# Intelligent transient calibration of a dual-loop EGR diesel engine using chaos-enhanced accelerated particle swarm optimization algorithm

Zhang, Yunfan; Zhou, Quan; Li, Ziyang; Li, Ji; Xu, Hongming

*DOI:*
10.1177/0954407018776745



*Document Version*
Peer reviewed version

*Citation for published version (Harvard):*
Zhang, Y, Zhou, Q, Li, Z, Li, J & Xu, H 2018, 'Intelligent transient calibration of a dual-loop EGR diesel engine using chaos-enhanced accelerated particle swarm optimization algorithm', Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering. https://doi.org/10.1177/0954407018776745

# Intelligent Transient Calibration of a Dual-loop EGR Diesel Engine using Chaos-enhanced Accelerated Particle Swarm Optimization Algorithm

Yunfan Zhang, Quan Zhou, Ziyang Li, Ji Li and Hongming Xu*
Mechanical Engineering Department, School of Engineering
University of Birmingham, Birmingham, UK, B15 2TT
* Corresponding author: Professor Hongming Xu (h.m.xu@bham.ac.uk)

**Abstract:**

This paper proposes an intelligent transient calibration method for the air-path controller of a light-duty diesel engine. This method is developed based on the chaos-enhanced accelerated particle swarm optimization (CAPSO) algorithm. The target is to reduce the engine's fuel consumption during transient scenarios by optimizing the controller parameters. The advanced dual-loop EGR system is firstly introduced. Then it formulates the transient calibration process as a multiple-objective optimization problem with constraints. Different from steady state calibration, the proposed method designs a new cost-function to evaluate the controller's transient performance. The intelligent transient calibration module is programmed in MATLAB code. Interface between the calibration module and a physical engine plant is established via ETAS INCA. The optimization result of the proposal method is discussed by comparing it with the result of existing calibration methods. The engine performance with the calibrated controller is evaluated based on engine tests.

***Keywords*—** Diesel Engine, Multiple-objective Optimization, Particle Swarm Algorithm, Transient Calibration, Test Bench Validation

**Highlights:**

- The reliability and predictability of the engine model were evaluated
- The engine control parameters were optimized by the proposed calibration algorithm.
- The optimization results for different working conditions were validated via engine tests.
- The repeatability of the optimization results was improved.

## 1. Introduction

Diesel engines achieve relatively attractive fuel economy due to their higher compression ratio [1]. To further improve the engine's fuel economy while satisfying the progressively stricter emission legislation, more and more new technologies have been applied to the air-path of modern automotive diesel engines [2].

Among these technologies, EGR is very effective in suppressing the nitrogen oxides (NOx) emissions. There are three main types of EGR configuration being used: high-pressure EGR (HPEGR), low-pressure EGR (LPEGR) and dual-loop EGR (DLEGR). Among them, the HPEGR is the most widely used EGR configuration [3]. It has the advantages of simple structure and quick response in transient conditions. However, HPEGR also carries the disadvantage of a deteriorated VGT performance and difficulties in driving sufficient amount of EGR gas in high load conditions [4]. The exhaust gas is cleaner and the VGT could perform better with the help of LPEGR. However, LPEGR still suffers the disadvantages of the long pipeline and the tiny pressure difference across the loop. This leads to a slower dynamic response compared to single HPEGR during transient operations [5]. The DLEGR combines the advantages of both HPEGR and

LPEGR [6]. Thus, this configuration has been adopted recently in diesel engines for lower emissions, a quicker dynamic response and improved fuel economy.

However, the air-path of the diesel engine including the dual-loop EGR is a multiple-input, multiple-output (MIMO) system with strong non-linearity, coupling effect and delay. Engine controllers on commercial ECUs usually adopt the separate single-input-single-output (SISO) proportional–integral–differential (PID) method [7]. These characteristics make the controller calibration, especially the transient calibration process more complex and time consuming. The conventional engine calibration approach based on the design of experiments (DOE) method is facing more challenges [8][9]. Therefore, there is a need for intelligent transient calibration method for an engine's controllers. PID tuning methods have been proposed for this in recent decades, such as the frequency domain method and the Ziegler-Nichols (ZN) method [10][11][12]. However, the major limitation of these methods is the requirement for excessive knowledge on the system's frequency response or controller transfer functions; which makes them difficult to apply to the case of the diesel engine's air-path. To cope with these difficulties, evolutionary algorithms have been proposed. Automotive industries have developed commercial software such as AVL CAMEO and Matlab Model-based Calibration toolbox based on Genetic Algorithm. But these software are mainly designed to calibrate static conditions and require a considerable amount of data to build the empirical model [8]. Comparing with other metaheuristic algorithms such as genetic algorithm [13]and population-based algorithm [14], the swarm intelligence algorithms have the advantages of easy implementation, less computational effort and fast convergence speed [15]. Therefore, it is widely used in controller intelligent calibration cases[16][17][18][19][20][21]. In order to further improve the convergence speed of the algorithm, accelerated particle swam optimization (APSO) is created. The evidence shows that the APSO algorithm outperforms the PSO algorithm on multiple objective optimization issues [20][23]. However, these PSO-based algorithms may occasionally trap the particles in the local optimal position instead of a global optimal position. Thus recently, an APSO algorithm with chaotic mapping strategies (CAPSO) has emerged to enhance the result's repeatability and get rid of the 'local optimal' trap [24][25][26][27]. While it is still capable of solving optimization issues rapidly. The modification applies 'chaos' to the attraction parameters of the standard APSO dynamically. The nature of this mapping strategy is periodicity, stochastic properties and regularity of the chaos [28].

In this paper, an intelligent transient calibration method is developed for the air-path controller of a diesel engine. This work is organized as follows: in section 2, the architecture of the diesel engine air-path and the structure of embedded real-time simulation platform are introduced. The real-time engine model is also validated. Section 3 first demonstrates and explains the proposed CAPSO-based calibration algorithm. Then follows the experimental apparatus and procedure. Section 4 validates the proposed transient calibration method via several case studies and proceeds a Monte Carlo analysis to compare the CAPSO algorithm and conventional APSO algorithm. Eventually, the conclusions are summarised in section 5.

