regulate the system output to zero. The resulting LQG controller is a combination of a Kalman filter (linear quadratic state estimator) and an LQR [80, 81]. Note that both Kalman filter and LQR can be designed based on the identified model. The theory of LQG control is introduced below in brief and more details can be found in literature [80, 81].

When all system states are known, a linear quadratic regulator law is shown below in Equation (5.27) that minimizes the integrated quadratic cost function in Equation (5.28).

$$u = -Kx \quad (5.27)$$

$$J_{LQR} = \frac{1}{2} \int_0^\infty (xQx^T + u^T Ru) dt \quad (5.28)$$

For LQG control, it is assumed that the physical system is excited by system disturbance (process noise) w (see (5.1)) with the following system model

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + Dw(k) \\ y(k) &= C_y x(k) + H_y w(k) \\ z(k) &= C_z x(k) + v(k) \end{aligned} \quad (5.29)$$

where $B = D$; C_y and H_y are part of the identified C and H , respectively, associated with KI output; C_z is part of C associated with δ exhaust temperature; and $v(k)$ is the measurement noise. Since states of identified system cannot be measured directly, the LQR relies on the estimated state vector $\hat{x}(k)$ based on noisy measurements z using a Kalman filter defined in Equation (5.30), which forms the LQG regulator. The Kalman filter is an optimal estimator when dealing with Gaussian white measurement noise that minimizes the covariance $\lim_{t \rightarrow \infty} E((x - \hat{x})(x - \hat{x})^T)$ of estimation error $x - \hat{x}$.

$$u = -K\hat{x} \quad (5.30)$$

where \hat{x} is the estimated state by Kalman filter. Its state-space representation is

$$\dot{\hat{x}} = [A - LM - (B - LD)K]\hat{x} + Lz \quad (5.31)$$

where z is the vector of system measurements and L is the Kalman gain. Control gain K is obtained by using the unique stabilizing solution of the corresponding Riccati equation [82]. The structure of this LQG regulator in our application is shown below in Figure 5.1.

The physical meanings of these two algorithms with application to knock cycle-to-cycle compensation using measured delta exhaust temperature as feedback will be detailed in the next section.

5.2 Exhaust Temperature Study

The same engine bench architecture, as shown in Figure 3.8, is utilized. Note that the engine was running at a fixed speed and load condition (1200 rpm and 90 Nm) for all the tests in this chapter.

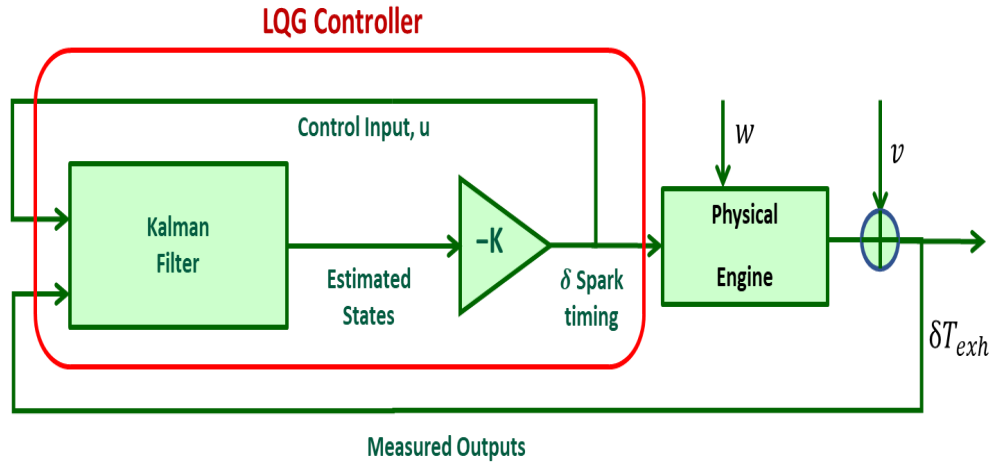


Figure 5.1 LQG control architecture

A bare-wire thermal couple with a 0.003s response time is installed close to the exhaust valve of the 3rd cylinder; see Figure 5.2.

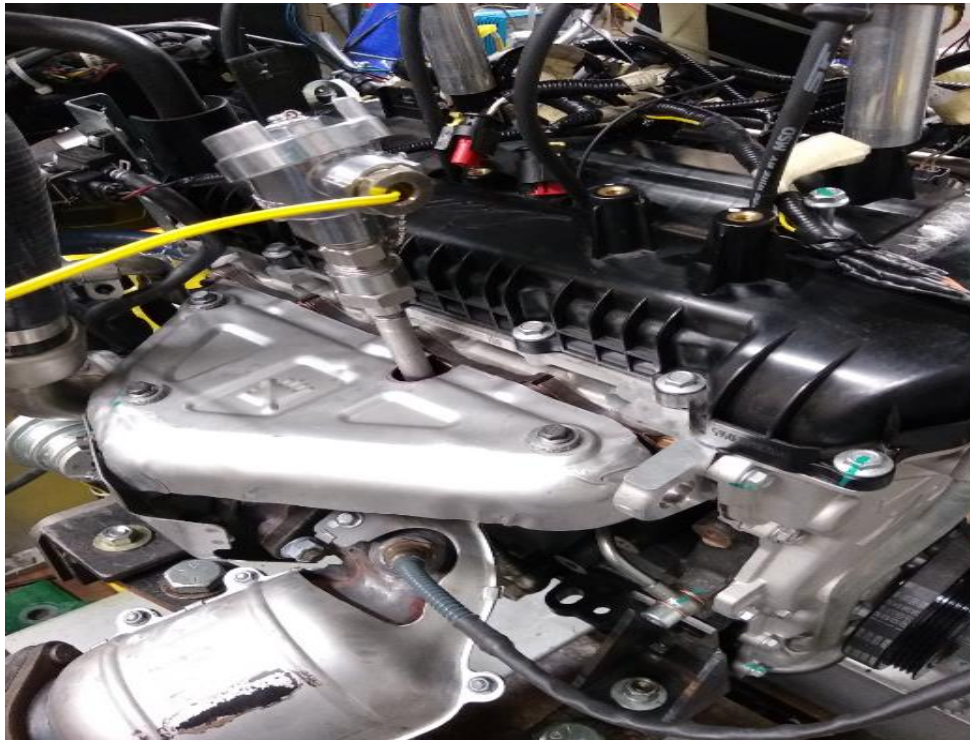


Figure 5.2 Thermal couple installation location

In order to better understand the engine combustion dynamics under small spark timing perturbation, some preliminary tests were conducted for the purpose of analyzing exhaust temperature signal when the input spark timing changes. The structure of open-loop test is shown below in

Figure 5.3 when δ spark timing is added to the baseline one. Note that this configuration will also be used for system identification based on PRBS q -Markov COVER.

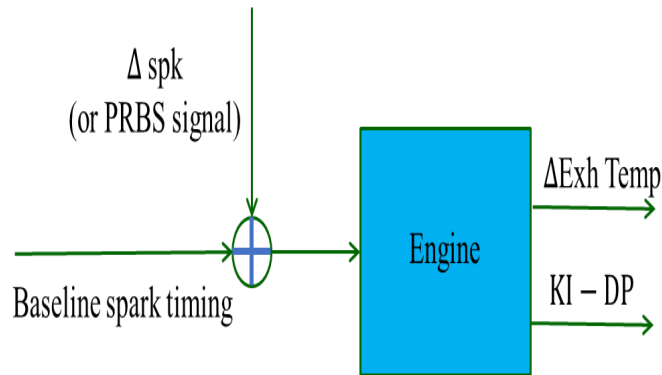


Figure 5.3 Architecture of open-loop test system

Since the thermal couple is installed close to the 3rd cylinder, only spark timing of the 3rd cylinder will be adjusted to affect the combustion so that the corresponding exhaust temperature changes could be detected. The step spark timing test results are shown in Figures 5.4 and 5.5 while the δ spark timing of 1 and 2 deg are commanded as a step input, respectively.

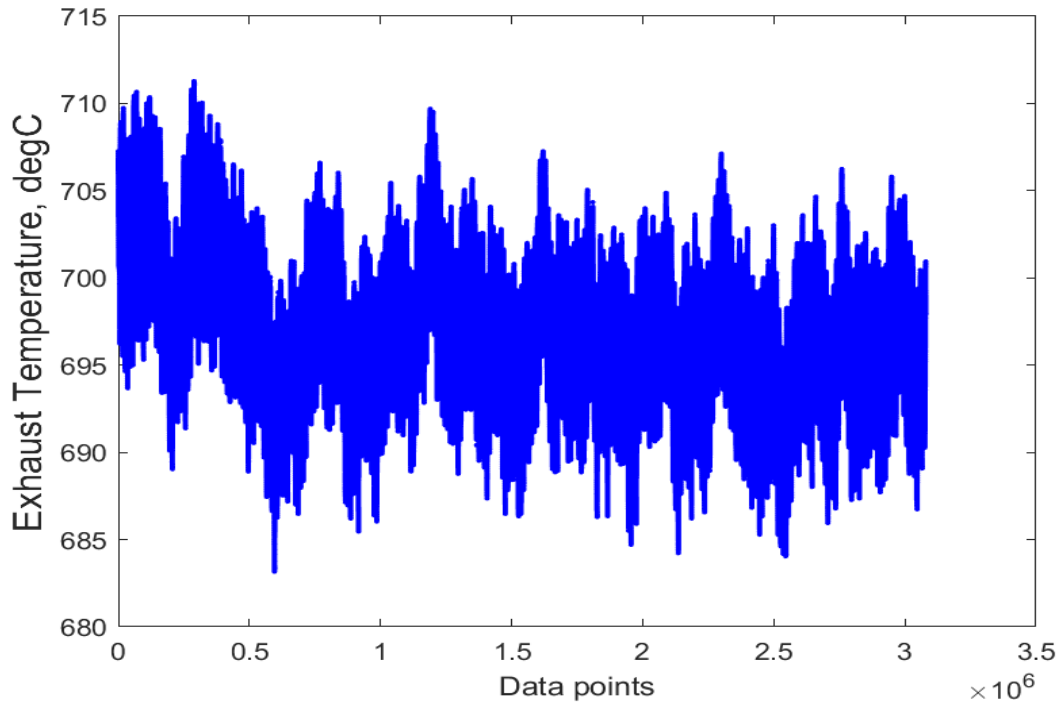


Figure 5.4 Exhaust temperature with unit δ spark timing step

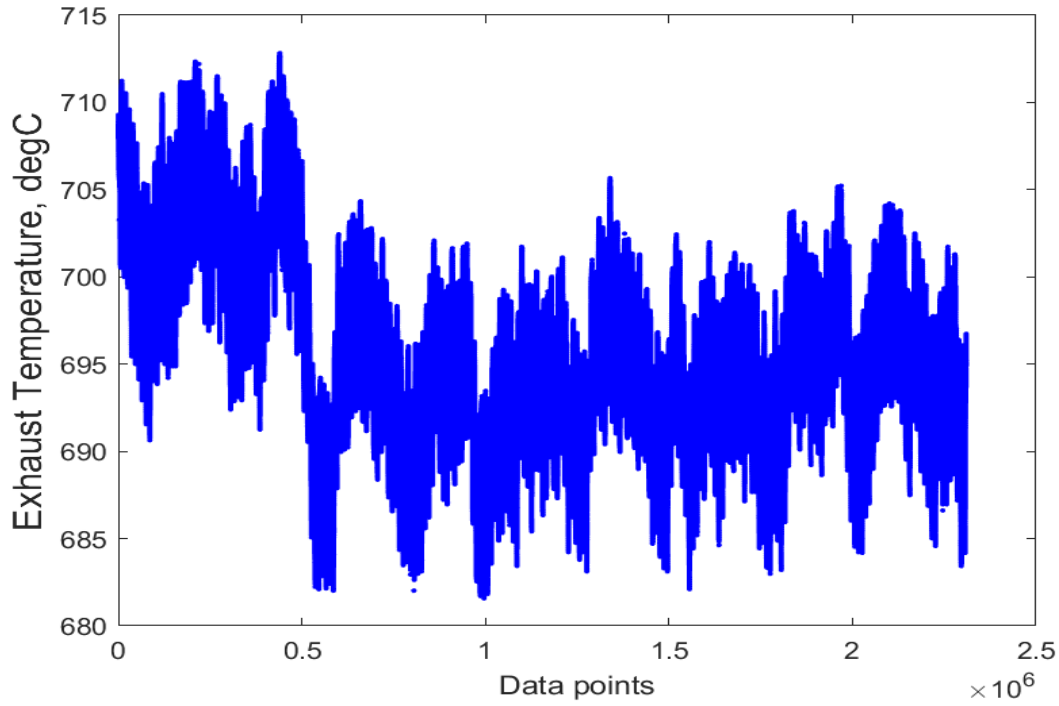


Figure 5.5 Exhaust temperature with two-degree δ spark timing step

The test data shows the obvious effect of step δ spark timing. A larger step in the advanced direction causes a larger exhaust temperature drop. This observation also confirms that the engine is excited successfully by the step input and the exhaust temperature response can be captured by the installed bare-wire thermal couple. The unit step input test data will be later used to validate the identified model by q -Markov COVER. Note that the absolute temperature is not a good indication of δ spark timing change since the absolute value would not be consistent even under the same operational condition (speed and load) due to the test environmental change and sensor noise. The δ temperature, defined as measured exhaust temperature minus the cycle-wised mean one, will be used for the system identification and its fluctuations represents the cycle-to-cycle combustion variations. The δ temperature of each cycle is plotted in Figures 5.6 and 5.7, corresponding to the previous test results of step δ spark timing inputs shown in Figures 5.4 and 5.5.

For offline post-data processing, the δ temperature can be easily obtained after some signal processing. However, this would not be possible in real-time, because the dynamic temperature data would not be maintained at a fixed mean value under different test conditions. Therefore,

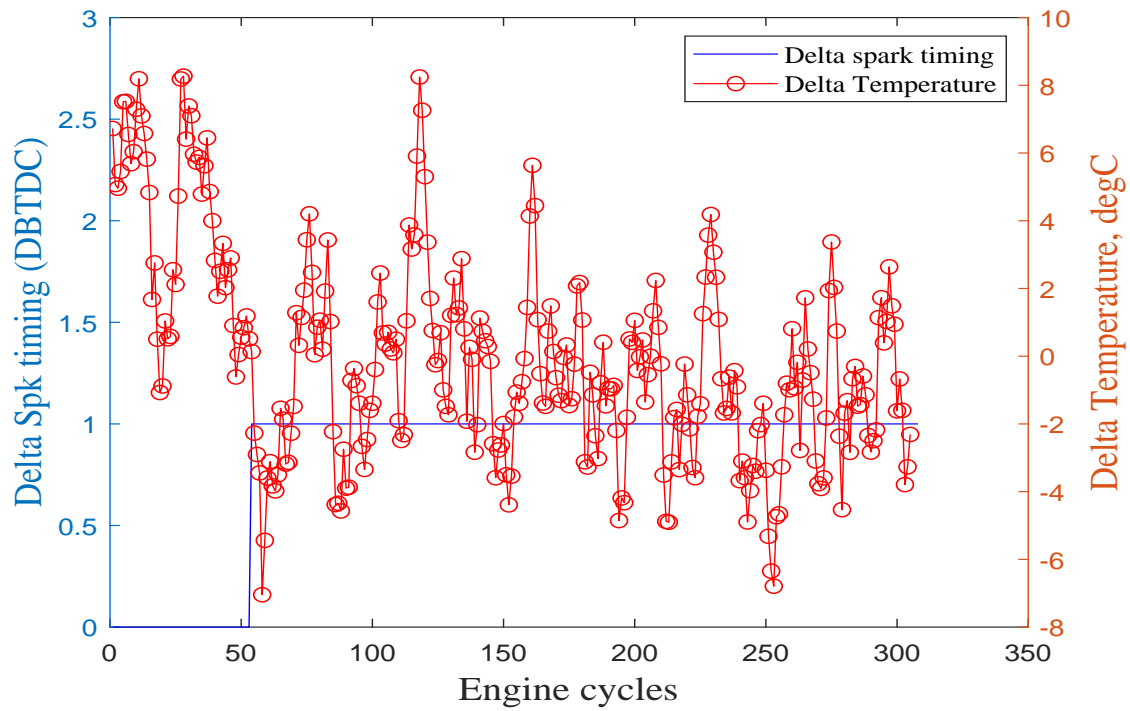


Figure 5.6 Δ exhaust temperature response with unit δ spark timing step

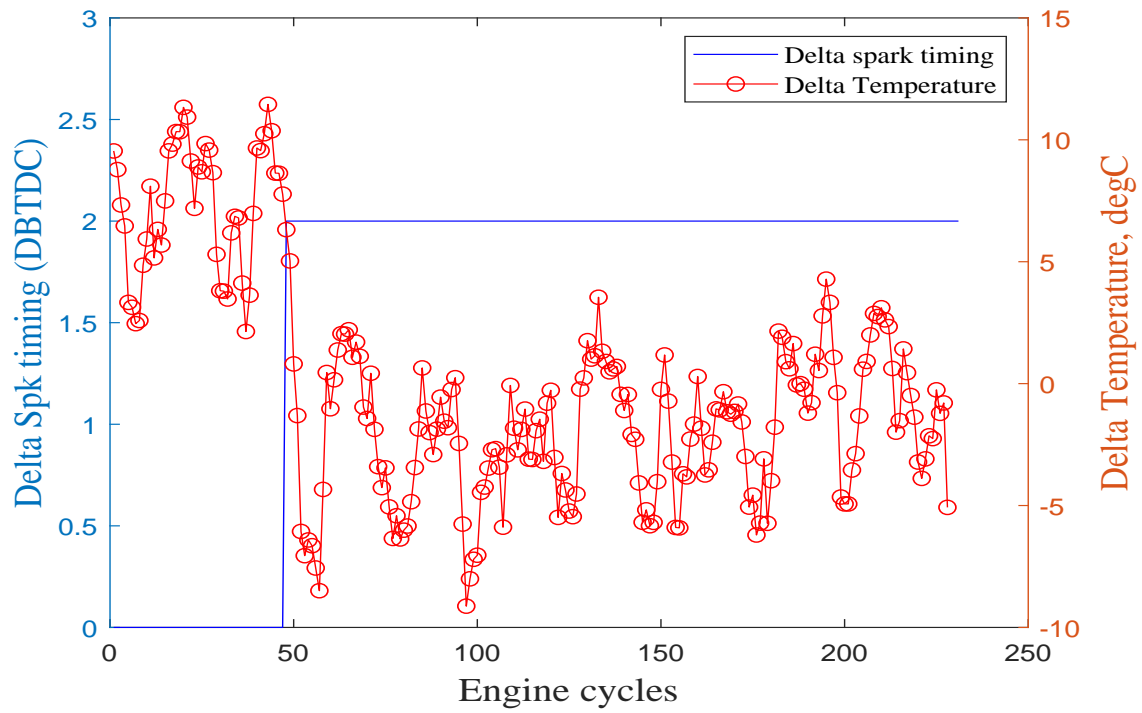


Figure 5.7 Δ exhaust temperature response with two-degree δ spark timing step

a high-pass (HP) filter is proposed to get rid of the mean component in the exhaust temperature signal. These two methods are compared in Figure 5.8 and both processed signals are fairly close.

