

Real-time Modelling and Parallel Optimisation of a Gasoline Direct Injection Engine

Wen Gu, Dezong Zhao, *Senior Member, IEEE*, and Byron Mason

Abstract—With the increasing complexity of engines and number of control parameters, optimal engine parameter sets need to be searched in the high dimensionality. Traditional calibration methods are too complicated, expensive and time-consuming. The model-based optimisation is of critical importance for engine fuel efficiency improvement and exhaust emissions reduction. The optimisation highly depends on the model accuracy. In this paper, a multi-layer modelling method is proposed, which can be used to generate the engine model at arbitrary operating points in real time with high accuracy. An enhanced heuristic-algorithm-based optimiser is combined with the real-time modelling method to perform a parallel optimisation. The proposed modelling and optimisation strategy can achieve the minimal fuel consumption fast and accurately. This strategy has been successfully verified using experimental data sets.

I. INTRODUCTION

The automotive industry is encountering huge challenges in developing the next generation internal combustion engines due to stricter economic and ecological requirements [1]. To meet such requirements, new engines are equipped with advanced technologies, for example variable valve lift, variable valve timing and exhaust gas recirculation [2]. With the increasing number of engine actuators, optimal settings are searched in a high-dimensional space. Therefore, the traditional engine calibration method is too complicated, expensive and time-consuming to solve these problems [3]. As a result, the current calibration methods which focus on steady-state mapping, is not capable of dealing with increasing demands of accuracy in the future engines development [4], [5].

In the last decade, many efforts have been put into model-based optimisation methods [6]. The model can be physically-based, or can be described by equations and artificial intelligence models. Model-based optimisation algorithms are developed to calculate the optimal engine parameter setting at steady-state operating points [7].

A 1-D simulation engine model is designed in the GT-power by [8]. After that, a Genetic Algorithm (GA) based optimiser is applied to the optimisation based on the 1-D model. In [3], a model-based calibration method for a homogeneous charge compression ignition engine is demonstrated. A co-simulation is performed between the engine model built in Simulink and an evolutionary algorithm written in Java. Recently, the online optimisation has gained in popularity where measurement, modelling and optimisation are treated

as integrated [9], [10]. System identification is critical for modelling. In [11], the engine operating region is segmented into several areas. The state-space model is identified at the centre of each area for the controller development. A polynomial-based system identification is applied in [4] and [12]. At each operating point, the polynomial equation is implemented to represent the engine behaviour. Bilinear interpolation [4] and barycentric interpolation [12] are used between operating points to calculate model outputs. However, the polynomial model cannot address high-dimensional problems due to the increasing number of control parameters [13]. In [9], a machine learning algorithm called Gaussian process is introduced. It shows that the Gaussian model is suitable for the stationary engine modelling since the required training data size is small. Further investigations of the Gaussian model are fulfilled in [13], where the Gaussian model is combined with a GA-based optimiser. Then an optimisation is performed at a particular operating point based on a real data set. Other popular system identification methods, for example multilayer perceptron neural network and local linear model trees method, are studied by [6] and [14] respectively. Although some model-based optimisation work have been done, a proper modelling method is still lacking for real-time modelling in a high-dimensional space. This paper proposes a global-local modelling architecture to generate models at arbitrary operating points in real time. Meanwhile, the architecture is suitable for dealing with multiple input variables. In addition, a heuristic-algorithm-based optimiser is improved for higher robustness.

The paper is organised as follows. After the introduction in Section I, the system description and optimisation problem formulation are given in Section II. The real-time modelling algorithm is presented in Section III. The parallel optimisation method is introduced in Section IV. Validation results of the optimisation are demonstrated in Section V. Finally, conclusions are summarised in Section VI.

II. SYSTEM DESCRIPTION

A. Engine Description

A control schematic of a gasoline direct injection (GDI) engine equipped with the variable camshaft timing (VCT) is shown in Fig. 1, while the parameters are defined in Table. I. The value of ξ_{ref} is determined by N and θ_0 . Then VCT actuators will adjust ξ_{cam} based on N and ξ_{ref} . In the block diagram, T is the output of the system, which is affected by m_{cyl} , N , σ_{spk} and A/F. The objective of this work is to calculate the ξ_{ref} at arbitrary operating points based on the nonlinear relationship between cam timing and

W. Gu, D. Zhao, and B. Mason are with the Department of Aeronautical and Automotive Engineering, Loughborough University, Loughborough LE11 3TU, U.K. (e-mail: w.gu@lboro.ac.uk, d.zhao@lboro.ac.uk, b.mason2@lboro.ac.uk).