## 2. System Description and Engine Modelling

### *2.1* Architecture of the Diesel Engine's Air-path

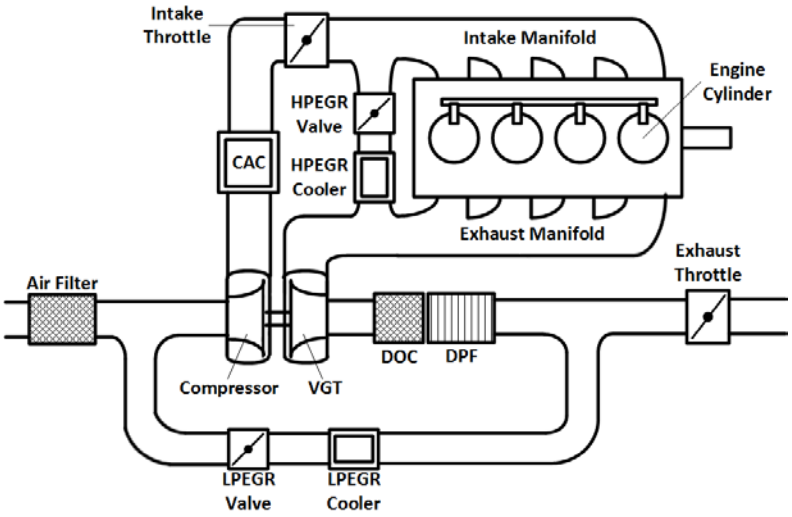



***Fig. 1*** *Architecture of a Diesel Engine's Air-path*

Fig.1 shows the diesel engine's air-path, which contains a HPEGR loop, a LPEGR loop, a VGT and a charge air cooler (CAC). In this case, various amount of EGR mass flow in each loop is manipulated by the EGR valve. The HPEGR loop is between the intake manifold and the exhaust manifold. The upstream of the LPEGR loop is located after the diesel oxidation catalyst (DOC) and the diesel particulate filter (DPF); while the downstream is linked to the air inlet tube before the compressor. Both of EGR loops are equipped with EGR coolers to further cool down the exhaust gas. The mixture between the filtered fresh air and the LPEGR gas occurs at the upstream of the compressor; then the mixed gas flows through the compressor, the CAC and the intake throttle into the intake manifold, where it combines with the HPEGR gas. The boost pressure is manipulated by the VGT rack position. In this paper, both the intake throttle and the exhaust throttle keep constantly opened.

### *2.2* Structure of the Embedded Model

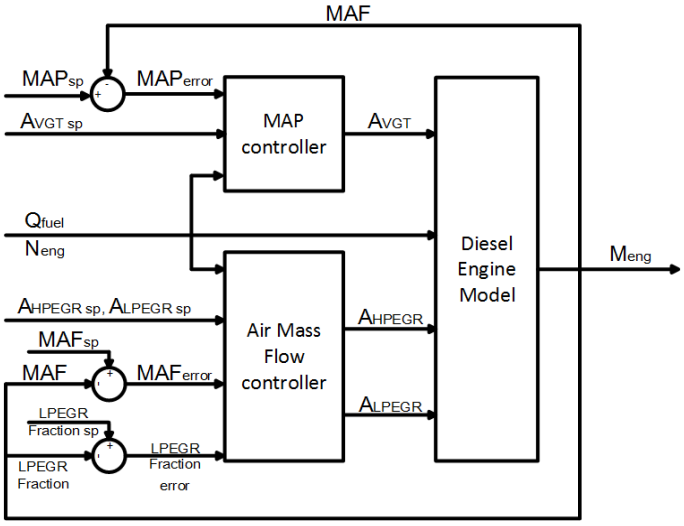



***Fig. 2*** *Structure of the embedded engine model*

The embedded model (shown in fig.2) consists of two parts which are an air-path controller and a diesel engine model.

The air-path controller adopts the conventional PI control strategy. The control strategy for the model is identical to the one used in the engine ECU. The air-path controller is divided into two sections. The first part is the boost pressure control block. The inputs are the MAP error, VGT rack position set point, engine speed and mass of fuel injection. The MAP error is calculated as the difference between the MAP setpoint and actual MAP:

$$e_{MAP\,(t)} = MAP_{sp} - MAP_{actual} \quad (1)$$

Based on this error signal, the MAP controller is implemented to manipulate the VGT rack position. The final output of the MAP controller involves both the feedback PI strategy and the feed-forward setpoint of the VGT rack position.

$$VGT_{(t)} = K_p \cdot e_{MAP\,(t)} + K_i \cdot \int_0^t e_{MAP\,(t)} \cdot dt + A_{VGT_{sp}\,(t)} \quad (2)$$

In terms of the air mass flow controller, it is accomplished by means of combining two PI controllers together. One PI controller focuses on the MAF trajectory tracking, the other takes charge of the LPEGR fraction. The required inputs are the MAF error, LPEGR fraction error, HPEGR valve position set point, LPEGR valve position set point, engine speed and mass of fuel injection. The controller outputs are the actual HPEGR and LPEGR valve positions. The detailed structure is introduced by the following equations

The MAF error and the LPEGR fraction error are calculated as follows:

$$\begin{gathered} e_{MAF\,(t)} = MAF_{sp} - MAF_{actual} \\ e_{LPEGR\,fraction\,(t)} = LPEGR\,fraction_{sp} - LPEGR\,fraction_{actual} \end{gathered} \quad (3)$$