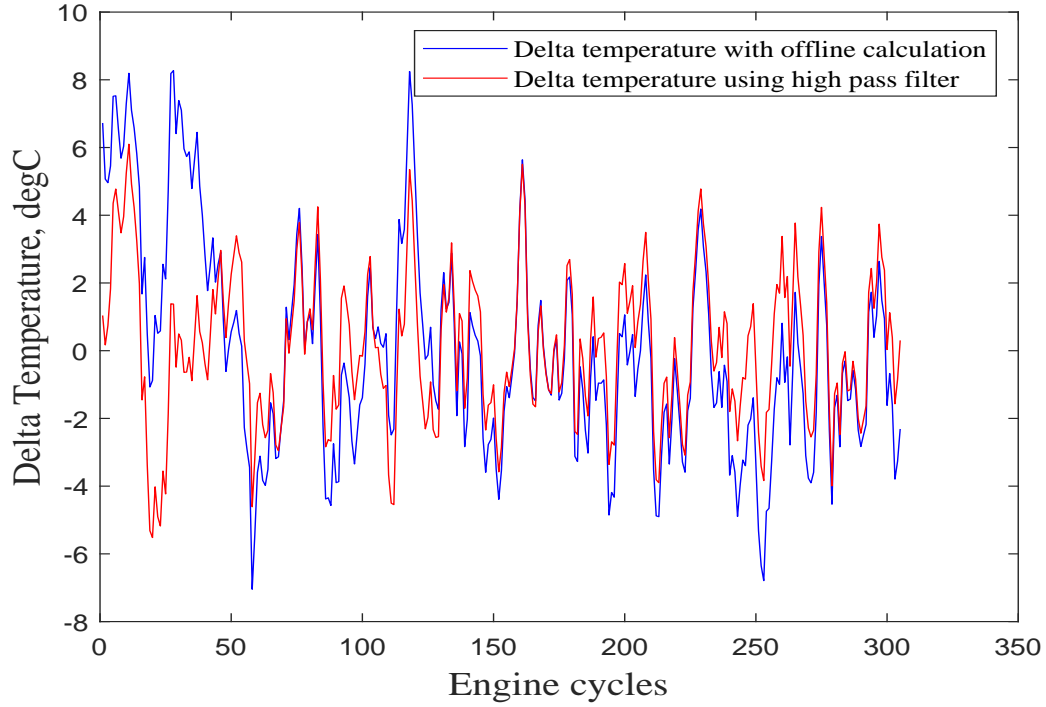


Figure 5.8 Δ exhaust temperature obtained by two different approaches

Figure 5.8 shows that the selected HP filter is able to obtain the accurate δ exhaust temperature. For the engine test, the sampled exhaust temperature signal from the bare-wire thermal couple is buffered in NI data acquisition system over a given window for cylinder #3 every cycle; and the cycle mean of temperature can be calculated and used as the high-pass filter input. The output of high pass filter is the desired δ temperature to be used for system identification and model-based control.

5.3 System Identification Open Loop Tests and Model Verification

5.3.1 Configuration for System Identification

As proposed, the model-based control will be developed for cycle-to-cycle compensation. An open-loop test is designed and implemented for system identification. The test structure is the same as that shown in Figure 5.3. The baseline spark timing is fixed and δ spark timing from PRBS is added to the baseline and all three signals (δ spark timing, δ exhaust temperature and KI) are collected for q -Markov COVER system identification. In this paper, a 7th order PRBS with 1

degree magnitude is sent to the engine controller as the δ spark timing. One period of the 7th order PRBS is shown below in Figure 5.9, where the pattern of PRBS sequence among each cycle can be noticed in zoom-in box.

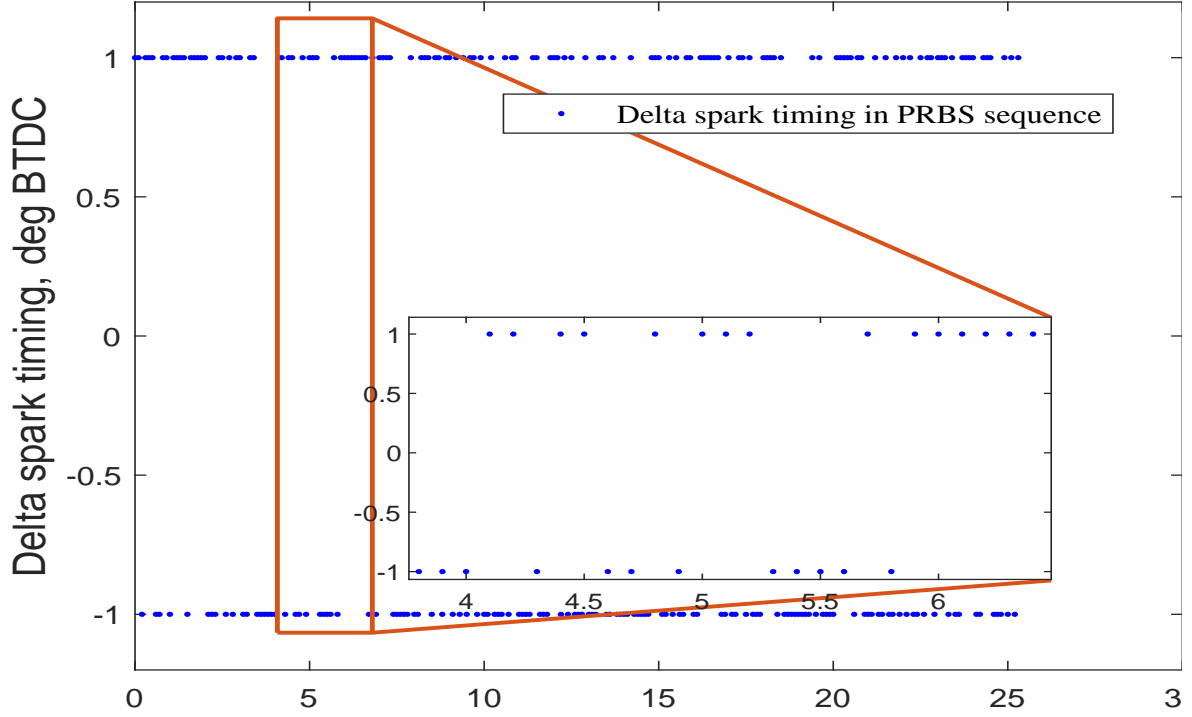


Figure 5.9 The 7th order PRBS (one period)

The length of PRBS signal is $l = 2 * (2^7 - 1) = 254$ engine cycles. When engine is running at 1200 rpm, it takes 25.4 seconds for completing one PRBS cycle. Since steady-state exhaust temperature and KI signals are needed for q -Markov COVER system identification (to make sure Δ_q is close to zero) and exhaust temperature has a slow response to the PRBS signal, engine tests of five continuous PRBS cycles are conducted to make sure that the engine is at the steady-state over the fifth PRBS cycle. Only the signals of the last (fifth) PRBS cycle are used for system identification.

5.3.2 Model Verification

Two modifications are made so that the identification algorithm can be used for this specific application (knock control). Firstly, the q -Markov COVER identification algorithm assumes Gaussian noise and excitation. However, Our early study on knock stochastic properties [8] shows that the

knock intensity probability density function (PDF) is not a symmetric Gaussian random process but has a log-nominal distribution, which was also verified by other researchers [61]. Thus a distribution conversion [50] is necessary to make it possible to implement the system identification algorithm. On the other hand, the fluctuation and absolute amplitude of δ temperature has a much larger range than that of the knock intensity; see Figures 5.6 and 5.8. If these two signals are not scaled properly, large system identification error of the signal with small magnitude will occur. Therefore a weighting matrix $diag[0.7, 0.1]$ is added to the outputs signals (KI and delta temperature) after the iterative tuning processes.

For the identification process, q needs to be selected for the best identification results. Note that the physical meaning of q is that the first q Markov parameters of identified model match with these of physical system. Therefore, the larger q is, the better the approximation of identified model to the real engine system. However, large q could increase computational cost and cause model over fitting. By trial and error, $q = 20$ was selected to achieve minimal identification error. Similarly, higher order identified model can reduce modeling error, but it may not realistic for further real-time control implementation since it will lead to a high order LQG controller. A 4th order model is adopted to balance the model accuracy and proper real-time controller order. For the 4th order model, the identified error was increased, and the identified KI magnitude is far below the actual one. However the overall oscillation trends of δ temperature and KI still present a good tendency and shall be useful for future cycle-to-cycle compensation. The identified results are shown below in Figures 5.10 and 5.11 for KI and δ exhaust temperature, respectively.

The accuracy of identified model needs to be verified. The step response of the identified model is compared with the test data with the same unit step input, where the step responses of the identified model are shown in Figure 5.12. The results show that positive δ KI and negative δ temperature with a positive step input. An obvious anti-phase phenomenon can be observed between the δ KI and temperature, which further verified the early observation of the relationship between the δ KI and exhaust temperature.

As highlighted earlier, the ID model cannot track the exact amplitude of δ exhaust temperature

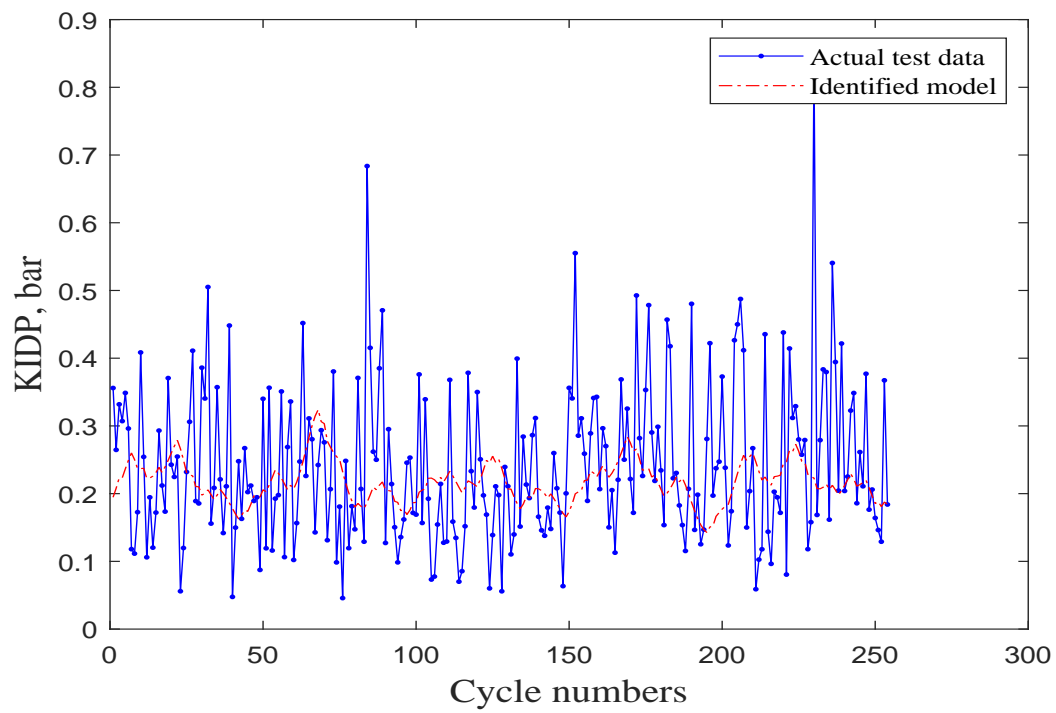


Figure 5.10 KI-DP of ID model

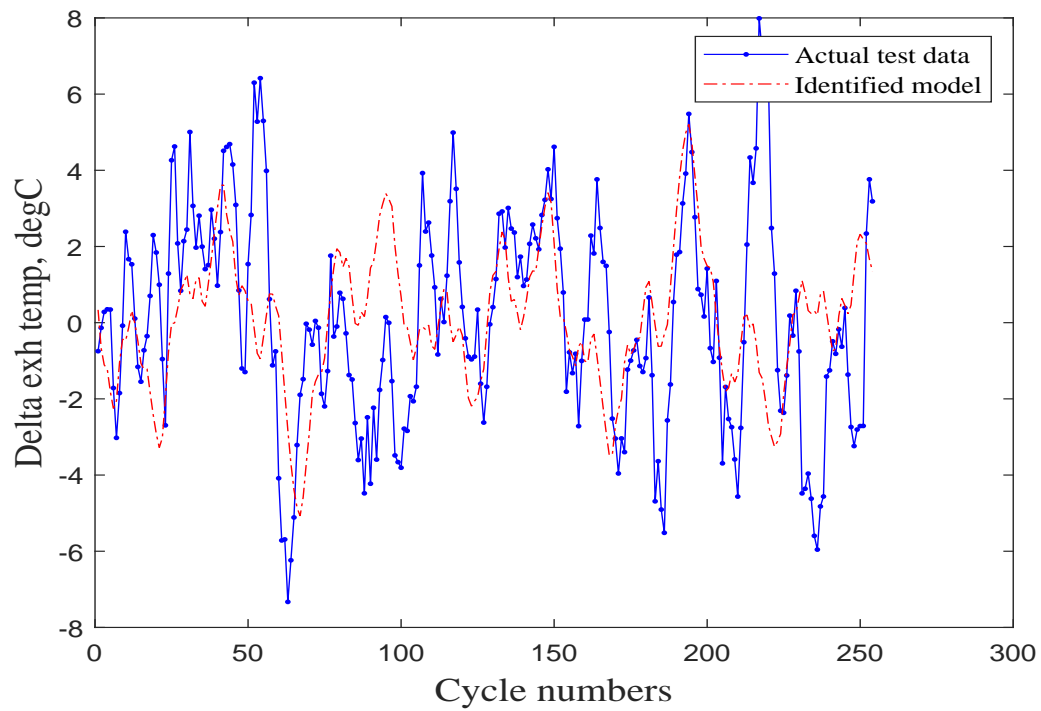


Figure 5.11 Δ exhaust temperature of ID model

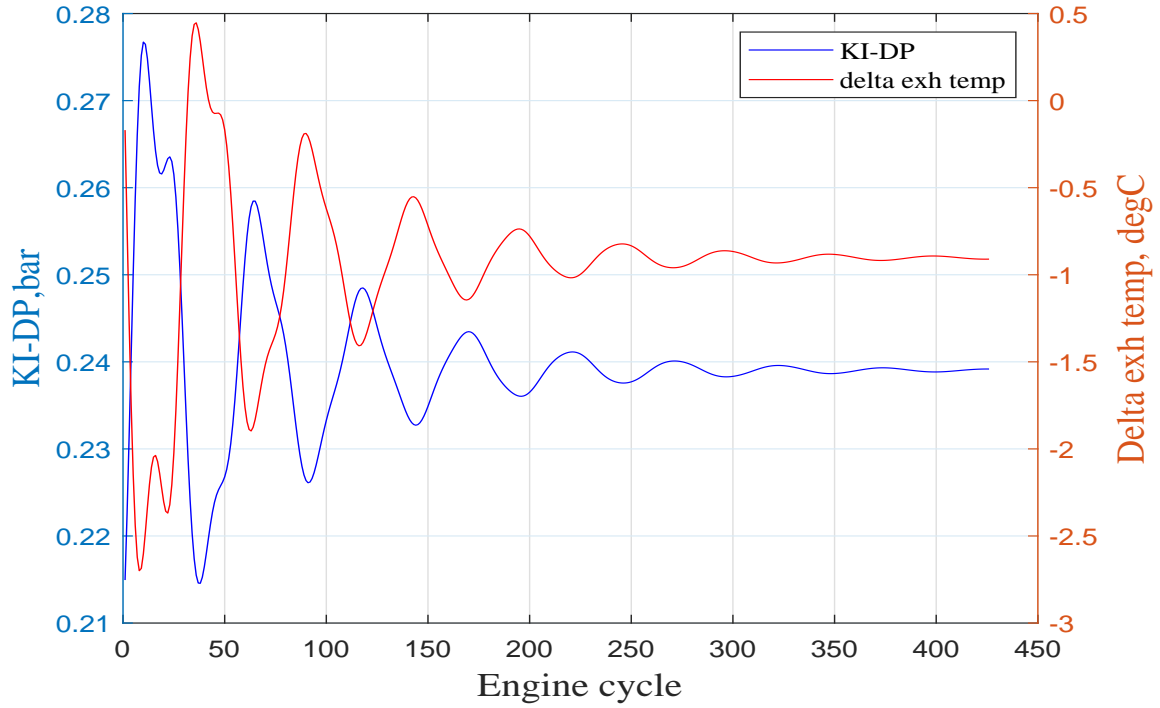


Figure 5.12 Step responses of ID model

due to identification error, but its general tendency aligns with the test data and same for the fluctuation frequency. For the purpose of comparison with the ID model, a band-stop filter is applied to the test data to filter out the large peaks caused by exhaust pipe pressure waves without sacrificing the overall trend. The comparison results are shown in Figure 5.13. The compared results show that the response frequency of test data is close to that of identified 4th order linear model with $q = 20$. The 4th order system model, obtained from the proposed q -Markov COVER, is able to recreate the fluctuation behavior of the nonlinear engine combustion dynamics. Its state-space model matrices is shown below in Equations (5.32) to (5.35). Thus LQG controller can be designed accordingly. The output of LQG controller is δ spark timing which is sent to the MotoTron engine controller. For the actual engine operation, LQG regulator tries to reduce the δ exhaust temperature down to zero. If that is positive, which means the exhaust temperature is increasing due to a late or slow combustion with a relative retard spark position, therefore, the spark timing is expected to be moved in an advanced direction. Conversely, if the δ temperature is negative, which means the engine is running at a relative advanced position than the baseline, a

retard compensation is required.