TABLE I: Engine parameters

Variable	Description
A/F	Air fuel ratio.
N	Engine speed.
m_{cyl}	Mass of air into cylinder.
P_m	Manifold pressure.
P_m^*	Desired manifold pressure.
T	Engine torque.
σ_{spk}	Advanced spark angle.
θ_0	Throttle angle due to driver request.
θ^*	Additive throttle angle due to compensation.
θ	Total throttle angle.
ξ_{ref}	Cam timing command.
ξ_{cam}	Cam timing

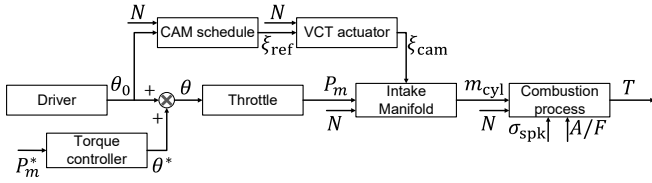


Fig. 1: Control schematic of a gasoline engine variable cam timing

fuel consumption. At a specified operating point, N and A/F are constants, while σ_{spk} is controlled at the maximum brake torque point which enables optimal ignition. In the combustion process block, the only variable parameter is m_{cyl} , which is determined by ξ_{cam} , P_m and N . The VCT creates large disturbance to m_{cyl} which leads to drivability issues [15]–[18]. To satisfy the torque demand when applying the optimal cam schedule, a torque controller is implemented to track the torque demand. An additive throttle angle θ^* is applied to compensate the effect of the scheduled cam timing on m_{cyl} , while θ^* is determined by the desired manifold pressure P_m^* [15].

B. Optimisation Structure and Constrains

The modelling and optimisation method is developed to calculate the optimal GDI engine parameter setting. The workflow of this method is described in Fig. 2. First, the experimental data is collected from an engine test bench. Then, the neuro-fuzzy algorithm is applied to train the engine model in a hierarchical global-local structure using the data. The global-local structure is capable of generating engine models at arbitrary operating points in real time. Inputs of the engine model, also known as manipulable variables in the optimisation phase, are selected as intake valve open timing (IVO) and exhaust valve close timing (EVC). The fuel rate is chosen as the model output. Finally, a parallel optimisation method is applied to calculate optimal manipulable variable sets. The cost value is updated in the optimisation algorithm during each evaluation, and the algorithm updates the input sets to minimise the model cost function.

During the optimisation, linear constraints are applied to

IVO and EVC so that the optimal cam positions do not exceed their physical limits. However, not all of engine parameter settings in the working range lead to a stable engine operation. The engine may achieve the lowest fuel consumption while working under an unstable condition. Therefore, discontinuous constraints are applied. The optimisation becomes a constrained optimisation problem:

$$\begin{aligned}
 &\text{minimise:} && \text{Fuel rate} = f_{\text{global-local}}(\text{IVO}, \text{EVC}) \\
 & && \text{IVO, EVC} \\
 &\text{subject to:} && \text{IVO}_{\min} \leq \text{IVO} \leq \text{IVO}_{\max} \\
 & && \text{EVC}_{\min} \leq \text{EVC} \leq \text{EVC}_{\max} \\
 & && (\text{IVO}, \text{EVC}) \notin \mathcal{D}_j
 \end{aligned} \tag{1}$$

where $f_{\text{global-local}}$ is the objective function required to be minimised, which holds a nonlinear relationship between inputs and output. \mathcal{D}_j is the discontinuous constraints set to avoid the engine mis-operation. To compensate torque variations caused by sweeping the cam timing, a torque controller is implemented to control the throttle angle.

III. GLOBAL-LOCAL ENGINE MODEL

A. Global-Local Model Architecture

An engine model is trained based on the experimental data collected from a 1.6-litre 4-cylinder GDI engine. To capture the highly nonlinear behaviour of the engine, the operating regime is splitted into several operating areas representing main physical effects. Local nonlinear models are developed at the centre of each operating area so that the specified area is covered. The model at an arbitrary operating point can be obtained by the interpolation among the nearest local nonlinear models [4], [12].