In this case, based on the report about the air-path controller [29], the MAF controller monitors the MAF value via a 'virtual EGR valve' ($EGR_{(t)}$). The LPEGR fraction controller actuates over a splitting factor ($\kappa$) that divides the virtual EGR valve signal into two different signals for both HP and LP EGR valves to keep the desired LPEGR fraction following the desire value. The equations are shown as following:

$$EGR_{(t)} = K_p \cdot e_{MAF\,(t)} + K_i \cdot \int_0^t e_{MAF\,(t)} \cdot dt + EGR_{sp\,(t)} \quad (4)$$

$$\kappa_{(t)} = K_p \cdot e_{LPEGR\,fraction\,(t)} + K_i \cdot \int_0^t e_{LPEGR\,fraction\,(t)} \cdot dt + LPEGR_{sp\,(t)} \quad (5)$$

$$EGR_{sp\,(t)} = HPEGR_{sp\,(t)} + LPEGR_{sp\,(t)} \quad (6)$$

$$HPEGR_{(t)} = EGR_{(t)} \cdot \kappa_{(t)} \quad (7)$$

$$LPEGR_{(t)} = EGR_{(t)} \cdot (1 - \kappa_{(t)}) \quad (8)$$

The control strategy of the diesel engine's air path is developed in an Open-ECU, it cannot be compared with strategy in the commercial ECU. But the feed-forward control still exists in the controller. The feed-forward values of the actuator positions are included in the controller design in question 1. In this case, the feed-forward setpoint is acquired by the engine working condition (engine speed and mass of fuel consumption). They are set to the same values of the actuator positions under steady state conditions to reduce the controller complexity. Focusing on this specific problem, the proposed optimization method mainly focuses on refining the PI values.

The engine model is designed with five inputs and five outputs. The inputs are the engine speed, mass of fuel injection, VGT rack position, HPEGR valve position and LPEGR valve position. The outputs include the engine MAP, MAF, HPEGR mass flow, LPEGR mass flow and the engine torque. The detailed modelling method is introduced in the author's previous work [30]. The validation process is presented in the following sections.

### *2.3* Validation of the Engine Model

In the validation process, the world harmonized light vehicle test procedure (WLTP) is conducted on both the real engine and the model. The comparison mainly focuses on the trajectories of the controller objects (MAP, MAF, LPEGR fraction) and the engine torque, which is shown in Fig.3. The model accuracy is evaluated by calculating the fitting rate and the dynamic error, the equation are defined as [31]:

$$fit(i) = \left[1 - \frac{\|y_{meas} - y_{model}\|}{\|y_{meas} - \bar{y}_{meas}\|}\right] \times 100\% \quad (2)$$

$$y_{error}(i) = \frac{y_{meas}(i) - y_{model}(i)}{1/N \sum_{i=1}^{i=n} y_{meas}(i)} \times 100\% \quad (3)$$

where $y_{meas}$ is the measured output from the engine; $y_{model}$ is the simulated output from the model; $\bar{y}_{meas}$ is the mean value of the data sequence from the engine and '|| ||' represents the Euclidean distance.

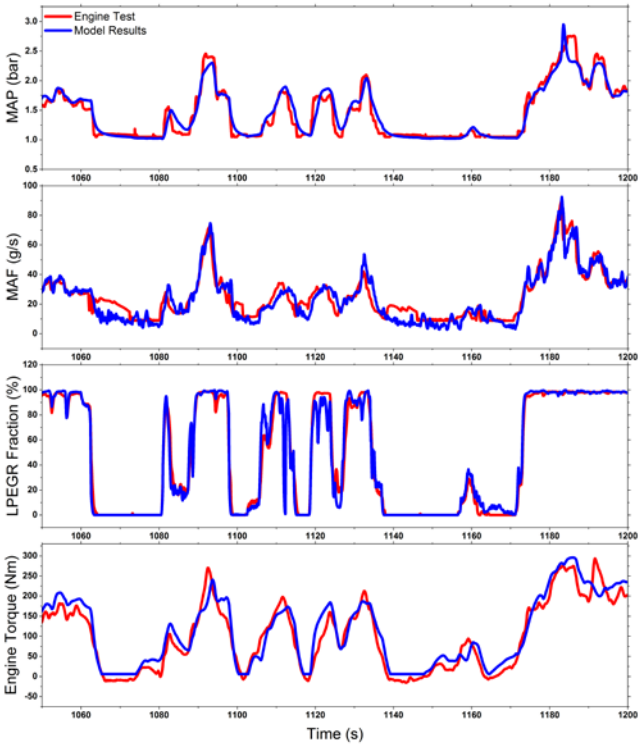



***Fig. 3*** *Comparison between the Simulation Results and Engine Test Results of WLTP (1070s-1200s); MAP; MAF; LPEGR Fraction and torque*

Even though the presented figure includes only part of the test cycle, the model fitting rate and the dynamic error are calculated based on the overall data. The simulation results show a fair agreement with the experimental data. The fitting rate of the engine MAP, MAF, LPEGR fraction and engine torque are 83.74%, 85.41%, 80.86% and 82.53% respectively (shown in Table 1). In terms of the dynamic error, it falls in the range of 10%, which is acceptable for engine modelling. Besides, this model is also capable of capturing most peaks and troughs of the controller objects during transient operations. It reflects the air dynamics characteristics properly. It could be concluded that this model is qualified of acting as the platform for the development of intelligent transient calibration algorithm.

TABLE 1
Model Fitting Rate and Dynamic Error

| SPECIFICATION | FITTING RATE | DYNAMIC ERROR |
|---|---|---|
| MAP | 83.74% | 5.57% |
| MAF | 85.41% | 9.49% |
| LPEGR Fraction | 80.86% | 8.22% |
| Torque | 82.53% | 7.84% |

# 3. Methodology

## 3.1. Multiple-objective Optimization Issue

The controller contains three manipulation objects which are the engine MAP, MAF and LPEGR fraction. The calibration of the air-path controller is formulated into a multi-object optimization problem with constraints, which allows the proposed CAPSO algorithm to be applied. The calibration parameters selected in this paper are the Kp and Ki values ($K_{p\_MAP}, K_{i\_MAP}, K_{p\_MAF}, K_{i\_MAF}, K_{p\_LPF}\ and\ K_{i\_LPF}$). At the beginning of the algorithm design, the search area and the cost-function must be designed properly.