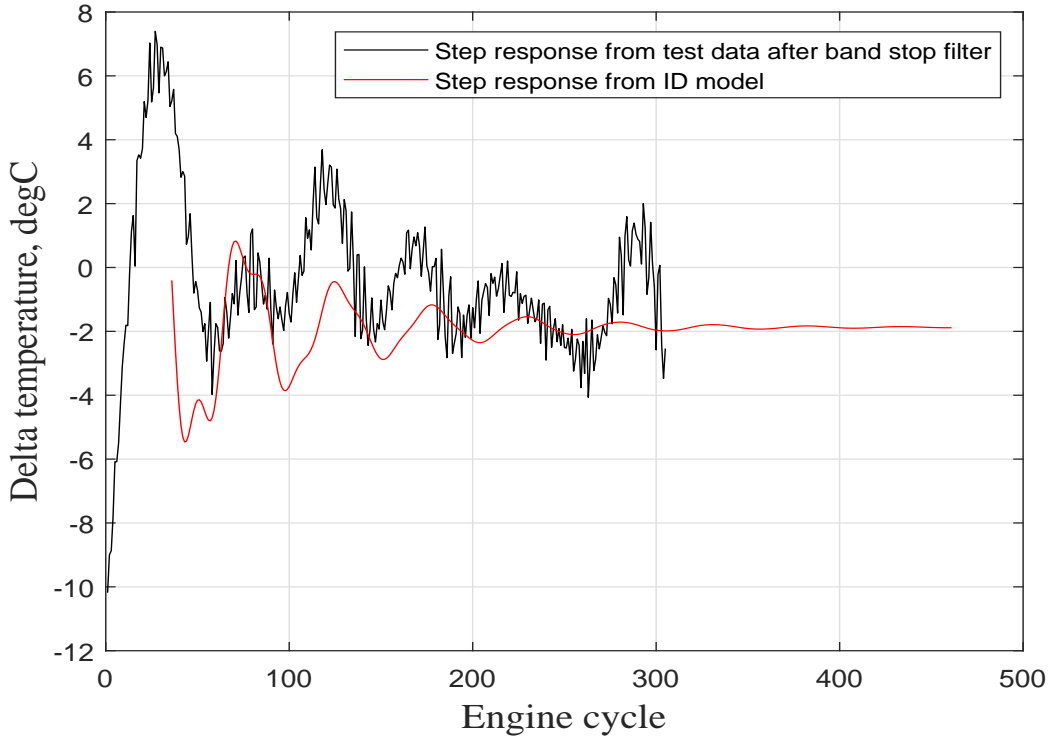


Figure 5.13 Comparison of step responses

$$A = \begin{bmatrix} 0.9880 & -0.1279 & -0.0042 & -0.0444 \\ 0.1032 & 0.9557 & -0.2010 & -0.0322 \\ -0.0065 & 0.1321 & 0.9248 & -0.2939 \\ 0.0443 & 0.0601 & 0.2709 & 0.9283 \end{bmatrix} \quad (5.32)$$

$$B = \begin{bmatrix} -0.0747 & -0.1819 & -0.1754 & -0.0244 \end{bmatrix}^T \quad (5.33)$$

$$C = \begin{bmatrix} -0.0311 & -0.0660 & -0.0457 & -0.0926 \\ 0.0753 & 0.1549 & 0.11460 & 0.0848 \end{bmatrix} \quad (5.34)$$

$$D = \begin{bmatrix} 5.2708e - 05 \\ -0.0235 \end{bmatrix} \quad (5.35)$$

5.4 Test Results with Cycle-based Knock Control

In order to verify the effect of proposed cycle-to-cycle LQG compensation strategy based on the identified linear model, three different tests were designed as listed in Table 5.1.

Table 5.1 Test scenarios for LQG cycle-to-cycle compensation at fixed speed and load condition

Case Number	Cycle-based algorithm	Online updating strategy	Coolant Temperature deg C
1	Yes	No	90
2	Yes	Yes	90
3	Yes	Yes	95

As shown in Table 5.1, all three test cases use the proposed cycle-based LQG algorithm. Note that Case 1 demonstrates the LQG control along with a fix baseline spark timing. Note that in our early research, an online likelihood strategy was proposed to online-update the baseline control parameter (spark timing) according to the engine KI stochastic variations; see reference [83] for details. For Cases 2 and 3, the LQG compensation strategy is integrated with the online updating strategy using the architecture in Figure 1.2 to form the complete knock control architecture. After combining the online updating strategy with the cycle-to-cycle LQG compensation, tests of Cases 2 and 3 were run under two different coolant temperature levels (90 and 95 deg C), respectively. These test results are discussed in the following sections, respectively.

5.4.1 Test Results with ID Model Only - Case 1

Firstly, test results with LQG control only is shown in Figure 5.14. At the beginning phase shown in the purple box, it can be noticed that the black line (δ spark timing) is a straight line, which means that cycle-based LQG control is not activated and the engine runs close to the knock borderline with a fixed baseline spark timing. After that period, the LQG control is enabled (see the blue box) and the spark timing starts to fluctuate within the range of $[-0.6, 0.6]$ degree. It can be seen that KI variance is reduced from 0.014 to 0.010 (a 28.6% reduction) after the LQG control is activated, which indicated that the overall combustion stability is improved by 28.6%.

This LQG control has suppressed the KI variation, therefore it has a potential to be integrated with previously developed online updating process so that the mean spark timing can be further advanced for improved engine performance.

5.4.2 Test Results After Integration with Online Updating Strategy (Cases 2 and 3)

After integrating the LQG spark timing control strategy with online updating scheme, the final engine knock control structure as shown in Figure 1.2 is configured. The overall spark timing

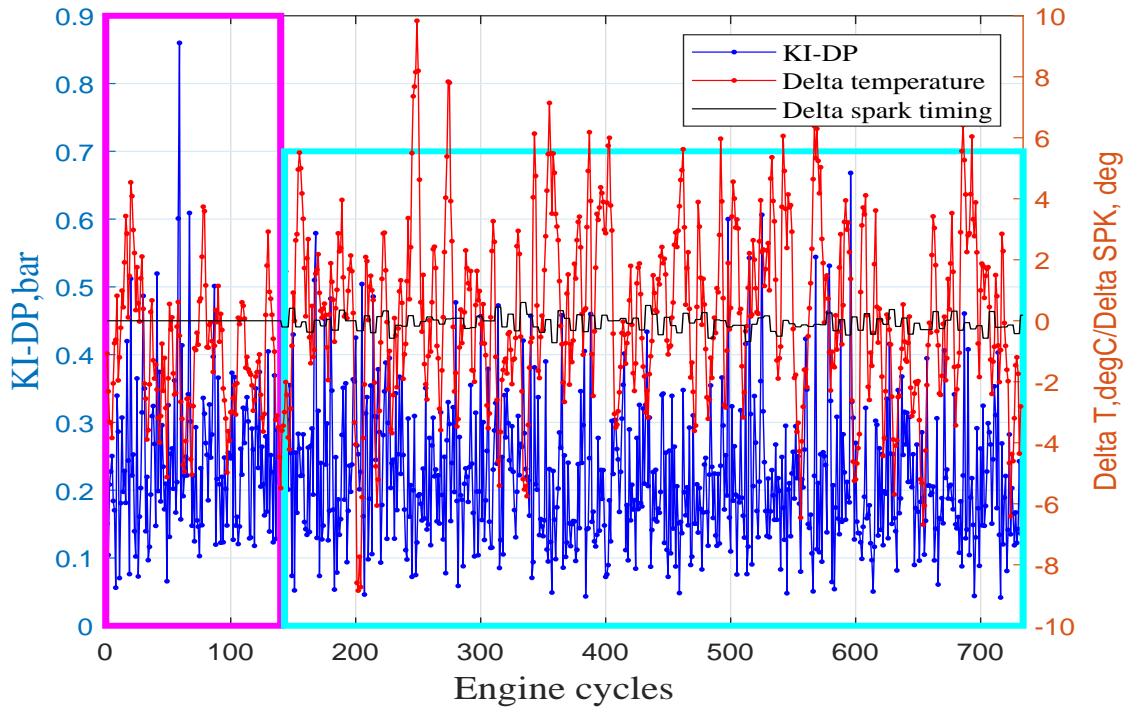


Figure 5.14 Test results with LQG control only(Case 1)

control is expected to be further advanced when coolant temperature is at normal condition (same temperature as that in Case 1). Two different test scenarios are defined to verify the entire knock borderline control structure: a) using the regular fuel with normal coolant temperature (90 degree C) and b) using the same fuel with elevated coolant temperature (95 degree C).

As shown in Figure 5.15, the changing tendencies of spark timing and exhaust temperature are obvious. When spark timing (baseline plus δ spark timing) is advanced, the exhaust temperature decreases. The spark timing is zoomed in Figure 5.16, where the δ spark timing compensation (from LQG control) is activated all the time and baseline spark timing is compensated accordingly. The relative larger modulation of the baseline spark timing is caused by the online updating control scheme developed earlier, which can be referred to the changes of α (a model parameter updated online). Its value is calculated based on the online updating strategy. More details of model updating strategy can be found in Chapter 4 or reference [83]. The minor (δ) fluctuations are added to the baseline spark timing. It can be noticed that the δ spark timing varies within the range of ± 1 degree. The effect of these varying spark timing tuning can be viewed in Figure 5.17. The

large change is triggered by the fact that the earlier segment of KI stochastic properties are lower than the offline calibrated results. LQG compensation assists to reduce the cycle-wise variations so the spark timing can be further optimized (advanced) with the overall KI staying within the defined probability (1/300).

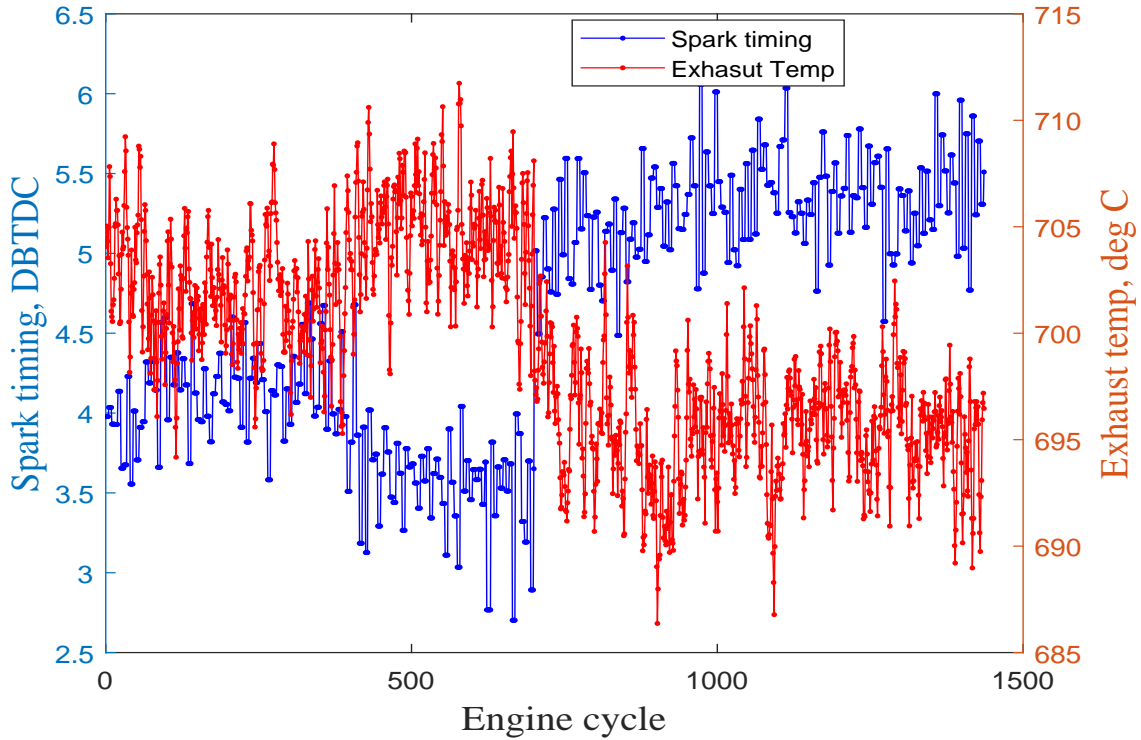


Figure 5.15 Spark timing and exhaust temperature with normal coolant temperature (Case 2)

When the coolant temperature is adjusted to 95 degree C in Case 3, the spark timing converges to a relative retarded position due to the likelihood algorithm: see Figure 5.18. Since the adjustment of spark timing is relatively small, the general change of exhaust temperature is also not as significant as before. From the perspective of cycle-wised spark timing compensation, minor spark timing changes are added (see Figure 5.19) based on the feedback of δ exhaust temperature. While the spark timing is stabilizing, the LQG controller continuously reduces overall KI variance. The time series of KI can also be viewed in Figure 5.20.

In summary, all three test cases show that the proposed control algorithm is able to provide the δ spark timing added to the baseline one for suppressing cycle-to-cycle knock variations. The LQG controller works well in real-time by utilizing the feedback of exhaust temperature data.

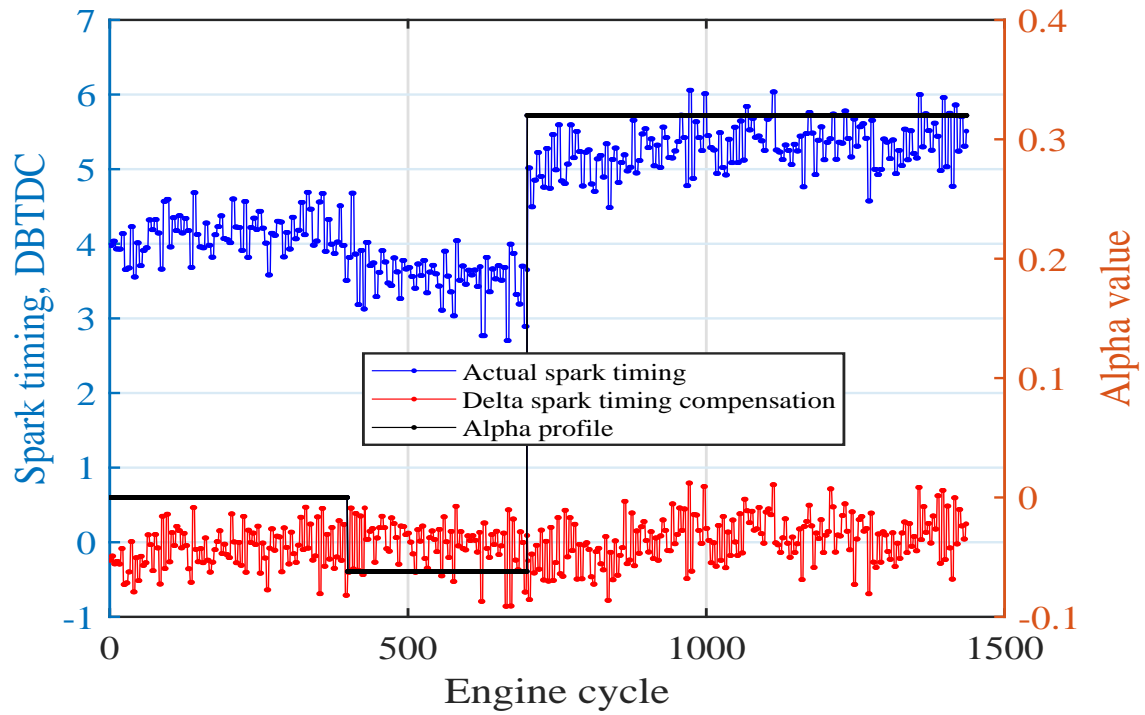


Figure 5.16 Spark timing modulations with normal coolant temperature (Case 2)

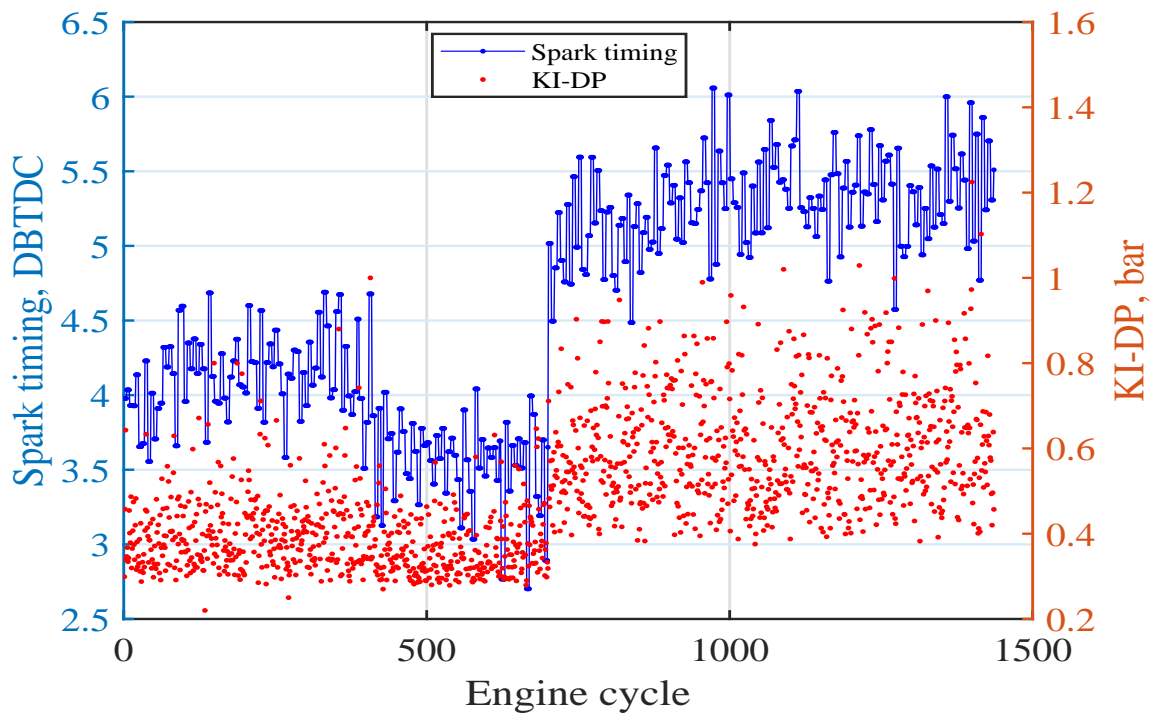


Figure 5.17 Spark timing and KI with normal coolant temperature (Case 2)