The global-local model holds a 2-layer model structure. On the global level, N and T are used for the preliminary partition as they have a dominating influence on the model output. The selection of operating points is determined by characteristics of the engine. In Fig. 3, the partitioning of a predefined operating region is described. There are 25 operating points, known as calibration points, selected to cover this area. The distance between calibration points is defined based on the trade-off between model accuracy and calibration effect.

The testing data at each calibration point are collected from the engine test bench for local nonlinear model training. A design-of-experiments methodology was applied to generate the training data. Afterwards, a Local Model Networks (LMN) using the Hierarchical Local Model Tree (HILO-MOT) algorithm is implemented at each calibration point to represent the engine behaviour.

The global-local model output at an arbitrary operating point $\hat{Y}(k)$ at time step k is expressed as a weighted aggregation of the local nonlinear model output:

$$\hat{Y}(k) = \sum_{i=1}^M \Psi_i(N, T) f_{N,T}(u_1(k), u_2(k) \dots) \tag{2}$$

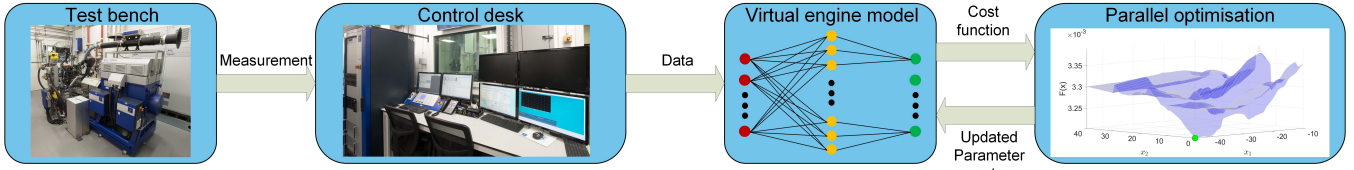


Fig. 2: Workflow of real-time modelling and optimisation

where M denotes the number of local nonlinear models, Ψ is the validity function value on the global level and $f_{N,T}$ is the local nonlinear model output at each calibration point.

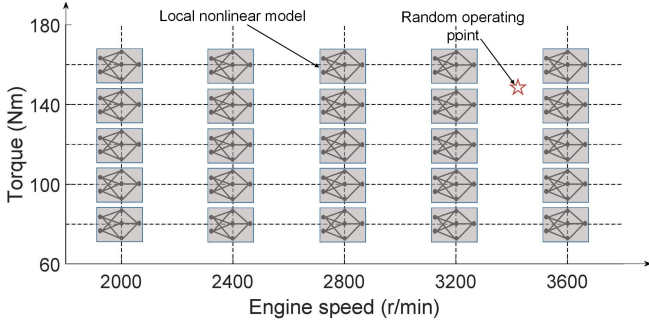


Fig. 3: Global-local model architecture

B. Local Nonlinear Model

The nonlinear behaviour of the engine at each calibration point is represented by the LMN. The HILOMOT algorithm is applied to train the LMN [19]. The HILOMOT performs axes-oblique partitioning of the input space, which guarantees the modelling accuracy with fewer neurons [20]. The simple model structure facilitates the online implementation of modelling and optimisation due to reduced computational burden. Moreover, because of the increasing number of engine control parameters, the number of LMN inputs is growing. Therefore, the HILOMOT algorithm is efficient in modelling systems of high dimensionality [21].

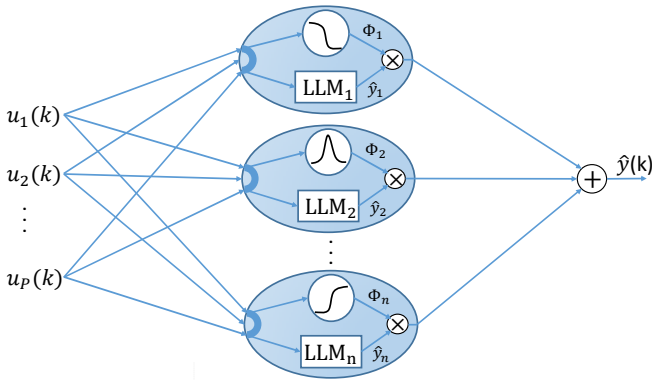


Fig. 4: Local model network structure

The structure of a LMN is illustrated in Fig. 4. A LMN is comprised of n neurons. For each neuron, a local linear model (LLM) is associated with a validity function Φ which