The multi-objective optimization process is similar to treasure hunting. The search area regulates boundaries of the calibration parameters. A wider searching range contributes to achieve more optimal results, but it also sacrifices the time consumption. It should be noticed that a full calibration of the diesel engine's air-path controller may involve various settings.

The cost function is the criterion for the optimization algorithm, like the cost function in model predictive control. The objective function of the parameter tuning is formulated as the integrals of square error (ISE) for this case [32].

$$ISE_i = \int_0^{t_s} e_i^2(t)\, dt \tag{4}$$

where $e_i(t) = r_i(t) - y(t)$ denotes the error value of each controller object; which represents the deviation between the desired value and the actual value of the manipulation object. The ISE value for each target is named as $ISE_{MAP}$, $ISE_{MAP}$ and $ISE_{LPF}$.

So the cost function of the controller behaviour is the overall ISE value of the three control targets using the weighted sum method [33]. The final format of the cost function is:

$$J_1 = W_1 \cdot \frac{ISE_{MAP}}{CF_1^*} + W_2 \cdot \frac{ISE_{MAF}}{CF_2^*} + W_3 \cdot \frac{ISE_{LPEGR\ fraction}}{CF_3^*} \tag{5}$$

$$W_1 + W_2 + W_3 = 1 \quad W_1, W_2, W_3 \in (0,1) \tag{6}$$

where $W_1$ is set to 0.5, $W_2$ and $W_3$ are set to 0.3 and 0.2 to achieve a balanced controller calibration. In this case, the three objects are equally important for the optimization problem in this case. A higher weight value indicates means it has more significant impact to the value of total cost function. But the MAP, MAF and LPEGR fraction also have affection to each other. One deteriorated optimization object will also affect the other optimization objects due to coupling effects in the air-path. For diesel engine's air-path, the MAP value is the top-priority, which regulates the total amount of gas pumped into cylinders. So the MAP weight should be relatively larger. $CF_1$, $CF_2$ and $CF_3$ are correlation factors to assure the ISE value for each control object falls in the same scale of range.

The stability of the optimal controller is also considered in the design process of the optimization cost function. Based on the literature review on this issue, the cost function is designed in the format of ISE (integrals of square error). When the system outputs are unstable, the instability is

reflected by the cost-function value (The cost function value would increase) as the ISE calculates the integral of the squared output error. Besides, the test sequences involve various engine speed and load, which are designed to test the optimal controller's stability under different working conditions (shown in the section of result & discussion).

### 3.2. Structure of the Transient Calibration Algorithm

The chaos-enhanced accelerated particle swarm optimization (CAPSO) algorithm is modified based on the conventional APSO algorithm; which is also a particle-based evolutionary optimization method. It is inspired by animal swarms in nature. Fig.4 shows the workflow of the optimization algorithm in the case of intelligent calibration of a diesel engine's air-path controller.

The CAPSO algorithm, shown in fig.4, consists of three parts. The first part contains the settings of the initial condition, which involve the number of particles in each swarm, number of iterations and the boundary conditions to regulate the search area. The initial particles for the iteration process are generated randomly from this information provided in this block. After generating the initial particles, the iteration process begins. The cost function for each particle could be acquired by the co-simulation with the engine model. They are all stored to retrieve the current local optimum results. Then based on the agents' current best results and optimal position of the initial particles, the positions of the particles are updated in each iteration. The position updates are based on three elements: the particle's current position, the best position in the swarm and a random factor. This is the end of one iteration. In next iteration, this process is repeated. The iteration stops when the pre-set boundaries are achieved and the final optimal results could be presented by the solution of the last iteration.

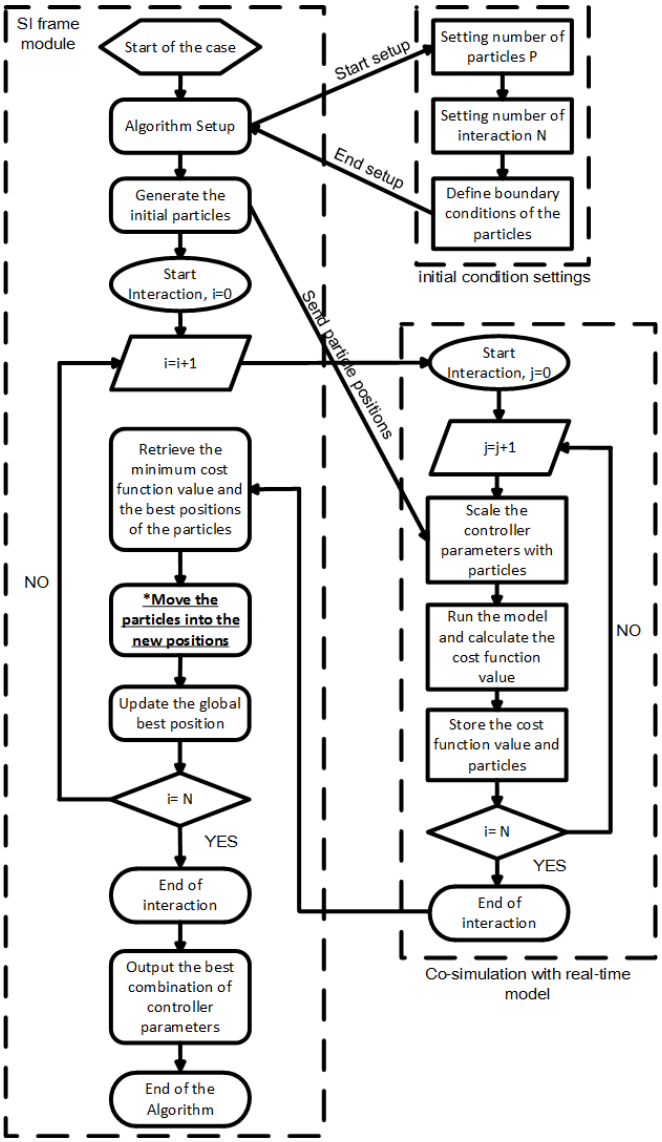



***Fig. 4*** Workflow of the CAPSO Algorithm in the calibration of a Diesel Engine's Air-path Controller

It is shown in section 3.1, the particles in the swarm are defined as:

$$x^{(i,j)} = [Kp_{map}{}^{(i,j)}, Ki_{map}{}^{(i,j)}, Kp_{maf}{}^{(i,j)}, Ki_{maf}{}^{(i,j)}, Kp_{LPEGR\ fraction}{}^{(i,j)}, Ki_{LPEGR\ fraction}{}^{(i,j)}] \quad (7)$$

where i (1, 2, 3…N) is the index of the interactions, for a CAPSO algorithm that has N iterations; j (1, 2, 3…P) is the index of the particles in each swarm.

The most important part in the CAPSO algorithm is the evolution of the particles. The governing equations are shown:

$$x^{(i+1,j)} = x^{(i,j)} + \beta\left(g^{(i,*)} - x^{(i,j)}\right) + \alpha^{(i)} \cdot r^{(i,j)} \quad (8)$$

$$\alpha^{(i)} = \alpha^{(0)} \cdot \gamma^{(i)} \quad (9)$$

where $g^{(i,*)}$ is the best position at ith interaction; $\beta$ is the attraction parameter of the CAPSO algorithm; $\alpha$ represents the convergence parameter that could be updated at each iteration. It is mentioned in [19] that the setting range of $\alpha^{(0)}$ falls between 0.5 and 1; $\gamma$ falls between 0 and 1. In this case, $\alpha^{(0)} = 0.9$ and $\gamma = 0.8$ are chosen for the controller intelligent tuning; $r^{(i,j)}$ is a random movement of the particle inside its searching area. The value of $\beta$ affects the convergence speed of the CAPSO algorithm and it is between 0 and 1. When $\beta = 1$, the convergence speed will

remain stationary and move towards to the current best location directly. If $\beta = 0$, the convergence speed becomes too low. The conventional APSO algorithm keeps $\beta$ at 0.5 as a constant value. [34] Even though this setting could work effectively, the optimization results still change slightly. Therefore, a variable setting of $\beta$ value in each iteration is needed to help the particles escape from the local best results. Thus, chaotic mapping strategies have been adopted. The chaotic mapping strategy used here is the logistic mapping strategy [35]. The equation is shown below:

$$\beta^{(i+1)} = a \cdot \beta^{(i)} \cdot (1 - \beta^{(i)}) \tag{10}$$

where $a$ is set to 0.4 and the initial value of $\beta$ equals 0.7 for this case. The author's previous work shows the biggest advantage of this strategy is the high dispersion of the randomly generated number [36].

Both alpha and beta values are the parameters to update the particle's position. Beta is a randomly generated number between 0 and 1, which will affect the convergence speed between each particle and the local best result. Alpha is the coefficient which determines the random move of each particle. The optimization problem in this case is solved offline, which means there is no strict limitation on the computational time. The alpha value will decrease gradually to make sure the results are converged. The settings of alpha and beta are not relevant to the engine load conditions. It will only affect the algorithm's performance.

### 3.3. Transient Calibration System

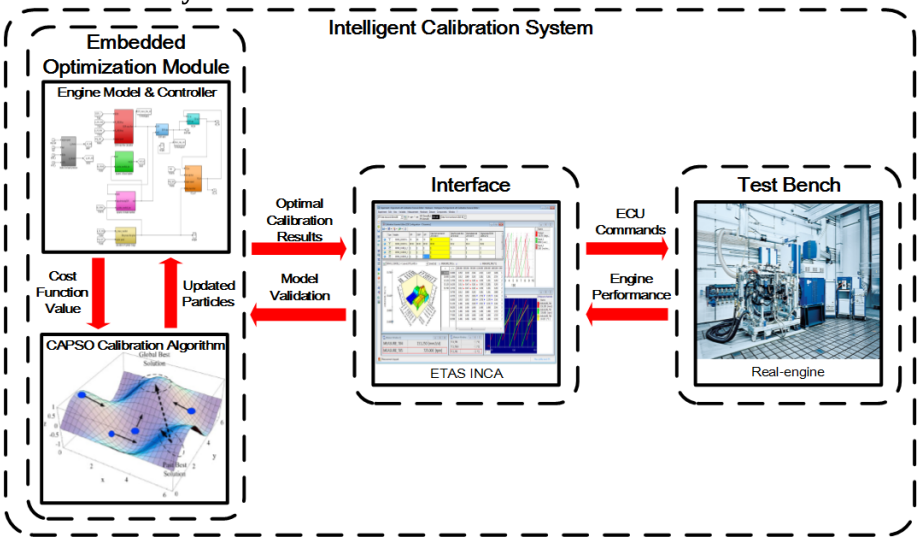



***Fig. 5*** *Interface of the real-time model and the CAPSO calibration algorithm*

Fig.5 shows the structure of the transient calibration system. This system contains three parts which are the embedded optimization module, ECU interface and the engine test bench. The embedded optimization module is formulated by the engine model and the transient calibration algorithm (introduced in section II C and section III A). The calculated results of the calibration algorithm would be transmitted into the ECU interface. The ECU interface plays the role to replace the ECU calibration by the results from the embedded optimization module. The model-based optimization results could be validated by the tests on a real engine and the test bench data is helpful for developers to further improve the accuracy of the engine model.

### 3.4. Experimental Apparatus and Procedure

The engine validation process is performed on a 4-cylinder turbocharged diesel engine equipped with common-rail direct injection system. More information of the engine specification is summarized in Table 2.