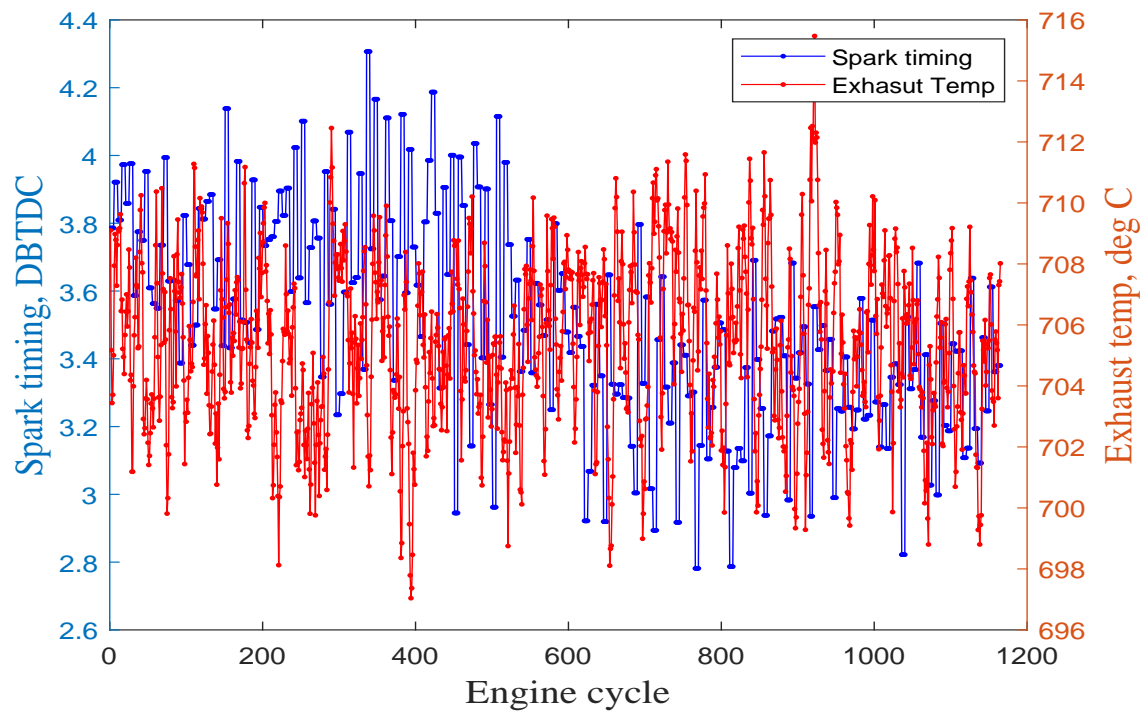


Figure 5.18 Spark timing and exhaust temperature with increased coolant temperature (Case 3)

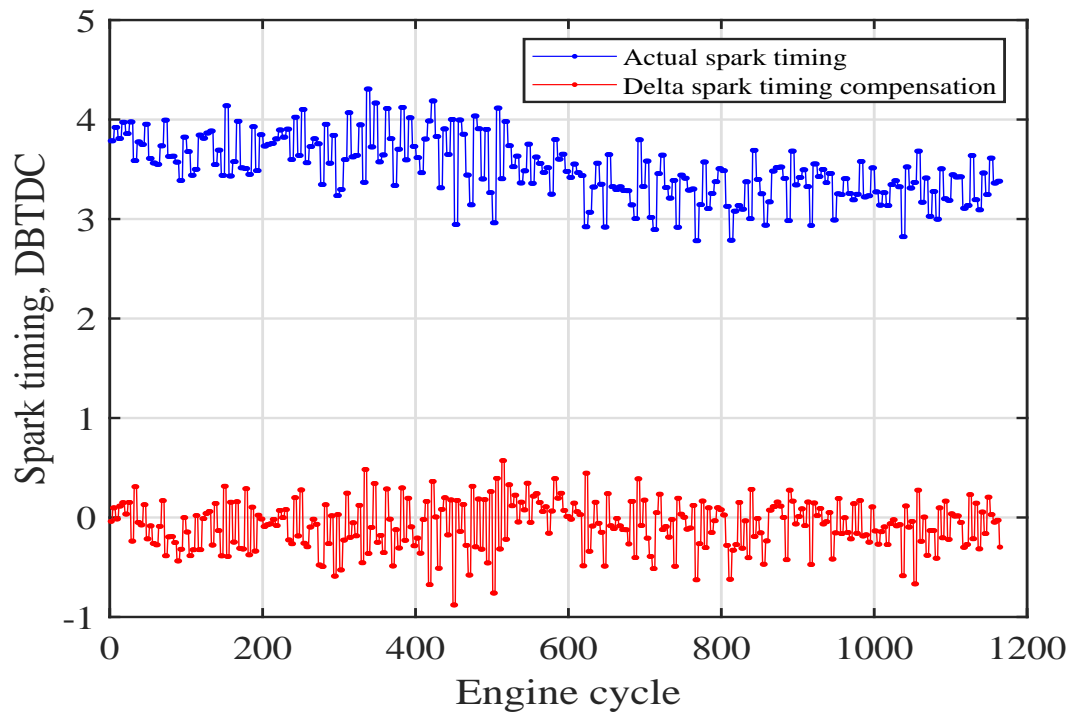


Figure 5.19 Spark timing modulations with increased coolant temperature (Case 3)

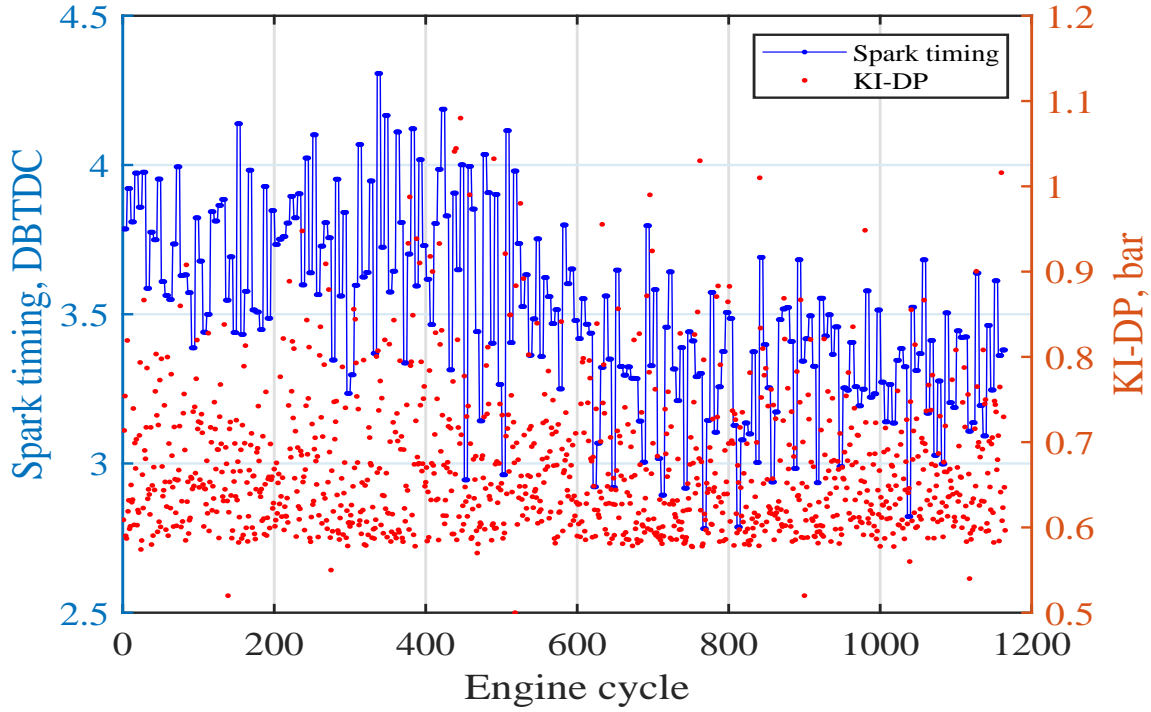


Figure 5.20 Spark timing and KI with increased coolant temperature (Case 3)

Additionally, it can be easily integrated with existing baseline algorithm without any issue under different scenarios. This indicates that it can be further extended to other applications in terms of the engine knock/combustion control.

5.5 Conclusions

This chapter proposes an cycle-based knock compensation architecture to suppress the cycle-to-cycle knock variation based on an LQG controller designed based on identified exhaust temperature deviation model using q -Markov COVER (COVariance Equivalent Realization). For the model identification, δ spark timing is selected as the model input, δ exhaust temperature and knock intensity (KI) as system outputs. An open-loop experiment was first conducted for collecting data to be used for system identification. Five continuous periods of the 7th order of PRBS (pseudo random binary signal) is used as δ spark timing excitation for the engine system, and only the steady-state data set (last period) is collected and used for system identification. An engine test bench is configured and assisted by the LabView-Matlab model communication interface for performing open-loop experiments and closed-loop engine control test later to verify the proposed algorithm.

During the system identification process, a log-normal to Gaussian distribution mapping and weighting matrix are used to modify the q -Markov COVER algorithm for the specific application. A 4th order linear model is identified and verified using model impulse response, which shows that their frequency of δ exhaust temperature signal is very close. This is important for the proposed LQG regulation since the model will be used for designing the model-based LQG control strategy. Three different scenarios are tested to verify the proposed LQG engine knock variation compensation strategy independently and with integration to early developed online updating scheme. The test results indicate that the LQG cycle-wised control with exhaust temperature feedback is able to reduce KI variation by 28.6%, which is able to further advance the spark timing for improved fuel economy over the traditional knock control logic. The proposed LQG control strategy is also easy to implement.

CHAPTER 6

TIRE-ROAD FRICTION COEFFICIENT ESTIMATION

In previous chapters, the data-driven based model embedded in a Bayesian algorithm is used to replace the traditional low efficient and high cost method for engine calibration and control, so that the entire process of knock borderline prediction and control becomes simple. Meanwhile, the cheap computational ability of Kriging model is utilized for the online updating process. As we discussed in Chapter 1, the data-driven based model can also be used to assist the estimation of an existing algorithm. This chapter covers this application scenario for estimating the tire-road friction coefficient.

In this chapter, significance of tire-road friction coefficient to the vehicle dynamics is highlighted. After that, a conventional vehicle and wheel dynamic models and sequential extended Kalman filter (S-EKF) are introduced in brief. After that, the S-EKF scheme using the CarsimTM simulated data is conducted and followed by a study for test data under different road conditions. A detailed derivation process and analysis of the proposed new evaluation criterion is presented, along with the signal fusion estimation scheme that is verified by using CarSimTM and test data set, respectively. Conclusions are drawn lastly.

6.1 Tire Road Friction Coefficient Introduction

Vehicle dynamics has attracted lots of research attention for a long period since the appearance of first car. Continuous progress has been made to improve the vehicle handling performance through vehicle control schemes, such as anti-block control [24], traction control [24, 25], stability control [26, 27], trajectory following [28], and so on. Nowadays, autonomous vehicles have even high requirements for vehicle safety, stability and control. Note that control efforts (forces) imposed on a vehicle play a crucial role in determining the vehicle dynamics [29]. The dominant forces applied to a vehicle are generated by tire-road interactions. It is intuitive that tire properties (such as tire material and texture, inflation pressure, and temperature) and road properties (dry, wet, snow, and ice) affect the force generation. For a given vehicle with all available information, an accurate estimation of road condition, that interacts with the vehicle, is the essential for vehicle

control to achieve optimized performance. At the same time vehicle skid can be avoided and its stability is enhanced. The characteristics of road surface can be represented by the tire-road friction coefficient, μ , which is the maximum of normalized traction force (see Equation (1.1)) and it varies under various road conditions such as dry, wet, snow, and ice. Profiles of friction coefficient μ on different road surfaces as a function of the tire slip ratio is plotted in Figure 6.1 as an example. Note that the longitudinal slip-ratio λ is normally defined by the following Equation (6.1).

$$\lambda = \begin{cases} 1 - \frac{R_e \cdot \omega}{V_x}, & \text{if } V_x \geq R_e \cdot \omega, \cdot V_x \neq 0 \\ 1 - \frac{V_x}{R_e \cdot \omega}, & \text{if } V_x < R_e \cdot \omega, \omega \neq 0 \end{cases} \quad (6.1)$$

where R_e is the effective tire radius; ω is the angular wheel speed; and V_x is the longitudinal vehicle speed.

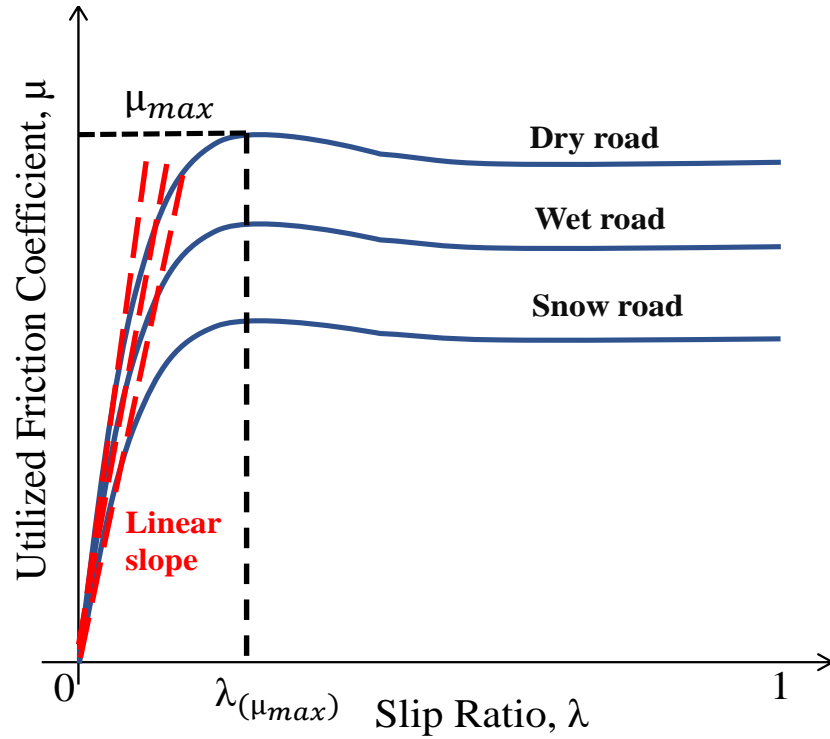


Figure 6.1 Friction coefficient curve for different surfaces

From the aforementioned Figure 6.1, the top curve corresponds to a road condition with dry rough surface. The middle and bottom curves in Figure 6.1 represent wet and snow road conditions. The corresponding slip-ratio (μ_{max}) is called the optimal slip. For a specific road, there is a linear stable region with relatively small slip-ratio. In this region, a constant slope rate $k = \mu/\lambda$ (see

red-dashed lines in Figure 6.1) can be used to describe the relationship between μ and λ for identifying the road condition. However, if the slip-ratio is too small (e.g., $\lambda < 0.005$) with very low excitation under normal driving, the correlation is fairly low [33]. On the other hand, at high slip-ratio, it is relatively easy to determine the road condition due to the obvious deviations among each curve. Beyond the optimal slip ratio, the normalized traction force is gradually saturated and slowly decreasing.

As reviewed in the beginning of this section, several different approaches have been proposed in literature for real-time estimation of tire-road friction coefficient. The 'cause-based' approach is too expensive due to its requirement of additional sensors, while the 'effect-based' approach based on a dynamic model is more attractive to be used in an actual production vehicle. In the model-based algorithm, researchers mostly utilize measurements of vehicle motion to obtain the expected estimation. Two types of vehicle dynamic model are normally considered, including longitudinal and lateral vehicle dynamics. Reference [84] discusses lateral vehicle motion-based estimation, where a cornering effect is added to the estimation algorithm. However, a measurement from a high-accuracy global positioning system (GPS) and a gyroscope is required, and the proposed algorithm was not verified for different roads. The longitudinal motion-based method is more general for the normal vehicle acceleration or deceleration. The vehicle model states can be obtained by either direct measurements or estimations using recursive Least Squares algorithm or observers developed. After these states are obtained, the relationship between actual friction coefficient and slip-ratio can be found. Based on this relationship, traditional model-based algorithm can be used to provide the maximum available friction coefficient. In order to get the current friction coefficient, both longitudinal force and vertical forces should be calculated first; and slip ratio can be calculated using Equation (6.1), based on the estimated (or measured) vehicle speed, measured wheel speed and effective tire radius. In this dissertation, an extended Kalman filter (EKF) based on a vehicle dynamic model will be developed as the traditional model-based algorithm. This model can be verified easily using CarSim simulations; however, it has certain issues when dealing with for test data. A new evaluation criterion of friction coefficient will be proposed based on data mining and

a Kriging model will be developed to fix these issues from these traditional algorithms.

6.2 Dynamic Model and Extended Kalman Filter

This section describes the vehicle mathematical model used and the structure of sequential EKF models for state estimation.

6.2.1 Vehicle and Wheel Model

A four-wheel vehicle dynamic model [35, 85] with a tire free body schematic is shown in Figure 6.2.

Note that, both pitch and roll motions are not considered.