determines the weight of the neuron output. Φ allows smooth interpolation between different LLMs when inputs vary from one point to another. The validity function holds the following relationship:

$$\sum_{j=1}^n \Phi_j(u(k)) = 1. \quad (3)$$

In the HILOMOT algorithm, the validity function is given by a sigmoidal activation function which enables axes-oblique partitioning of the input space [20]:

$$\Phi(u(k)) = \frac{1}{1 + e^{\tau(v_0 + v^* u(k))}} \quad (4)$$

with

$$\tau = \frac{20}{\|v\| \cdot \|\Delta c\| \cdot \sigma}. \quad (5)$$

The transition smoothness between different LLMs is adjusted by Δc and σ which are the distance between centres of adjacent local models, and user-defined smoothness coefficient respectively. v_0 determines the split position and $v^* = [v_{i1}, v_{i2}, \dots, v_{ip}]^T$ determines the direction of the split. The estimated output \hat{y}_j of each LLM $_j$ is achieved by:

$$\hat{y}_j(k) = \omega_{j0} + \omega_{j1}u_1(k) + \omega_{j2}u_2(k) + \dots + \omega_{jp}u_p(k) \quad (6)$$

where ω_{jl} ($l = 1, 2, \dots, p$) denotes coefficients in each LLM. The output of the LMN is calculated as a weighted aggregation of \hat{y}_j ($j = 1, 2, \dots, n$):

$$\hat{y}(k) = \sum_{j=1}^n \Phi_j(k) \hat{y}_j(k) \quad (7)$$

where n is also the number of LLMs.

It is worthy to note that in the global-local architecture, the local nonlinear model represents the engine behaviour on the local layer. Therefore, IVO and EVC are LMN inputs, and fuel rate is the LMN output.

C. Global Partition

To obtain the models at arbitrary operating points, weights are assigned to each local nonlinear model according to (2). The validity function is chosen in the form of a Gaussian function. The advantage of the Gaussian function to split the input space is using the axes-orthogonal partitioning. This feature facilitates selection of calibration points and to determine the size of the validity area of each model.

On the global level of the model, N and T are chosen to perform partitioning. The weight is given by:

$$\Psi_i(u(k)) = \exp\left(-\frac{1}{2} \frac{(u(k) - \varepsilon_i)^2}{\eta^2}\right) \quad (8)$$

where ε denotes the coordinate of each calibration point and centre of Gaussian function, and η is the smoothness parameter. At the same time, the validity function holds the following relationship:

$$\sum_{i=1}^M \Psi_i(u(k)) = 1. \quad (9)$$

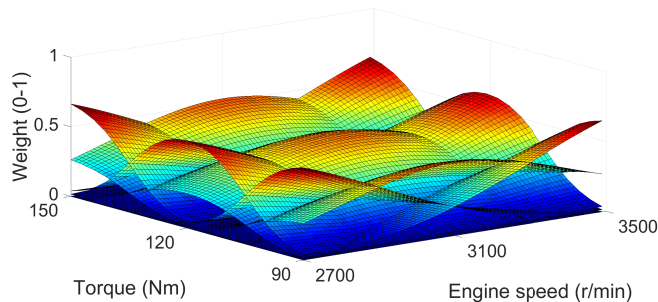


Fig. 5: Global partition result for a predefined regime

The partitioning result of a selected operating regime including 9 calibration points is shown in Fig. 5. There are 9 local nonlinear models in the map, while each of them dominates an engine operating area, as shown in red. Following the movement of the operating point, the weight of each local nonlinear model varies, which changes its contribution to the global output. The diagram shows a smooth transition between different validity areas.