**TABLE 2**

Engine Specifications

| SPECIFICATION | |
|---|---|
| Engine | Four-cylinder Diesel Engine |
| Bore | 83mm |
| Stroke | 92mm |
| Displacement Volume | 1993ml |
| Maximum Torque | 430 Nm (1750-4500rpm) |
| Maximum Power | 132 Kw (4000rpm) |
| Compression Ratio | 15.5 |

The engine fuel consumption is measured by an AVL fuel meter (AVL 735). The fuel temperature is maintained at ambient condition at 25 ℃ by the conditioning system (AVL 753C). In each test sequence, the engine fully warmed up and the intake air temperature is also maintained at ambient condition. The fuel injection system follows the same calibration in each test. The whole engine test bench is operated and monitored by PUMA system from AVL. The PUMA system sends the demand dyno speed and pedal position to the dynamometer while the recorded test bench signals (such as engine torque) are sent back to PUMA. The ECU signals are collected by the ETAS INCA through CAN interface. The communication between AVL PUMA and ETAS INCA is established by ASAM3 protocol through Ethernet cable. The recorded data includes the engine speed, mass of fuel injection and all measurable parameters in the engine air-path. The detailed schematic diagram of the engine test bench is shown in Fig.6.

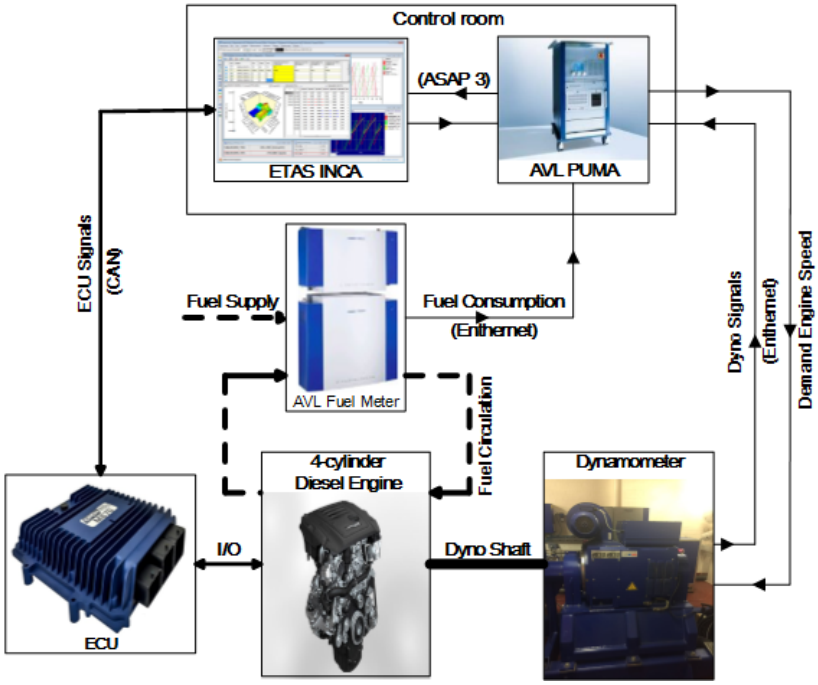
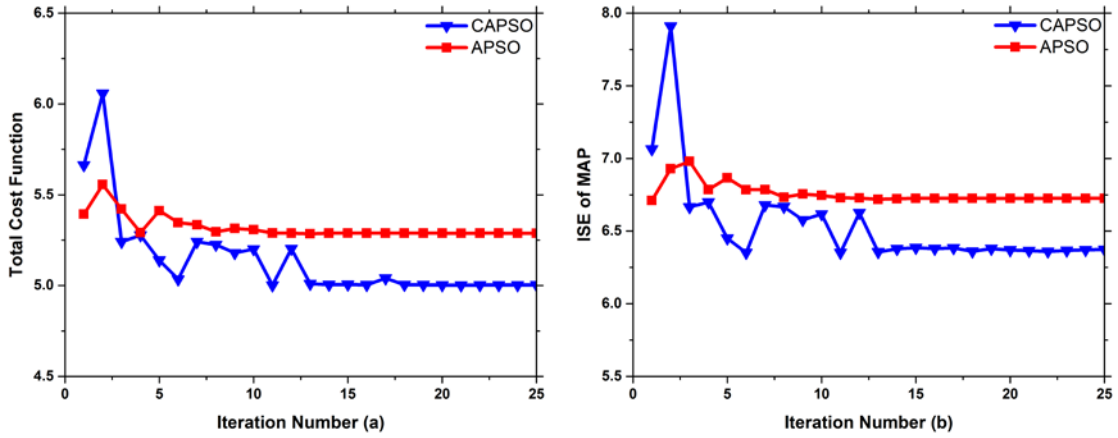



*Fig. 6 Schematic Diagram of the Engine Test Bench*

# 4. Results and Discussion

## 4.1. Comparison between CAPSO and Conventional APSO Algorithm

In this paper, the proposed CAPSO calibration algorithm is upgraded from conventional APSO algorithm. The evaluation should consider not only case studies of the engine transient scenarios but also the algorithm itself. Hence, the comparison on convergence speed, Monte Carlo analysis and the reputation evaluation between the two algorithms based on one case study are presented in the following paragraphs.