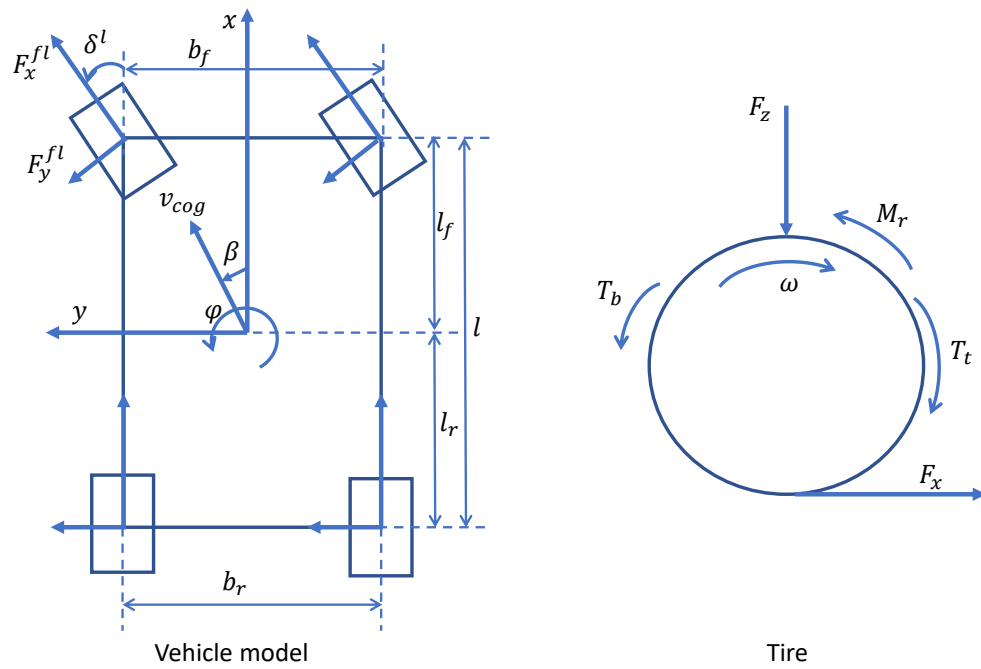


Figure 6.2 Vehicle and tire schematics

The corresponding dynamic equations defining longitudinal, lateral and yaw motions are listed below from Equations (6.2) to (6.4) :

$$M(\dot{v}_x - v_y \dot{\phi}) = F_x^{fl} \cos \delta^l - F_y^{fl} \sin \delta^l + F_x^{fr} \cos \delta^r - F_y^{fr} \sin \delta^r + F_x^{rl} + F_x^{rr} - F_x^a \quad (6.2)$$

$$M(\dot{v}_y + v_x \dot{\phi}) = F_y^{fl} \cos \delta^l + F_x^{fl} \sin \delta^l + F_y^{fr} \cos \delta^r + F_x^{fr} \sin \delta^r + F_y^{rl} + F_y^{rr} \quad (6.3)$$

$$\begin{aligned}
I_z \ddot{\varphi} = & (F_y^{fl} \cos \delta^l + F_x^{fl} \sin \delta^l + F_y^{fr} \cos \delta^r + F_x^{fr} \sin \delta^r) l_f \\
& - (F_y^{rr} + F_y^{rl}) l_r - 0.5(F_x^{fl} \cos \delta^l - F_y^{fl} \sin \delta^l \\
& - F_x^{fr} \cos \delta^r + F_y^{fr} \sin \delta^r) b_f - 0.5(F_x^{rl} - F_x^{rr}) b_r
\end{aligned} \tag{6.4}$$

where M is the vehicle mass; v_x and v_y are the longitudinal and lateral vehicle speeds, respectively; φ is the yaw angle; δ is the steering angle; F_x^a is the air drag force; and F_x^{ij} and F_y^{ij} represent the longitudinal and lateral forces of each wheel, respectively, where i denoted for the front and rear axles and j for the left and right wheels. The tire behavior is described in Equation (6.5) below [37].

$$I_r \dot{w} = T_t - T_b - M_r - F_x R_e \tag{6.5}$$

where I_r is the wheel moment of inertial; w is the wheel speed; T_t and T_b are the traction and brake torque, respectively; M_r is the rolling resistance associated with the tire pressure and wheel speed; and R is the tire radius.

6.2.2 Structure of Sequential Extended Kalman Filters

Note that vehicle speed can be obtained by integrating accelerometer signal, using GPS or wheel speed signals, or through sensor fusion. However, direct integrating acceleration leads to signal drift due to signal bias; and high-accuracy GPS sensor is often not available for production vehicles. Wheel speed could be used to approximate vehicle velocity, which could lead to using single signal source for slip-ratio calculation (see Equation (6.1)) since this calculation requires accurate difference between wheel and vehicle speed signals. In addition, tire forces are not available for all production vehicles, and it is necessary to estimate unknown states accurately.

The Kalman filter [82] is an algorithm that estimates system states from measurements. It uses a two-step process: estimating system states based on the system model and then correcting the estimation using noisy measurements. A S-EKF approach was proposed in [35] and adopted in this dissertation as a model-based algorithm for friction coefficient estimation. The S-EKF estimation signal flow is shown in Figure 6.3.

Note that in the first Kalman filter block, the random walk model in [35] is used.

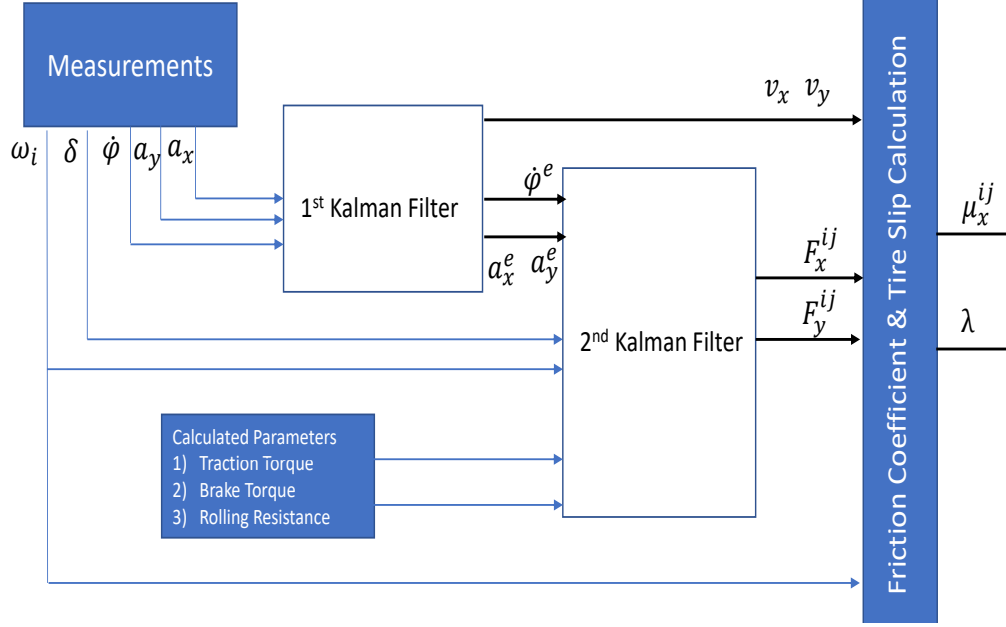


Figure 6.3 Schematics of sequential EKF algorithm

$$\begin{aligned}
 v_x(t) &= v_x(t-1) + \Delta t (a_x(t-1) + v_y(t-1)\dot{\varphi}(t-1)) \\
 v_y(t) &= v_y(t-1) + \Delta t (a_y(t-1) - v_x(t-1)\dot{\varphi}(t-1)) \\
 \dot{\varphi}(t) &= \dot{\varphi}(t-1); a_x(t) = a_x(t-1); a_y(t) = a_y(t-1)
 \end{aligned} \tag{6.6}$$

where system state vector is chosen as $x_s = [v_x, v_y, a_x, a_y, \dot{\varphi}]^T$ and system input vector as $u = [a_y, a_x, \dot{\varphi}]^T$; system outputs are the same as states, along with the estimated longitudinal and lateral velocities; the acceleration and yaw rate are filtered as part of inputs to the second EKF followed. In the second EKF, a random walk model $F_x^{ij}(t) = F_x^{ij}(t-1)$ is used for modeling force dynamics using outputs from the first EKF along with the wheel speed and steering angle. The wheel forces $[F_x^{fr}, F_x^{fl}, F_y^f, F_x^{rr}, F_x^{rl}, \text{ and } F_y^r]^T$ are states and system outputs, respectively.

6.3 CarSim Simulation and Result Discussion Using Test Data

It is straightforward to utilize CarSimTM [86] simulation data to verify the S-EKF, since it is able to provide vehicle dynamics based on predefined scenarios (driver inputs and road surface). An acceleration event along a straight line is simulated (see in Figure 6.4) and the vehicle velocity and forces are estimated well using S-EKF. Accordingly, the slip ratio and friction coefficient can be calculated and its relationship can be found in Figure 6.5.

Two observations are highlighted below based on simulation results:

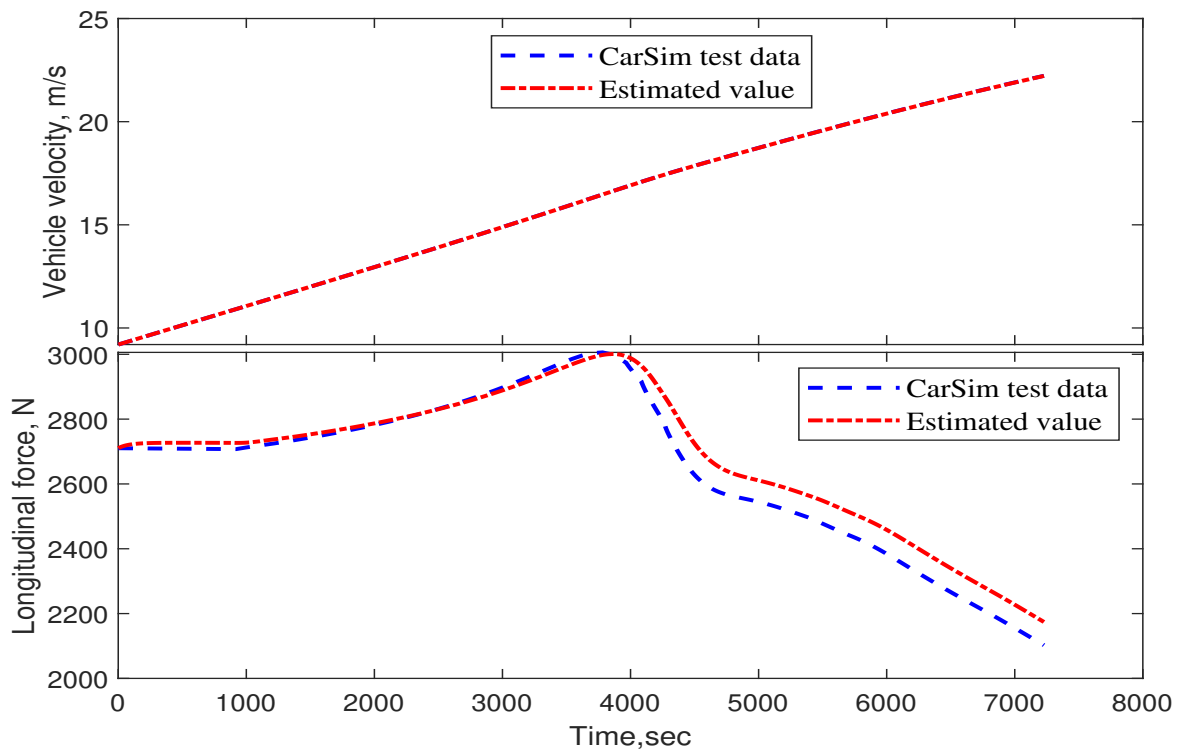


Figure 6.4 CarSimTM velocity and force estimation

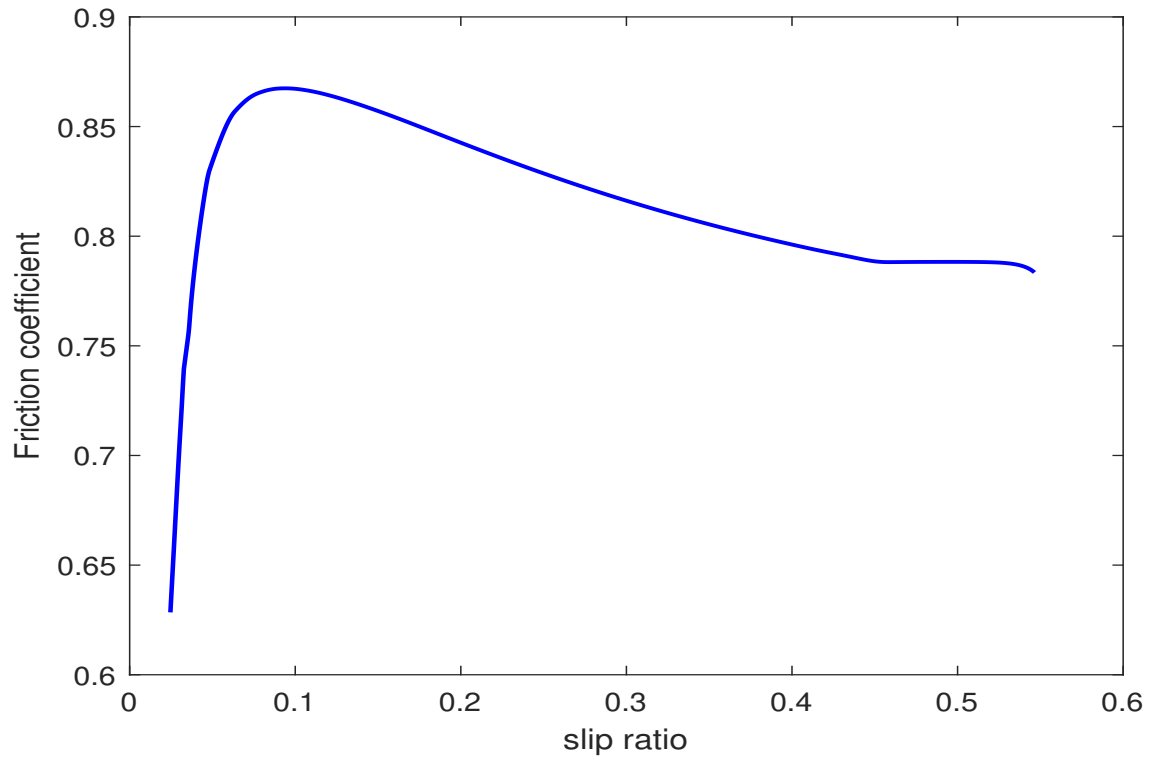


Figure 6.5 Friction coefficient vs. slip ratio

- 1) CarSimTM simulation environment assumes ideal driving situation without any road and sensor noises. White measurement noises can be added, but it cannot match with practice application scenario. A further verification using test data set with both system and measurement noises is necessary;
- 2) The simulation generates large slip-ratio data when vehicle is operated in an unstable region so that a complete nonlinear friction curve can be obtained. However, in practice, it is desired to operate the vehicle within a stable region (normal driving condition with small slip-ratio). Therefore, it is expected to use the slope method to estimate friction coefficient based on test data in practice.

Based on discussions above, three acceleration test data sets under different road conditions are used for validating the S-EKF algorithm. The acceleration profiles are shown in Figure 6.6 and the estimated linear slope is shown in Figure 6.7. A calibrated Magic formula based tire model [87] is used so that the projection between slope and friction coefficient can be constructed as shown in Figure 6.8.