D. Model validation

To validate the performance of the global-local model, a set of data consisting of steady states and transients is used. The performance of the established model is illustrated in Fig. 6. The engine operating points are presented in Fig. 6a. Fig. 6b shows variations of the cam timing. Validation results are depicted in Fig. 6c. The stationary behaviour is well captured by the global-local model. Moreover, the transient behaviour is modelled with high accuracy when N , T , IVO and EVC vary in a wide range.

IV. PARALLEL OPTIMISATION

The aim of the optimisation is to calculate the optimal engine parameters setting over arbitrary operating points or over any given operating point sequence. To further reduce the computational burden and to improve the robustness of heuristic optimisation methods, a parallel optimisation approach has been investigated [22]. This algorithm increases the probability of finding the global minimum and accelerates the algorithm in converging.

The parallel optimisation can be described from two aspects. The first is the optimisation over a given operating point sequence. The sequence can be divided into several

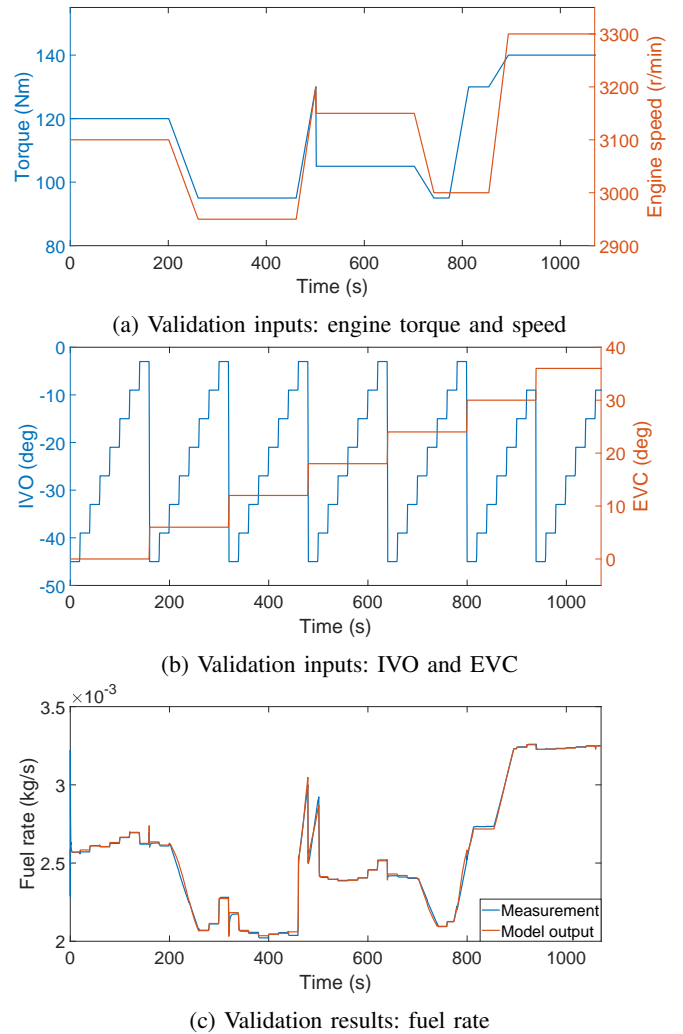


Fig. 6: Dynamical validation of the global-local model

TABLE II: Parameters in Algorithm. 1

Variable	Description
c_α	Children of optimiser α .
c_β	Children of optimiser β .
c^*	Child which provides minimum cost value.
f	Objective function.
$F(0)$	Initial function value.
F_α	Function value of optimiser α .
F_β	Function value of optimiser β .
$p(0)$	Initial population.
p_α	Population of optimiser α .
p_β	Population of optimiser β .
S	Solution.
V	Moving velocity of children.

discrete operating points, while optimisations are performed at all operating points simultaneously. The second is implementing the parallel optimisation algorithm itself. The parallel optimisation algorithm is outlined in Algorithm. 1, while the parameters are defined in Table. II. The optimi-