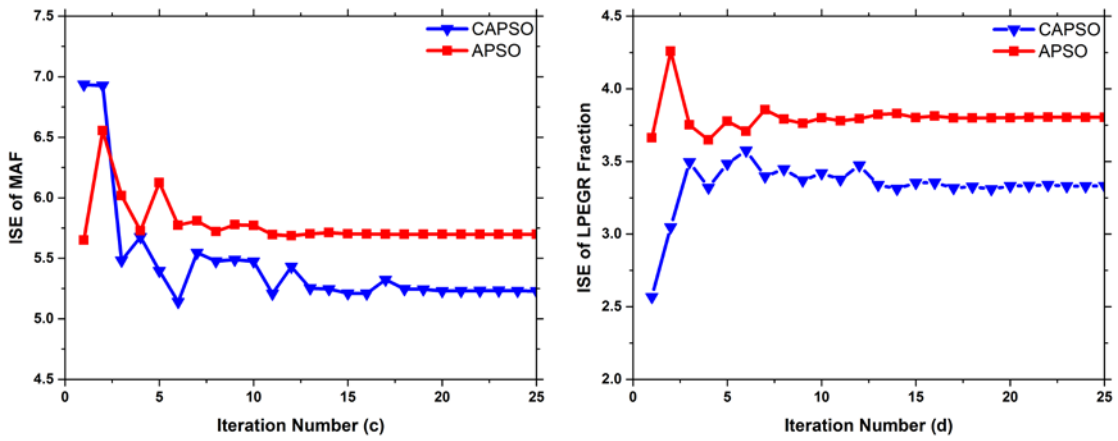



***Fig. 7*** *Trajectory of the Total Cost Function Value (a); ISE of MAP (b); ISE of MAF (c); ISE of LPEGR Fraction (d); Using the CAPSO Algorithm and conventional APSO Algorithm*

Fig.7 presents the trajectory of the intelligent tuning results using the standard APSO and CAPSO algorithm. The settings of the algorithms are introduced in section 3. It can be seen that both the CAPSO and APSO algorithms converge to stable controller parameters within 25 iterations. The result in the figures above indicate that the CAPSO algorithm could achieve smaller values of both the total cost function and the sub cost function, which is able to obtain better controller performance. However, it should be noticed that due to the chaotic mapping strategy in the CAPSO algorithm, the trajectories have greater randomness than the standard APSO algorithm. This characteristic ensures the CAPSO has a wider searching range in each iteration and avoids trapping in the best local results. No matter whether the conventional APSO algorithm or the CAPSO algorithm is used, they all involve random number generation. This means that the performance of the two algorithms could not be fully revealed through one single attempt. Thus, a repeatability test and statistical analysis should be taken into consideration.

For the above purposes, a Monte Carlo analysis is carried out to evaluate the performance of the CAPSO and APSO algorithms. Each algorithm is operated 20 times with uniformly distributed random initial values. Table 3 shows the mean value and the standard deviation of the cost function using the CAPSO and APSO algorithms.

**TABLE *3***

Mean Value and Standard Deviation of the Cost Function Values using CAPSO and Conventional APSO

| MEAN VALUE | CAPSO | APSO | STANDARD DEVIATION | CAPSO | APSO |
|---|---|---|---|---|---|
| Total Cost Function | 5.163 | 5.387 | Total Cost Function | 0.139 | 0.213 |
| MAP | 6.479 | 6.813 | MAP | 0.128 | 0.241 |
| MAF | 5.753 | 5.748 | MAF | 0.554 | 0.568 |
| LPEGR Fraction | 3.173 | 3.481 | LPEGR Fraction | 0.296 | 0.341 |

Table 3 shows that the both the total cost function values and the cost function's standard deviation using the CAPSO algorithm is smaller than that using the standard APSO algorithm. The total cost function through the CAPSO tuning process is 4.1% less than that through the

APSO tuning. The average cost function values of the engine MAP and LPEGR fraction are also decreased by 4.9% and 8.8%; which means more smooth trajectories of the controller objects could be acquired. Due to the existence of the random factors in both types of the algorithms, it is essential to consider the deviation of the optimization results. A smaller standard deviation value indicates that results could fall into a tiny range, which is helpful to increase the possibility of obtaining the optimal controller parameters in one time. The standard deviation of the CAPSO algorithm is only 0.139; while the APSO algorithm would enlarge it to 0.213. This is mainly caused by the significant difference on the MAP and LPEGR fraction cost function.

### 4.2. Investigation of Weight Tuning on the Calibration Results

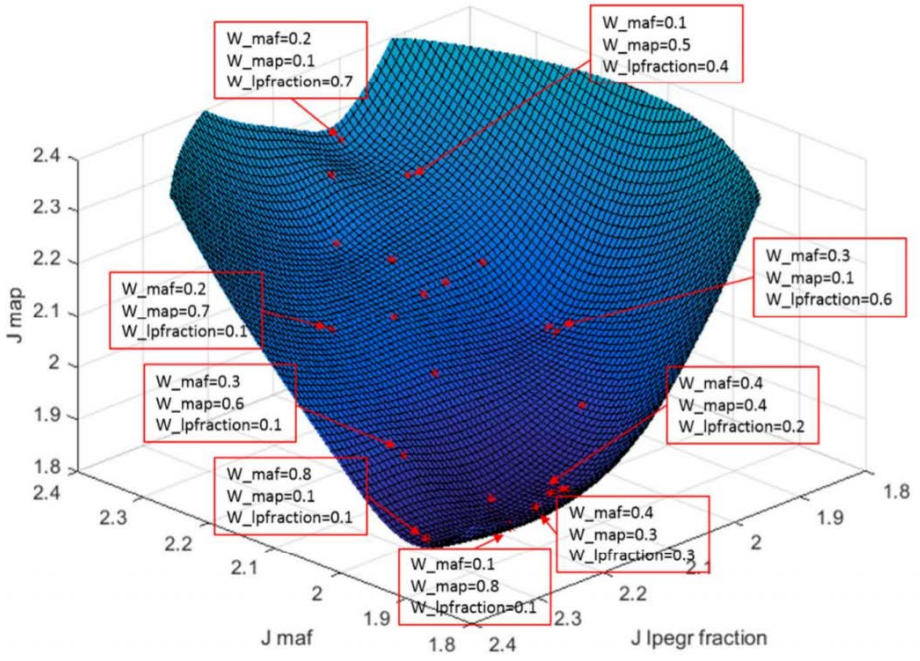



***Fig. 8*** *Pareto Frontier for Different Weight Value Settings*

Fig.8 shows the Pareto front of the optimization results as the weighting is adjusted in a specific range. The red dots in the figure are the collected test data and the blue part shows the generated surface. One phenomenon is that if one weight value is set to some extreme conditions (extremely large or extreme small values), the cost function values for the other two optimization objects also get larger as it compares with other weight settings. For example, when the MAP and MAF trajectories are deteriorated, the cost function value and trajectory tracking performance are worse. Another phenomenon is the coupling effect inside the diesel engine's air-path, there is a trade-off between three optimization objects. The characteristic of the slow dynamic response in the LPEGR loop is also reflected in the figure. Regardless of the weight settings, the cost function value of the LPEGR fraction falls in a smaller range when compares with other optimization objects. The figure indicates the diesel engine's air path is more sensitive to the MAF and MAP weight settings. In this case, when the MAF and MAP weights are set between 0.3 and 0.4, the total cost function value achieves its minimum level.