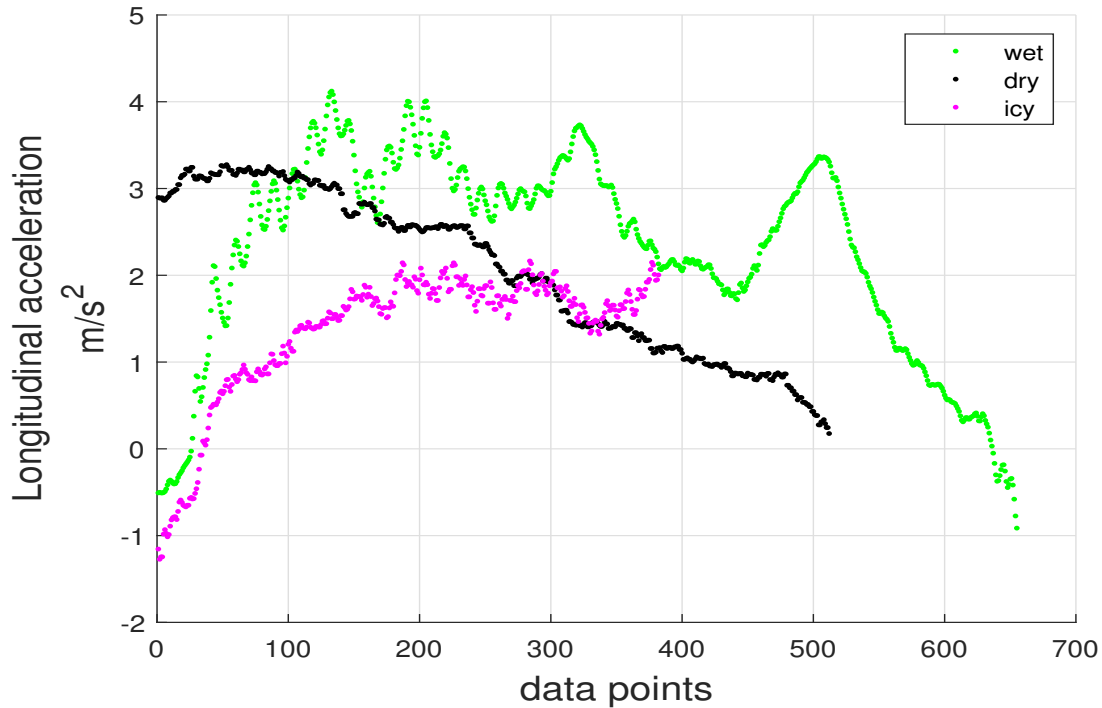


Figure 6.6 Three longitudinal acceleration profiles of test data

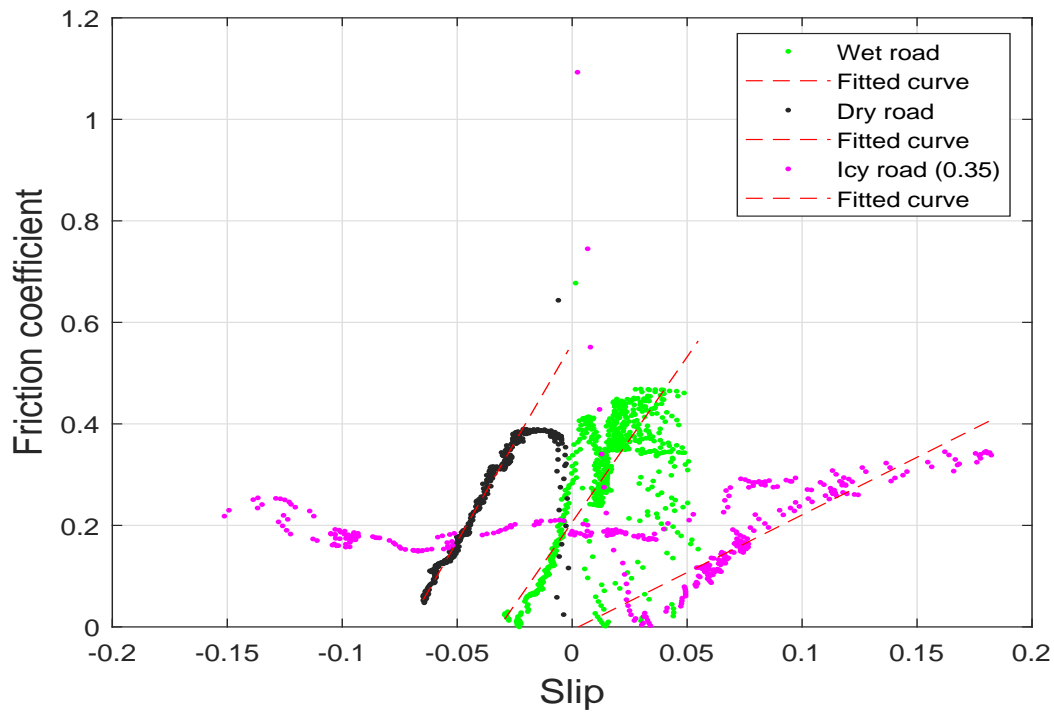


Figure 6.7 Simulation results using test data and sequential EKF methods

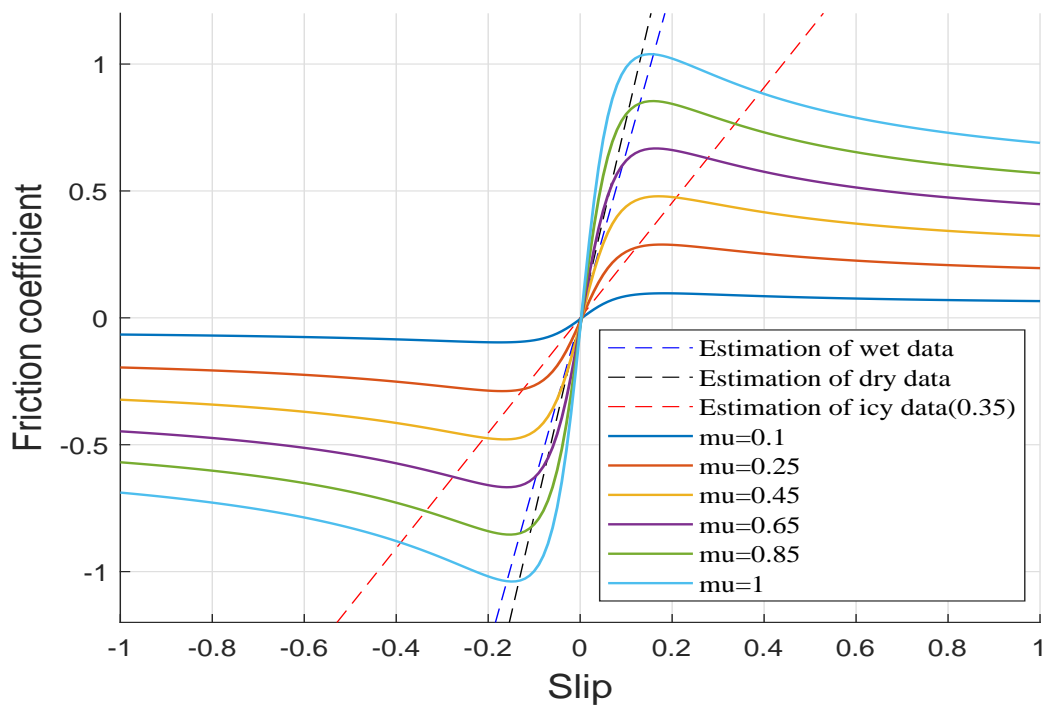


Figure 6.8 Slope approximations and tire model

From these results, it is noticed that the calculated slip-ratio becomes negative even under acceleration condition. Note that the estimation errors are mainly from two sources: a) there is no directly measured vehicle velocity available and its estimation is often based on wheel speed, leading to large slip-ratio calculation errors (for example, negative slip ratio shown in the figure); and b) the model used in S-EKF algorithm needs to be calibrated for different road conditions, and tuning a single estimation model to fit all road conditions is not possible. Another drawback of this algorithm is that the selected range of data points affects the slope approximation and the final estimated friction coefficient.

6.3.1 New Evaluation Criterion and Its Validation Using CarSim Data

Note that slip-ratio is a direct consequence of road condition variation. Let us revisit slip-ratio Equation (6.1) under acceleration, where the tire radius is assumed to be a constant for given specific tire and wheel speed can be measured accurately from the installed sensors used for anti-lock brake system (ABS); only the vehicle velocity needs to be estimated accurately. If this estimation could be avoided, the entire calculation and estimation process would be simplified without any new sensor. The derivative of slip-ratio is shown below.

$$\dot{\lambda} = \frac{R_e^2 \dot{\omega} v_x - R_e \omega \dot{v}_x}{(R_e \omega)^2} = \frac{R_e^2 \dot{\omega} \omega (1 - \lambda) - R_e \omega \dot{v}_x}{(R_e \omega)^2} \quad (6.7)$$

In this equation, all the variables can be obtained from the available sensor set of production vehicles (wheel speed and vehicle acceleration) except slip-ratio λ itself. If this term can be ignored (set to zero), calculations can be completed based on only available sensors. In the practical application scenarios, the vehicle is normally controlled within stable region, leading to a fairly small slip ratio. As a result, λ in the numerator of Equation (6.7) can be neglected, and derivative of slip-ratio can be approximated by the following format:

$$\dot{\lambda} = \frac{R_e^2 \dot{\omega} \omega - R_e \omega \dot{v}_x}{(R_e \omega)^2} \quad (6.8)$$

The difference before and after neglecting the λ term is plotted using CarSimTM data; see Figure 6.9. After neglecting the term λ , the absolute amplitude deviates from the true value, but the general trend does not change much. Especially, when slip ratio is small (less than 0.1), the blue

and black lines are almost the same with minor difference, which confirms the previous assumption for Equation (6.8). The deviation becomes more eminent when the slip-ratio is large. Another observation of slip-ratio derivative is that, there is a turning point similar to the relationship between friction coefficient and slip-ratio, which encourages us to investigate the possibility of using the slip-ratio derivative as a new evaluation criterion for tire-road friction coefficient estimation. A correlation between proposed criterion and friction coefficient needs to be developed.

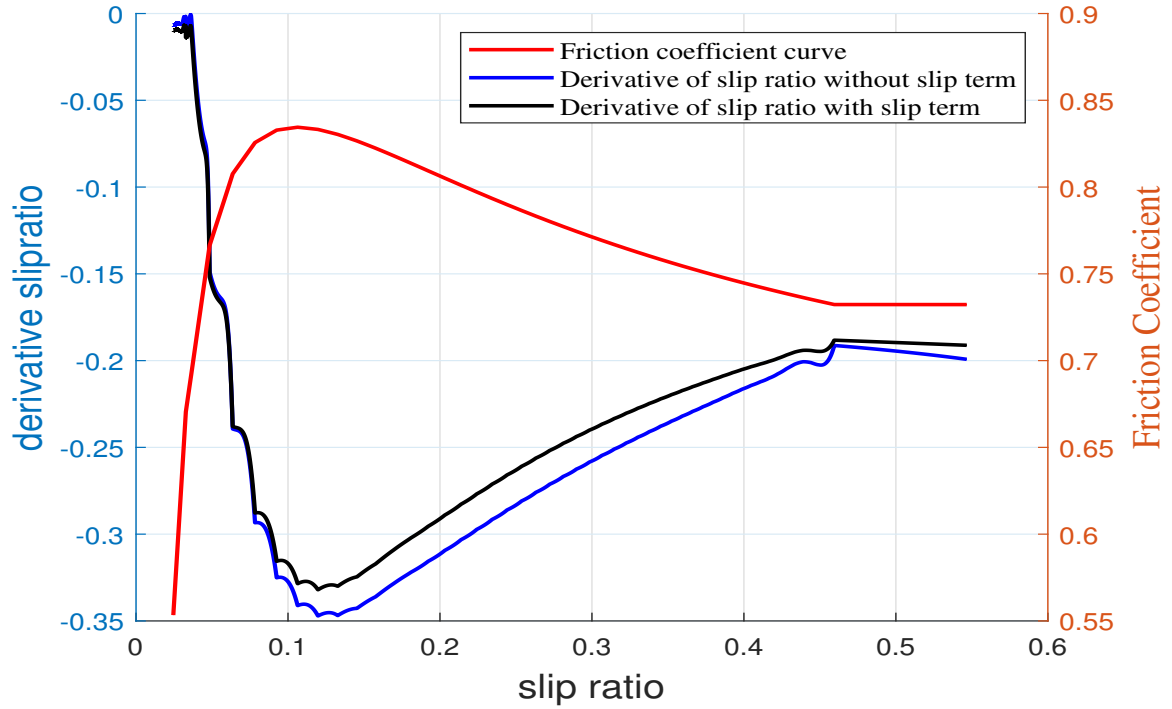


Figure 6.9 Derivative of slip-ratio

In order to investigate the potential of the proposed criterion, an acceleration event along a straight line is simulated in CarSimTM when friction coefficient is set to 0.25, 0.5 and 0.85, respectively. As shown in Figure 6.10, the fluctuation of slip-ratio derivative under different road conditions is quite different, although the environment and measurement noises are not included in this simulation. It is noticed that the lower the road friction coefficient is, the larger the oscillation magnitude. Thus the mean and standard deviation of slip-ratio derivative is expected to have high correlation with the road condition (friction coefficient).

Next, five different friction coefficients are set for CarSimTM simulations. The corresponding statistic circles are plotted in Figure 6.11, where the center of circle is the mean value of proposed

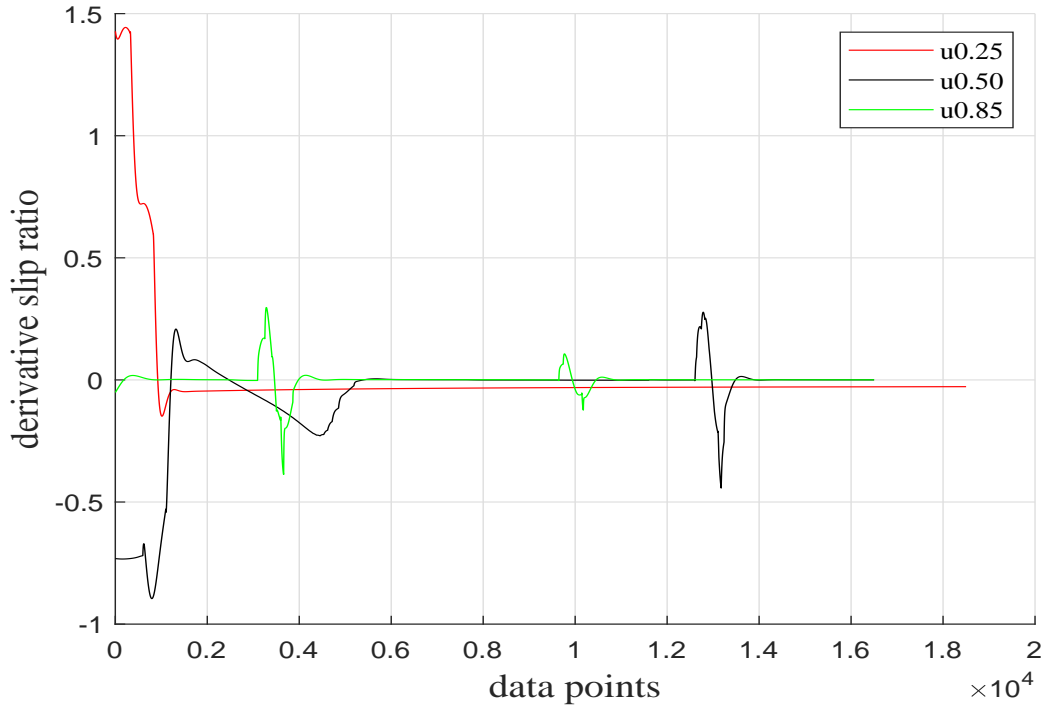


Figure 6.10 Derivative of slip-ratio on different road

criterion and the radius is associated with its variance information. Due to confidential requirements of sponsored project, only numerical pattern is presented, and the actual values of each signal cannot be shown. Note that each circle delegates the statistic information of slip-ratio derivative under a specific road condition. It is obvious that the radius of circle is strongly correlated to the road surface condition.

A data-driven Kriging model is used to generate relationship between the slip-ratio derivative and friction coefficient. The model is trained based on the known paired inputs (radius of statistic circle) and outputs (tire-road friction coefficient) data sets. The detailed development process of Kriging model can be found in Chapter 2 and [50]. Note that during the training process of Kriging model, a spatial distance Gaussian correlation is used for unknown design sites, which means the correlation between points decreases as their distance increases. The mean value curve of approximated Kriging model using CarSimTM data, along with validation results, is shown in Figure 6.12. Note that four data points is selected for training process, and the radius obtained when μ equals to 0.55 is kept aside for validation purpose. The estimation (red dot in the figure) is

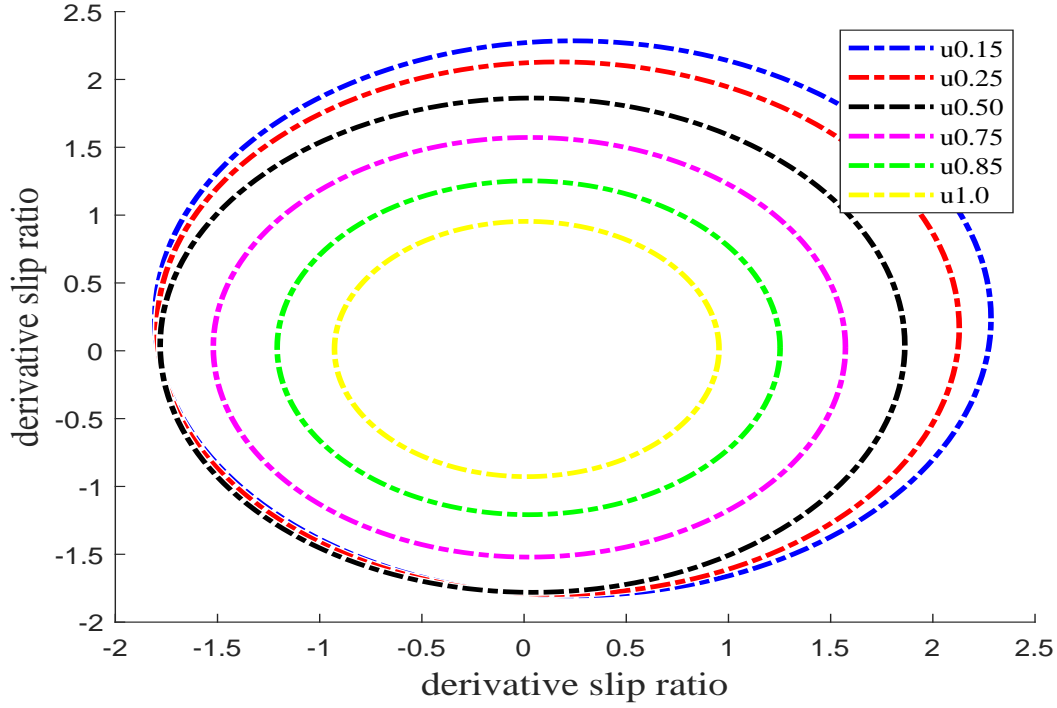


Figure 6.11 Statistic circles of new evaluation index on different road

close to true value (green dot in figure).

6.3.2 Proposed signal fusion estimation scheme

The estimation results perceived based on simulation data provide the useful information about using a slip-ratio derivative, calculated based on only production sensor signals, to estimate tire-road friction coefficient. A dual-path integrated estimation algorithm is proposed for friction coefficient estimation, which combines the S-EKF and statistic methods together. Note that the S-EKF algorithm is accurate with small slip-ratio and the proposed criterion is good under relatively larger slip-ratio. Therefore, these two criteria can be complemented each other. The proposed structure is shown in Figure 6.13 in details. For the top path, the traditional S-EKF algorithm is used with slope method, where the range of slip ratio (less than 0.1) is used to trigger the algorithm and confident coefficient α_1 is also calculated based on root mean squared error (RMSE) of approximated slope which delegates the goodness of fitted slope. In the bottom path, the proposed statistic information of slip ratio derivative is calculated and used to estimate the friction coefficient, where acceleration threshold (greater than $0.1m/s^2$) is used to activate this algorithm

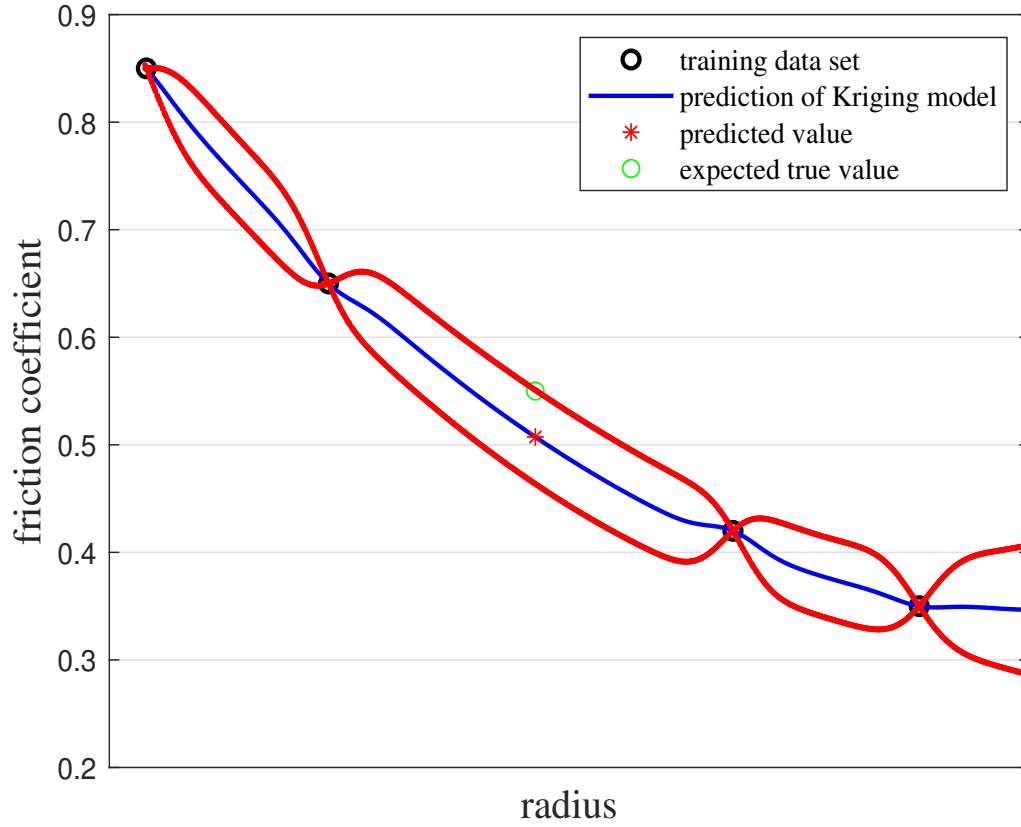


Figure 6.12 Surrogate model for new evaluation criterion and tire-road friction coefficient estimation using CarSimTM data

for estimating the friction coefficient estimation and the weighting coefficient α_2 will be calculated based on the variance predicted by surrogate model.