Algorithm 1: The Parallel Optimisation Algorithm

```
1 Initialization;
2 Set optimiser  $\alpha$  and  $\beta$ ;
3 Generate  $p(0)$ ;
4  $F(0) = f(p(0))$ ;
5 for  $t = 1 : g$  do
6   for  $s = 1:h$  do
7     Selection of  $p_\alpha^t(s)$ ;
8     Crossover;
9     Mutation of  $c_\alpha^t(s)$ ;
10     $F_\alpha^t(s) = f(c_\alpha^t(s))$ ;
11  end
12  for  $s = 1:M$  do
13     $c_\beta^t(s) = p_\beta^t(s) = p_\beta^t(s-1) + V^t(s-1)$ ;
14     $F_\beta^t(s) = f(c_\beta^t(s))$ ;
15    Update  $V^t(s)$ ;
16  end
17   $F_{\text{global}} = \min(F_\alpha^t(s), F_\beta^t(s))$ ,  $S_{\text{global}} = c^{t*}(s), \forall s$ ;
18  if termination criteria = true then
19    Algorithm output =  $[S_{\text{global}}, F_{\text{global}}]$ ;
20  else
21    Interchange between  $p(\alpha)_t$  and  $p(\beta)_t$  randomly;
22  end
23 end
```

sation algorithm uses multiple optimisers at the same time which increases the chance of finding the global optimum. In the case study, GA and Particle Swarm Optimisation (PSO) are implemented. After defining optimiser α and β , each optimiser will be evaluated iteratively to achieve the convergence. Due to the high nonlinearity of the engine model, the optimal solution given by heuristic optimisation algorithms may be trapped locally. During the optimisation, GA or PSO algorithm might converge to the local optimal area while another is heading to the global optimum. To address this issue, immigrants are introduced to heuristic algorithms' populations. After h iterations of both optimisers, the interchange operator will exchange populations among optimisers randomly. By introducing immigrants, the probability of finding the global optimum is increased. Moreover, population are diffused, which expands the search space. In addition, the parallel optimisation algorithm is effective for multi-objective optimisation problems. When the algorithm gives the completely non-dominated population, the interchange operator will replace this population with a dominated one, which enables the algorithm to continue searching the Pareto front.

V. SIMULATION RESULTS

To demonstrate the performance of the optimisation algorithm, two cases including randomly distributed operating points and an operating point sequence are optimised. Results are compared against the fuel consumption with the fixed cam settings.

A. Evaluation at Random Operating Points

In Fig. 7, the function surface is presented with the optimal solution at 142 Nm, 3200 r/min. The pink depicts the area that the global optimum is unlikely to locate in, while the blue indicates the area with the highest probability. As mentioned above, due to the high nonlinearity of the engine model, optimisation results may trap in the local optimal field, shown in Fig. 7. Blue dots are solutions from the PSO which are located in the local optimal area. The green dot is the global optimal point found by the parallel optimisation method. It shows that the parallel optimisation method is capable of moving the solution out of the local optimal region.

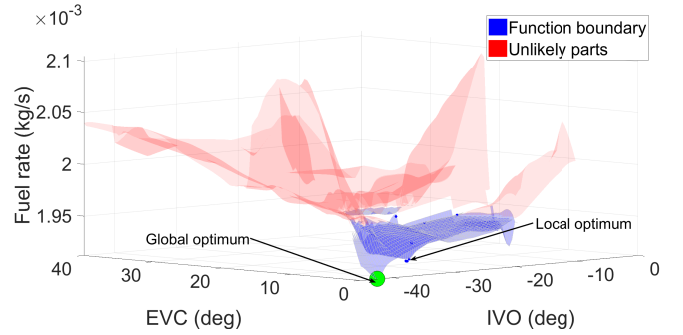


Fig. 7: Optimisation result at 93 Nm, 2800 r/min

The optimisation result at 142 Nm, 3200 r/min is elaborated in Fig. 8. It is observed that the global optimum is found rather than being trapped in the suboptimal field. The groove on the function surface which could lead to suboptimal results is ruled out during the optimisation. Therefore, particles will not move to that area.