In this structure, once the required signals satisfy the pre-defined trigger signal, a fusion algorithm is used to provide the final predicted friction coefficient; see Equation (1.2) in introduction section and repeat here in Equation (6.9).

$$\mu_{final} = (\alpha_1 * \mu_{EKF} + \alpha_2 * \mu_{new}) / (\alpha_1 + \alpha_2) \quad (6.9)$$

where α_1 and α_2 are two coefficients provided by a calibrated correlations as shown in Figure 6.14.

6.3.3 Validation Using Test Data

The validation results based on CarSimTM simulation data indicate that the proposed criterion is able to estimate the road friction coefficient using only production sensor signals. In practice, the road surface is not uniform, measurement noise exists, and acceleration signal fluctuates. As a result, validation using physical vehicle test data is necessary for the proposed criterion and

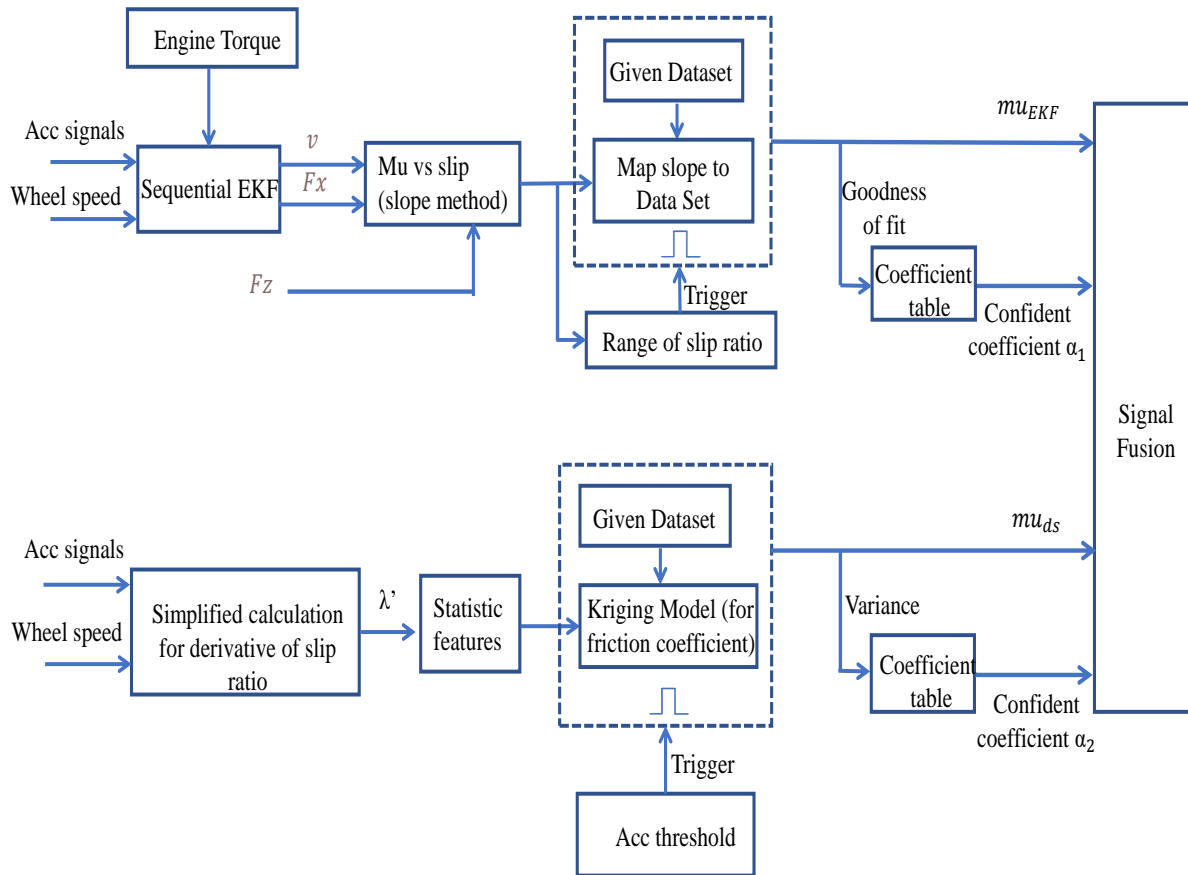


Figure 6.13 Proposed fusion estimation scheme

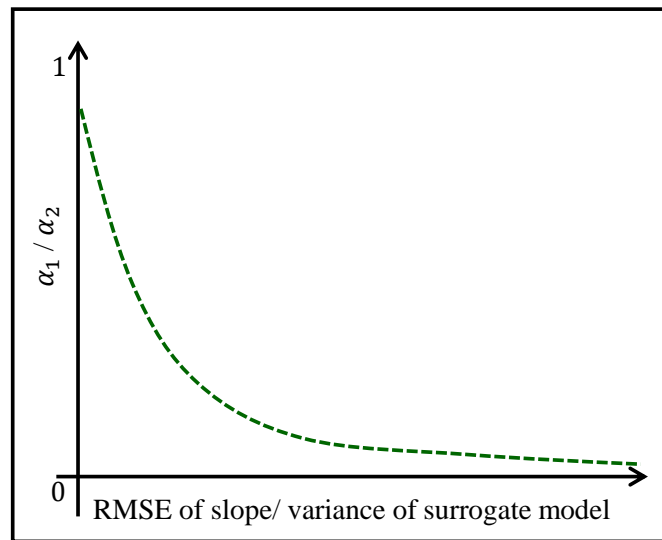


Figure 6.14 Correlations between weight coefficients (α_1 and α_2) and RSME/predicted variances

Kriging model. For this validation, vehicle acceleration tests were conducted on four different road conditions: dry, wet, icy and snow. Using Equation (6.8), slip-ratio derivative is calculated for each road condition, and time series data is shown in Figure 6.15. As shown in the zoom-in plot, the fluctuation changes associated with different road conditions are obvious and consistent with the phenomenon observed using CarSimTM data.

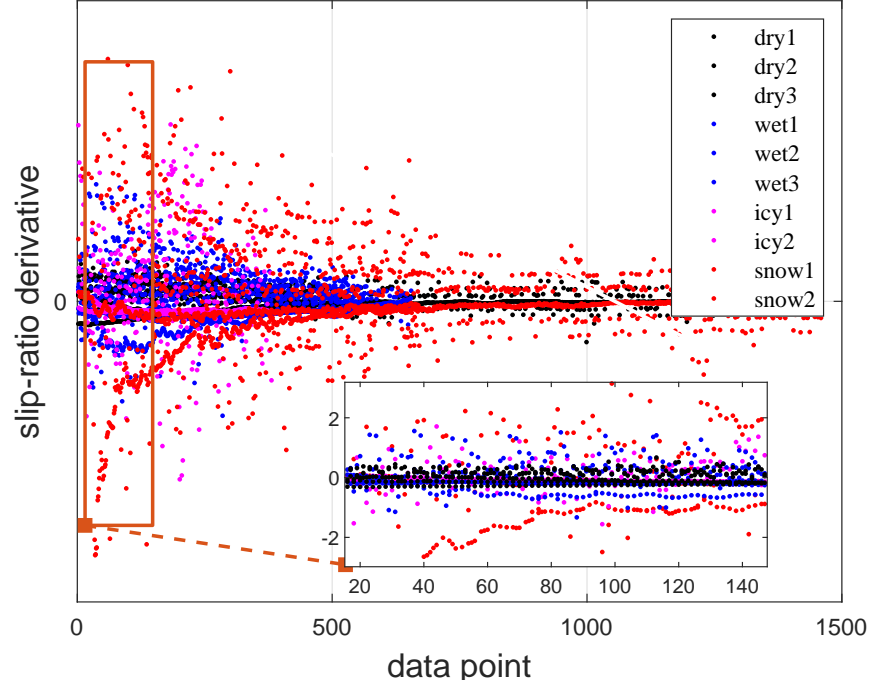


Figure 6.15 Slip-ratio derivative of test data on different road conditions

This phenomenon can be explained using the relationship between longitudinal force F and slip ratio λ . As stated in [88], for small tire slip-ratio, longitudinal force F can be approximated by $F = C\lambda$, where C is the tire longitudinal stiffness. A tire on the road with low friction coefficient has a relative low composite stiffness if tire and road are assumed to be serially-connected rigid system. Therefore, with the same longitudinal forces generated, a low friction road leads to high slip-ratio, and vice versa. This leads to a intensive slip variation before vehicle gets stable, which corresponds to a large slip-ratio change rate or magnitude of slip-ratio derivative for the data obtained under a low friction road surface.

Similar to CarSimTM validation results, the corresponding statistic circles are plotted in Figure 6.16. Compared with the CarSimTM simulation environment, except difference in terms of

environment noise, the actual road surface does not maintain a fixed friction coefficient. For example, it could be any value within the range from 0.8 to 0.9 for a dry road condition. Therefore, there could be several circles corresponding to the same road condition as shown in Figure 6.16. However, the differences or boundaries of circles for different road conditions are distinct.

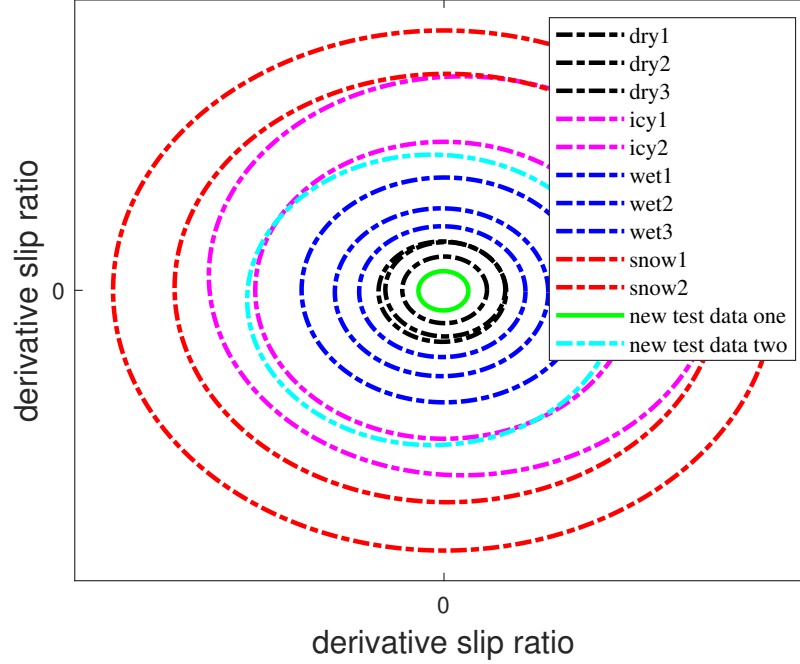


Figure 6.16 Statistic circles of new evaluation criterion on different road using test data

In order to obtain an one-to-one projection between radius of statistic circle and tire-road friction coefficient, mean radius of each road condition is calculated and used for training a deterministic Kriging model. This model is used in the lower path of proposed dual-path estimation algorithm for predicting unknown road surface. The mean and variance predicted by the trained Kriging model are shown in Figure 6.17. Note that, the variance of unknown region is smaller than those shown in Figure 6.12, which depends on the spatial distance between different design sites (radius of statistic circle). A closer distance generates a less variance between each two design sites. By averaging test data, the radius difference among the neighboring data points is less than those of simulation data.

Two additional tests are run over different road conditions, respectively. The target tire-road friction coefficient of unknown road is expected to be within $[0.80, 0.95]$ for new test data one and

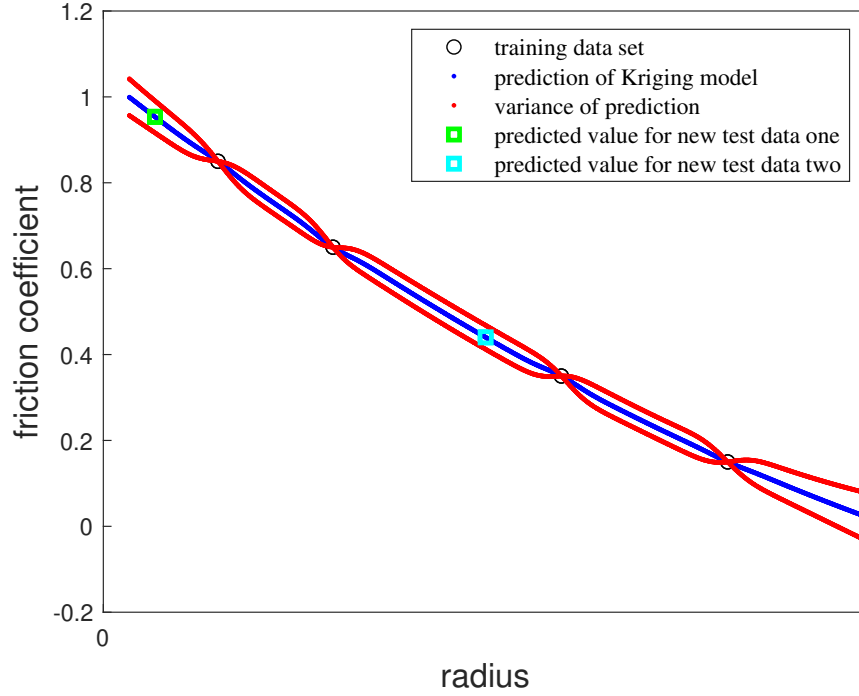


Figure 6.17 Surrogate model of proposed evaluation criterion using test data

[0.3, 0.5] for new test data two. The estimates using the surrogate model are 0.95 and 0.44 (see Figure 6.17); and corresponding statistic circles are also shown in Figure 6.16. The estimates for two new test data using S-EKF are 0.70 and 0.53, respectively. Accordingly the final results after signal fusion are 0.825 and 0.485.

6.4 Conclusions

This paper starts with a full review of existing algorithms for tire-road friction coefficient estimation. A S-EKF based on vehicle model and slope method is studied as the baseline method using CarSimTM and test data. After discussing about the drawbacks of S-EKF method, a new evaluation criterion based on slip ratio derivative is proposed and fully investigated by utilizing its statistic features observed from both simulation and test data sets. These statistic features provide a new approach for tire-road friction coefficient estimation in terms of improved accuracy, simplified implementation, and reduced cost. The proposed evaluation criterion shows a strong correlation to road condition and a data-driven Kriging model can be trained for estimation. A signal fusion scheme is also proposed based both S-EKF and slip ratio derivative methods, which provides a reliable estimation under different road conditions. The data-driven based Kriging model has a

potential to be used for real-time estimation of tire-road friction coefficient, which is very important for vehicle dynamic control.

CHAPTER 7

CONCLUSION AND FUTURE WORK

In summary, this dissertation presents two different application scenarios in the automotive estimation and control field using the data-driven based algorithm. These data-driven methods assist our current algorithm to improve the system performance. For the application of engine knock borderline prediction, a Bayesian optimization algorithm with the Kriging model is used to replace the earlier mapping-based method. Both simulation and experimental results shown in Chapter 3 demonstrated the applicability of data-driven surrogate model assisted optimization method for reducing overall evaluation budget. For this specific application, experiments were conducted and show the efficacy of proposed algorithm for predicting knock borderline, after necessary modifications are made to fit statistic information. The potential of using obtained surrogate models integrated with likelihood ratio controller for real-time updating is also validated in Chapter 4 through both simulation and experimental studies. In Chapter 5, a compensation strategy is added to the baseline control within proposed knock control architecture. The cycle-to-cycle knock variation is reduced obviously by using the feedback of exhaust temperature and model based LQG controller. Finally, in Chapter 6, the vehicle level application of tire-road friction coefficient estimation was presented. Utilizing this new criterion, a Kriging model can be trained based on the data set. The statistic information of a new evaluation criterion is discussed and studied in simulations. The results demonstrated potential of this new criterion for estimating tire-road friction, leading to the proposed scheme that combines two estimation algorithms together.

Through the whole process, it is easy to notice that the great role of statistic information (including mean, variance, 3-sigma rules and probability distribution) when the data-driven methodology is used. All these statistic characteristics, coupled with other matured algorithms plus certain necessary modifications, provide a novel perspective to many traditional problems. Both prediction efficiency and accuracy are improved. In this dissertation, the current research only presents these results in specific areas of engine and vehicle systems, but the potential of using statistic principles and data-driven model is foreseeable. They can be further applied to other real-time

vehicle applications to improve the estimation efficiency and accuracy, which will be helpful for rapid development in autonomous vehicle industry.

Following future work is recommended within the scope of novel performance prediction algorithms for automotive systems using statistic features of data-driven models.

- Future work for offline calibration using SMAO algorithm includes extending the current design space to a dimension higher than two, so that more control parameters could be added for optimizing the borderline knock. A systematic method, that determines the sensitive factors affecting knock borderline, needs to be developed for different applications.
- For cycle-wised compensation, now it is only applied for a fixed operation condition. An open problem is to study the proposed approach under different speed and load conditions. Among different operational conditions, a single identified model currently used may not be able to provide an accurate prediction. A parameter varying model may need to be developed for this. Furthermore, a linear parameter varying (LPV) control could be a solution for this real-time control problem under dynamic operational condition.
- The current verification through CarSim data shows that the proposed criterion could be used to estimate the road conditions using only production sensors. This algorithm provides an estimation based on the history data. It can be extended to predict the future road condition. This challenge could be achieved by utilizing information over vehicle-to-vehicle (V2V) and cloud network. The predicted information can be shared among vehicles and stored in cloud and the prediction from these vehicles in front could be used for estimating the friction coefficient of target vehicle.

BIBLIOGRAPHY

- [1] U. Kiencke and L. Nielsen, “Automotive control systems: for engine, driveline, and vehicle,” 2000.
- [2] J. B. Heywood, *Internal combustion engine fundamentals*. McGraw-hill, 1988.
- [3] K. C. Prajapati, R. Patel, and R. Sagar, “Hybrid vehicle: A study on technology,” *International Journal of Engineering Research & Technology (IJERT)*, vol. 3, no. 12, p. 8, 2014.
- [4] Z. Wang, H. Liu, and R. D. Reitz, “Knocking combustion in spark-ignition engines,” *Progress in Energy and Combustion Science*, vol. 61, pp. 78–112, 2017.
- [5] Z. Lou and G. Zhu, “Review of advancement in variable valve actuation of internal combustion engines,” *Applied Sciences*, vol. 10, no. 4, p. 1216, 2020.
- [6] R. C. Li and G. G. Zhu, “Model-based knock prediction and its stochastic feedforward compensation,” in *2020 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 637–642, IEEE, 2020.
- [7] G. G. Zhu, I. Haskara, and J. Winkelman, “Closed-loop ignition timing control for si engines using ionization current feedback,” *IEEE transactions on control systems technology*, vol. 15, no. 3, pp. 416–427, 2007.
- [8] G. Zhu, I. Haskara, and J. Winkelman, “Stochastic limit control and its application to spark limit control using ionization feedback,” in *Proceedings of the 2005, American Control Conference, 2005.*, pp. 5027–5034, IEEE, 2005.
- [9] Y. Zhang, X. Shen, Y. Wu, and T. Shen, “On-board knock probability map learning-based spark advance control for combustion engines,” *International Journal of Engine Research*, vol. 20, no. 10, pp. 1073–1088, 2019.
- [10] J. Tang, G. G. Zhu, and Y. Men, “Review of engine control-oriented combustion models,” *International Journal of Engine Research*, p. 1468087421992955, 2021.
- [11] R. Isermann, J. Schaffnit, and S. Sinsel, “Hardware-in-the-loop simulation for the design and testing of engine-control systems,” *Control Engineering Practice*, vol. 7, no. 5, pp. 643–653, 1999.
- [12] J. Powell, “A review of ic engine models for control system design,” *IFAC Proceedings Volumes*, vol. 20, no. 5, pp. 235–240, 1987.
- [13] I. Arsie, C. Pianese, G. Rizzo, R. Flora, and G. Serra, “A computer code for si engine control and powertrain simulation,” *SAE transactions*, pp. 935–949, 2000.
- [14] R. F. Turkson, F. Yan, M. K. A. Ali, and J. Hu, “Artificial neural network applications in the calibration of spark-ignition engines: An overview,” *Engineering Science and Technology, an International Journal*, vol. 19, no. 3, pp. 1346–1359, 2016.

- [15] X. Luo, M. Donkers, B. de Jager, and F. Willems, "Systematic design of multivariable fuel injection controllers for advanced diesel combustion," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 5, pp. 1979–1990, 2018.
- [16] B. P. Maldonado, N. Li, I. Kolmanovsky, and A. G. Stefanopoulou, "Learning reference governor for cycle-to-cycle combustion control with misfire avoidance in spark-ignition engines at high exhaust gas recirculation–diluted conditions," *International Journal of Engine Research*, p. 1468087420929109, 2020.
- [17] S. Leonhardt, N. Muller, and R. Isermann, "Methods for engine supervision and control based on cylinder pressure information," *IEEE/ASME transactions on mechatronics*, vol. 4, no. 3, pp. 235–245, 1999.
- [18] M. Kang, K. Sata, and A. Matsunaga, "Control-oriented cyclic modeling method for spark ignition engines," *IFAC-PapersOnLine*, vol. 51, no. 31, pp. 448–453, 2018.
- [19] E. Hellström, A. G. Stefanopoulou, and L. Jiang, "Cyclic variability and dynamical instabilities in autoignition engines with high residuals," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 5, pp. 1527–1536, 2012.
- [20] R. C. Li and G. G. Zhu, "A real-time pressure wave model for knock prediction and control," *International Journal of Engine Research*, vol. 22, no. 3, pp. 986–1000, 2021.
- [21] L. Zhou, A. Shao, J. Hua, H. Wei, and D. Feng, "Effect of retarded injection timing on knock resistance and cycle to cycle variation in gasoline direct injection engine," *Journal of Energy Resources Technology*, vol. 140, no. 7, 2018.
- [22] N. Kawahara and E. Tomita, "Visualization of auto-ignition and pressure wave during knocking in a hydrogen spark-ignition engine," *International Journal of Hydrogen Energy*, vol. 34, no. 7, pp. 3156–3163, 2009.
- [23] M. Abu-Qudais, "Exhaust gas temperature for knock detection and control in spark ignition engine," *Energy conversion and management*, vol. 37, no. 9, pp. 1383–1392, 1996.
- [24] Z. Wang, J. Zhu, L. Zhang, and Y. Wang, "Automotive abs/dyc coordinated control under complex driving conditions," *IEEE access*, vol. 6, pp. 32769–32779, 2018.
- [25] P. Khatun, C. M. Bingham, N. Schofield, and P. Mellor, "Application of fuzzy control algorithms for electric vehicle antilock braking/traction control systems," *IEEE Transactions on Vehicular Technology*, vol. 52, no. 5, pp. 1356–1364, 2003.
- [26] S. Anwar, "Yaw stability control of an automotive vehicle via generalized predictive algorithm," in *Proceedings of the 2005, American Control Conference, 2005.*, pp. 435–440, IEEE, 2005.
- [27] J. Tang, H. B. Jiang, G. Q. Geng, and N. W. Xue, "Research on modeling of eps with bond-graph theory and simulating of steering maneuverability," in *Key Engineering Materials*, vol. 464, pp. 736–740, Trans Tech Publ, 2011.

- [28] G. G. Zhu and C. Miao, “Real-time co-optimization of vehicle route and speed using generic algorithm for improved fuel economy,” *Mechanical Engineering*, vol. 141, no. 03, pp. S08–S15, 2019.
- [29] T. D. Gillespie, “Fundamentals of vehicle dynamics,” tech. rep., SAE Technical Paper, 1992.
- [30] W. Wei, H. Dourra, and G. Zhu, “Vehicle tire traction torque estimation using a dual extended kalman filter,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 144, no. 3, 2022.
- [31] H. Lee and M. Tomizuka, “Adaptive vehicle traction force control for intelligent vehicle highway systems (ivhss),” *IEEE Transactions on Industrial Electronics*, vol. 50, no. 1, pp. 37–47, 2003.
- [32] R. Rajamani, N. Piyabongkarn, J. Lew, K. Yi, and G. Phanomchoeng, “Tire-road friction-coefficient estimation,” *IEEE Control Systems Magazine*, vol. 30, no. 4, pp. 54–69, 2010.
- [33] R. Rajamani, G. Phanomchoeng, D. Piyabongkarn, and J. Y. Lew, “Algorithms for real-time estimation of individual wheel tire-road friction coefficients,” *IEEE/ASME Transactions on Mechatronics*, vol. 17, no. 6, pp. 1183–1195, 2011.
- [34] K. B. Singh and S. Taheri, “Estimation of tire–road friction coefficient and its application in chassis control systems,” *Systems Science & Control Engineering*, vol. 3, no. 1, pp. 39–61, 2015.
- [35] J. J. Castillo Aguilar, J. A. Cabrera Carrillo, A. J. Guerra Fernández, and E. Carabias Acosta, “Robust road condition detection system using in-vehicle standard sensors,” *Sensors*, vol. 15, no. 12, pp. 32056–32078, 2015.
- [36] M. Acosta, S. Kanarachos, and M. Blundell, “Road friction virtual sensing: A review of estimation techniques with emphasis on low excitation approaches,” *Applied Sciences*, vol. 7, no. 12, p. 1230, 2017.
- [37] S. Khaleghian, A. Emami, and S. Taheri, “A technical survey on tire-road friction estimation,” *Friction*, vol. 5, no. 2, pp. 123–146, 2017.
- [38] O. L. De Weck, “Multiobjective optimization: History and promise,” in *Invited Keynote Paper, GL2-2, The Third China-Japan-Korea Joint Symposium on Optimization of Structural and Mechanical Systems, Kanazawa, Japan*, vol. 2, p. 34, 2004.
- [39] A. Cassioli and F. Schoen, “Global optimization of expensive black box problems with a known lower bound,” *Journal of Global Optimization*, vol. 57, no. 1, pp. 177–190, 2013.
- [40] J. Knowles, “Parego: a hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems,” *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 1, pp. 50–66, 2006.
- [41] D. Huang, T. T. Allen, W. I. Notz, and N. Zeng, “Global optimization of stochastic black-box systems via sequential kriging meta-models,” *Journal of global optimization*, vol. 34, no. 3, pp. 441–466, 2006.

- [42] M. J. Asher, B. F. Croke, A. J. Jakeman, and L. J. Peeters, "A review of surrogate models and their application to groundwater modeling," *Water Resources Research*, vol. 51, no. 8, pp. 5957–5973, 2015.
- [43] R. P. Liem, C. A. Mader, and J. R. Martins, "Surrogate models and mixtures of experts in aerodynamic performance prediction for aircraft mission analysis," *Aerospace Science and Technology*, vol. 43, pp. 126–151, 2015.
- [44] A. Pal, L. Zhu, Y. Wang, and G. Zhu, "Constrained surrogate-based engine calibration using lower confidence bound," *IEEE/ASME Transactions on Mechatronics*, 2021.
- [45] A. Pal, Y. Wang, L. Zhu, and G. G. Zhu, "Multi-objective surrogate-assisted stochastic optimization for engine calibration," *Journal of Dynamic Systems, Measurement, and Control*, vol. 143, no. 10, p. 101004, 2021.
- [46] K. Deb, *Multi-objective optimization using evolutionary algorithms*, vol. 16. John Wiley & Sons, 2001.
- [47] A. Pal, W. Wei, L. Zhu, Y. Wang, and G. Zhu, "Dynamic system identification for a nonlinear vehicle model using q -markov cover under different operational conditions," in *2021 American Control Conference (ACC)*, pp. 2830–2835, IEEE, 2021.
- [48] G. G. Zhu, "Weighted multirate q -markov cover identification using prbs—an application to engine systems," *Mathematical Problems in Engineering*, vol. 6, no. 2-3, pp. 201–224, 2000.
- [49] K. Liu, R. Skelton, and J. Sharkey, "Modeling hubble space telescope flight data by q -markov cover identification," *Journal of guidance, control, and dynamics*, vol. 17, no. 2, pp. 250–256, 1994.
- [50] J. Tang, A. Pal, W. Dai, C. Archer, J. Yi, and G. Zhu, "Stochastic bayesian optimization for predicting borderline knock," *International Journal of Engine Research*, p. 14680874211065237, 2021.
- [51] B. Ankenman, B. L. Nelson, and J. Staum, "Stochastic kriging for simulation metamodeling," in *2008 Winter Simulation Conference*, pp. 362–370, IEEE, 2008.
- [52] A. Forrester, A. Sobester, and A. Keane, *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons, 2008.
- [53] A. I. Forrester, A. J. Keane, and N. W. Bressloff, "Design and analysis of" noisy" computer experiments," *AIAA journal*, vol. 44, no. 10, pp. 2331–2339, 2006.
- [54] S. N. Lophaven, H. B. Nielsen, J. Søndergaard, *et al.*, *DACE: a Matlab kriging toolbox*, vol. 2. Citeseer, 2002.
- [55] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, "Design and analysis of computer experiments," *Statistical science*, vol. 4, no. 4, pp. 409–423, 1989.

- [56] T. Mukhopadhyay, S. Chakraborty, S. Dey, S. Adhikari, and R. Chowdhury, “A critical assessment of kriging model variants for high-fidelity uncertainty quantification in dynamics of composite shells,” *Archives of Computational Methods in Engineering*, vol. 24, no. 3, pp. 495–518, 2017.
- [57] P. J. Diggle, J. A. Tawn, and R. A. Moyeed, “Model-based geostatistics,” *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 47, no. 3, pp. 299–350, 1998.
- [58] D. R. Jones, M. Schonlau, and W. J. Welch, “Efficient global optimization of expensive black-box functions,” *Journal of Global optimization*, vol. 13, no. 4, pp. 455–492, 1998.
- [59] D. Zhan, Y. Cheng, and J. Liu, “Expected improvement matrix-based infill criteria for expensive multiobjective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 6, pp. 956–975, 2017.
- [60] K. Deb, *Optimization for engineering design: Algorithms and examples*. PHI Learning Pvt. Ltd., 2012.
- [61] J. D. Naber, J. R. Blough, D. Frankowski, M. Goble, and J. E. Szpytman, “Analysis of combustion knock metrics in spark-ignition engines,” *SAE Transactions*, pp. 223–243, 2006.
- [62] A. Pal, *Model Assisted Iterative Calibration of Internal Combustion Engines*. PhD thesis, Michigan State University, 2021.
- [63] G. G. Zhu, C. F. Daniels, and J. Winkelman, “Mbt timing detection and its closed-loop control using in-cylinder pressure signal,” tech. rep., SAE Technical Paper, 2003.
- [64] M. Mittal, G. Zhu, and H. Schock, “Fast mass-fraction-burned calculation using the net pressure method for real-time applications,” *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 223, no. 3, pp. 389–394, 2009.
- [65] X. Zhen, Y. Wang, S. Xu, Y. Zhu, C. Tao, T. Xu, and M. Song, “The engine knock analysis—an overview,” *Applied Energy*, vol. 92, pp. 628–636, 2012.
- [66] K. M. Chun and K. W. Kim, “Measurement and analysis of knock in a si engine using the cylinder pressure and block vibration signals,” *SAE transactions*, pp. 56–62, 1994.
- [67] K. Deb, R. Hussein, P. C. Roy, and G. Toscano-Pulido, “A taxonomy for metamodeling frameworks for evolutionary multiobjective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 1, pp. 104–116, 2018.
- [68] J. C. P. Jones, J. Frey, and K. R. Muske, “A statistical likelihood based knock controller,” *IFAC Proceedings Volumes*, vol. 43, no. 7, pp. 809–814, 2010.
- [69] A. Thomasson, H. Shi, T. Lindell, L. Eriksson, T. Shen, and J. C. P. Jones, “Experimental validation of a likelihood-based stochastic knock controller,” *IEEE Transactions on Control Systems Technology*, vol. 24, no. 4, pp. 1407–1418, 2015.
- [70] J. C. P. Jones, J. M. Spelina, and J. Frey, “Likelihood-based control of engine knock,” *IEEE Transactions on Control Systems Technology*, vol. 21, no. 6, pp. 2169–2180, 2013.

- [71] X. Shen, Y. Zhang, T. Shen, and C. Khajorntraidet, "Spark advance self-optimization with knock probability threshold for lean-burn operation mode of si engine," *Energy*, vol. 122, pp. 1–10, 2017.
- [72] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, p. 20150202, 2016.
- [73] F. Ainouz and J. Vedholm, "Mean value model of the gas temperature at the exhaust valve," 2009.
- [74] L. Eriksson, "Mean value models for exhaust system temperatures," *SAE Transactions*, pp. 753–767, 2002.
- [75] M. Wallson, "Estimation of engine gas temperatures during pressure transients," 2018.
- [76] S. L. Brunton and J. N. Kutz, *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2022.
- [77] L. Malm, "Control oriented modeling of engine out temperatures," 2018.
- [78] H. Kerai and A. Verem, "Physically based models for predicting exhaust temperatures in si engines," 2016.
- [79] G. G. Zhu, R. E. Skelton, and P. Li, "Q-markov cover identification using pseudo-random binary signals," *International Journal of Control*, vol. 62, no. 6, pp. 1273–1290, 1995.
- [80] A. Lindquist, "On feedback control of linear stochastic systems," *SIAM Journal on Control*, vol. 11, no. 2, pp. 323–343, 1973.
- [81] T. T. Georgiou and A. Lindquist, "The separation principle in stochastic control, redux," *IEEE Transactions on Automatic Control*, vol. 58, no. 10, pp. 2481–2494, 2013.
- [82] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, vol. 82, pp. 35–45, 03 1960.
- [83] J. Tang, W. Dai, C. Archer, J. Yi, and G. Zhu, "Borderline knock adaptation based on online updated surrogate models," *International Journal of Engine Research*, 2022.
- [84] J.-O. Hahn, R. Rajamani, and L. Alexander, "Gps-based real-time identification of tire-road friction coefficient," *IEEE Transactions on Control Systems Technology*, vol. 10, no. 3, pp. 331–343, 2002.
- [85] J. J. Castillo, J. A. Cabrera, A. J. Guerra, and A. Simon, "A novel electrohydraulic brake system with tire-road friction estimation and continuous brake pressure control," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 3, pp. 1863–1875, 2015.
- [86] T. D. Gillespie, "Carsim data manual," *Ann Arbor, Michigan: Mechanical Simulation Corporation*, 2004.

- [87] H. B. Pacejka and E. Bakker, “The magic formula tyre model,” *Vehicle system dynamics*, vol. 21, no. S1, pp. 1–18, 1992.
- [88] W. Wei, H. Dourra, and G. Zhu, “Transfer case clutch torque modeling and validation under slip and overtaken conditions,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 143, no. 7, 2021.