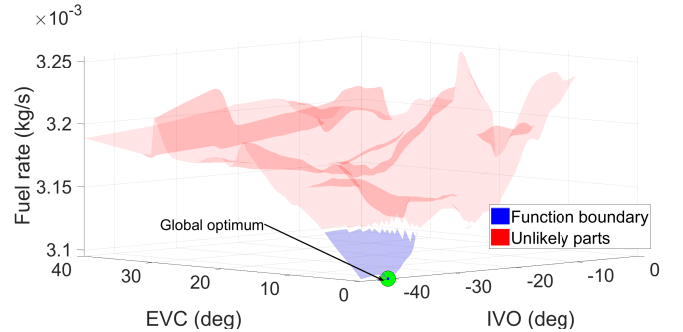


Fig. 8: Optimisation result at 142 Nm, 3200 r/min

From the parallel optimisation method, optimal solutions are generally promising and lead to lower engine fuel consumption under stable operating conditions. In addition, validation results indicate that the parallel optimisation method improves the robustness of heuristic optimisation algorithms. It is important to note that the optimisation result relies on the model accuracy. Engine model errors, such as extrapolation, can mislead the optimisation algorithm to search the wrong area on the function surface.

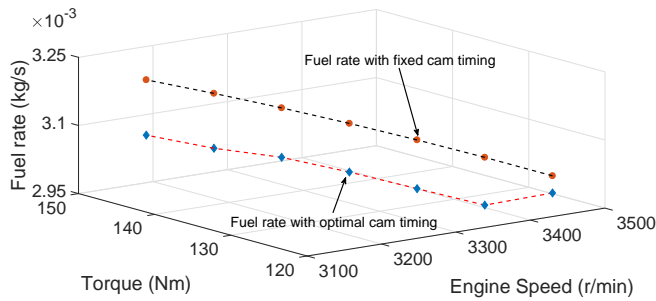


Fig. 9: Optimisation results of a random operating point sequence

B. Evaluation for an Operating Point Sequence

In practice, the engine will work continuously with changing operating points. Therefore, an optimisation over a given operating point sequence is also desired, shown in Fig. 9. For a given route, it can be treated as a number of discrete operating points. At each operating point, a model is generated by the global-local modelling method. Then, the optimisation of each model is performed simultaneously and a set of control signals for the given operating point sequence can be obtained. The control signal set is smoothed and sent to the engine control unit. The fuel rate with the optimal cam timing is measured and compared against the fuel rate with the fixed cam timing, as shown in Fig. 9. It indicates that the fuel consumption is reduced by around 5.5% at most of operating points.

VI. CONCLUSION

A global-local model-based optimisation method is developed for the engine optimisation. In generating engine models for an operating point sequence, both computing speed and accuracy are considered. The enhanced optimisation algorithm shows strong capability in moving the solution out of the local optimal area and leading to the global optimum. Simulation results of both real time modelling and parallel optimisation are presented. In the future, the optimisation task will be extended to a multi-objective optimisation problem including fuel efficiency and emissions. Additionally, the proposed global-local modelling architecture and optimisation algorithm will be implemented on dSAPCE processors for online application.

ACKNOWLEDGEMENT

The work was cofunded by the Digital Engineering and Test Centre (DETC), under a grant for the virtually connected hybrid vehicle. DETC is a unique joint industry-academic centre, also as an Advanced Propulsion Centre spoke. It develops and uses virtual engineering tools and techniques to accelerate the development, test and manufacture of automotive propulsion systems.

This work was also supported by the Engineering and Physical Sciences Research Council of U.K. under the EPSRC-UKRI Innovation Fellowship scheme (EP/S001956/1).

REFERENCES

- [1] R. Isermann and H. Sequenz, "Model-based development of combustion-engine control and optimal calibration for driving cycles: general procedure and application," *IFAC-PapersOnLine*, vol. 49, no. 11, pp. 633–640, 2016.
- [2] H. Ma, Z. Li, M. Tayarani, G. Lu, H. Xu, and X. Yao, "Computational Intelligence Non-model-based Calibration Approach for Internal Combustion Engines," *J. Dyn. Syst. Meas. Control*, vol. 140, pp. 1–9, 2017.
- [3] H. Ma, H. Xu, J. Wang, T. Schnier, B. Neaves, C. Tan, and Z. Wang, "Model-based Multi-objective Evolutionary Algorithm Optimization for HCCI Engines," *IEEE Trans. Veh. Technol.*, vol. 64, no. 9, pp. 4326–4331, 2015.
- [4] K. Fang, Z. Li, K. Ostrowski, A. T. Shenton, P. G. Dowell, and R. M. Sykes, "Optimal-Behavior-Based Dynamic Calibration of the Automotive Diesel Engine," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 3, pp. 979–991, 2016.
- [5] C. Atkinson and G. Mott, "Dynamic Model-Based Calibration Optimization: An Introduction and Application to Diesel Engines," *SAE Technical Paper*, pp. 1–12, 2005.
- [6] M. Deflorian, F. Klöpper, and J. Rückert, "Online dynamic black box modelling and adaptive experiment design in combustion engine calibration," *IFAC Proc. Vol.*, vol. 43, no. 7, pp. 703–708, 2010.
- [7] A. A. Malikopoulos, D. N. Assanis, and P. Y. Papalambros, "Real-Time, Self-Learning Optimization of Diesel Engine Calibration," in *ASME Intern. Combust. Engine Div. Fall Tech. Conf.*, 2007, pp. 537–545.
- [8] S. Zaglauer and U. Knoll, "Evolutionary Algorithms for the Automatic Calibration of Simulation Models for the Virtual Engine Application," in *IFAC Proceedings*, vol. 45, no. 2, 2012, pp. 177–181.
- [9] B. Berger, F. Rauscher, and B. Lohmann, "Analysing Gaussian processes for stationary black-box combustion engine modelling," *IFAC Proceedings*, vol. 44, no. 1, pp. 10633–10640, 2011.
- [10] Q. Zhou, Y. Zhang, Z. Li, J. Li, H. Xu, and O. Olatunbosun, "Cyber-Physical Energy-Saving Control for Hybrid Aircraft-Towing Tractor based on Online Swarm Intelligent Programming," *IEEE Trans. Ind. Informatics*, vol. 14, no. 9, pp. 4149–4158, 2017.
- [11] D. Zhao, C. Liu, R. Stobart, J. Deng, S. Member, E. Winward, and G. Dong, "An Explicit Model Predictive Control Framework for Turbocharged Diesel Engines," *IEEE Trans. Ind. Electron.*, vol. 61, no. 7, pp. 3540–3552, 2014.
- [12] V. Fachbereich, "Emission Modelling and Model-Based Optimisation of the Engine Control," Ph.D. dissertation, 2013.
- [13] F. Millo, P. Arya, and F. Mallamo, "Optimization of automotive diesel engine calibration using genetic algorithm techniques," *Energy*, vol. 158, pp. 807–819, 2018.
- [14] M. Hafner, M. Schüler, O. Nelles, and R. Isermann, "Fast neural networks for diesel engine control design," *Control Eng. Pract.*, vol. 8, no. 11, pp. 1211–1221, 2000.
- [15] M. Jankovic, F. Frischmuth, A. Stefanopoulou, and J. A. Cook, "Torque Management of Engines with Variable Cam Timing," *IEEE Control Syst.*, vol. 18, no. 5, pp. 34–42, 1998.
- [16] B. Wu, R. G. Prucka, Z. S. Filipi, D. M. Kramer, and G. L. Ohl, "Cam-phasing Optimization Using Artificial Neural Networks as Surrogate Models Fuel Consumption and NOx Emissions," *Engineering*, vol. 2006, no. 724, 2006.
- [17] A. Kakaee and M. Keshavarz, "Comparison the Sensitivity Analysis and Conjugate Gradient algorithms for Optimization of Opening and Closing Angles of Valves to Reduce Fuel Consumption in XU7 / L3 Engine," *Int. J. Automot. Eng.*, vol. 2, no. 3, pp. 143–155, 2012.
- [18] P. Sawant and S. Bari, "Effects of Variable Intake Valve Timings and Valve Lift on the Performance and Fuel Efficiency of an Internal Combustion Engine," *SAE Technical Paper*, pp. 1–11, 2018.
- [19] O. Nelles, "Axes-oblique partitioning strategies for local model networks," *Proc. Int. Symp. Intell. Control*, pp. 2378–2383, 2006.
- [20] T. Fischer, B. Hartmann, and O. Nelles, "Increasing the Performance of a Training algorithm for Local Model Networks," in *Proc. World Congr. Eng. Comput. Sci.*, 2012.
- [21] O. Nelles, *Nonlinear system identification: from classical approaches to neural networks and fuzzy models*, 2nd ed. Springer Science & Business Media, 2013.
- [22] M. Roberfröid, *Global Optimum Determination by Linking and Interchanging Kindred Evaluators*, 2005.