### 4.3. Test Bench Validation

To validate the results of the proposed CAPSO algorithm on the test bench, a transient sequence is selected as an example. The engine speed is maintained at constant values and the mass of fuel injection follows stepped changes. This kind of scenario is suitable for engines on the test bench and could be seen in numerous studies on engines' transient behaviors.

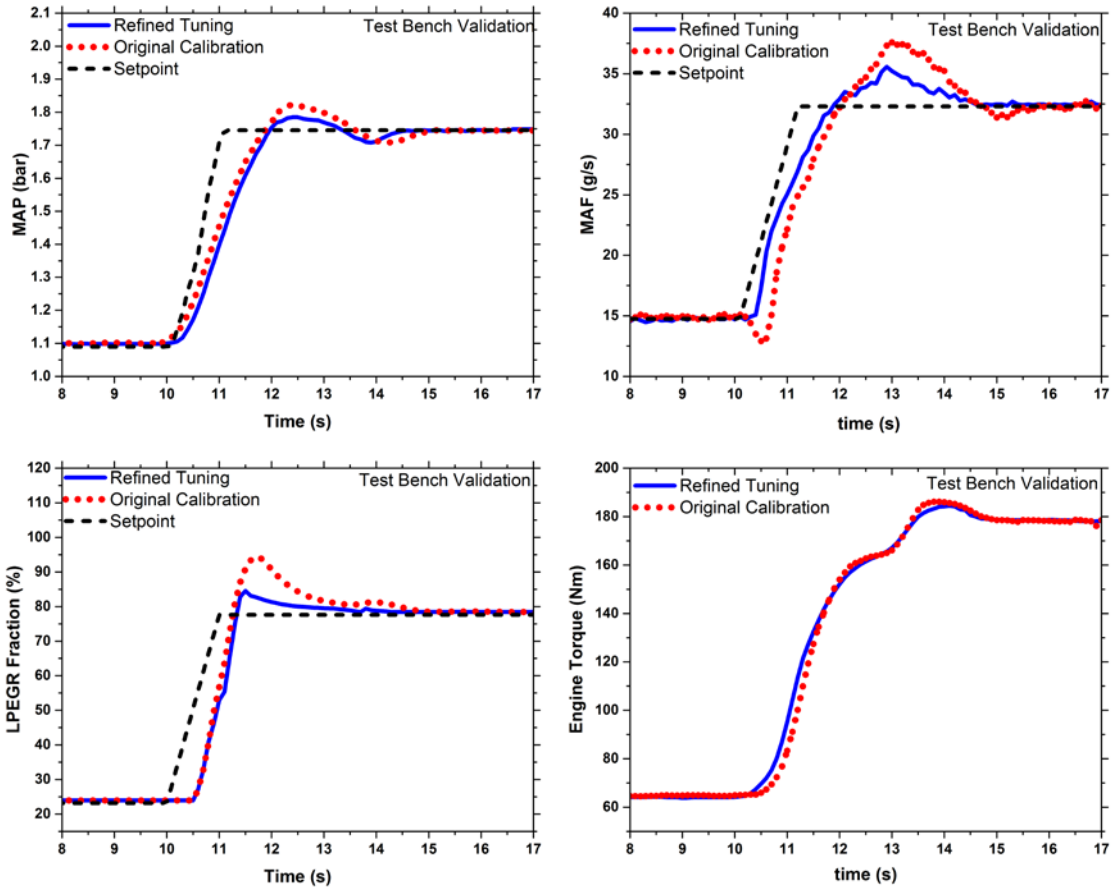



***Fig. 9*** *Trajectories of Engine Parameters using CAPSO Calibration Algorithm by Engine Tests*

Fig.9 shows the test results obtained from the engine test bench. The trajectories selected are the three controller objects and the engine torque. The engine working condition is set at 1500 rpm with a stepped change of fuel injection from 15 to 30 mg/stroke. It can be seen from the first three figures that the overshoot and settling time of the controller objects are successfully reduced. It is known that the key elements for NOx formation are high combustion temperature and the overdosed amount of air [37]. The sudden increased fuel injection at 10s would obviously raise the temperature inside the cylinder. The higher overshoot of the MAF trajectory at this moment using the original calibration also provides an excessive amount of air for NOx formation; so it is foreseeable that the NOx spike at these moments would be larger than that using the transient intelligent calibration. The MAF value is also closely linked to diesel engines' particulate emissions because the conditions for particulate formation are an over-rich air-fuel mixture and low combustion temperature [38]. The larger overshoot of the MAF trajectory at 12.5s in Fig.8 provides a low level of MAF value and it contributes to the PM increment. Besides, a 0.78% reduction of the accumulated BSFC is still observed via the test bench validation. According to the findings in the author's previous work, the proper controlled VGT and HPEGR play a dominating role [30]. The engine pumping mean effective pressure (PMEP) is successfully reduced which increases the brake torque generation of the engine.

To demonstrate the stability of the optimal control results, the trajectories of the actuators are shown in Fig.10. It can be seen from the figure that the trajectories of the HPEGR valve, LPEGR valve and VGT are all stabilized using the CAPSO calibration algorithm.

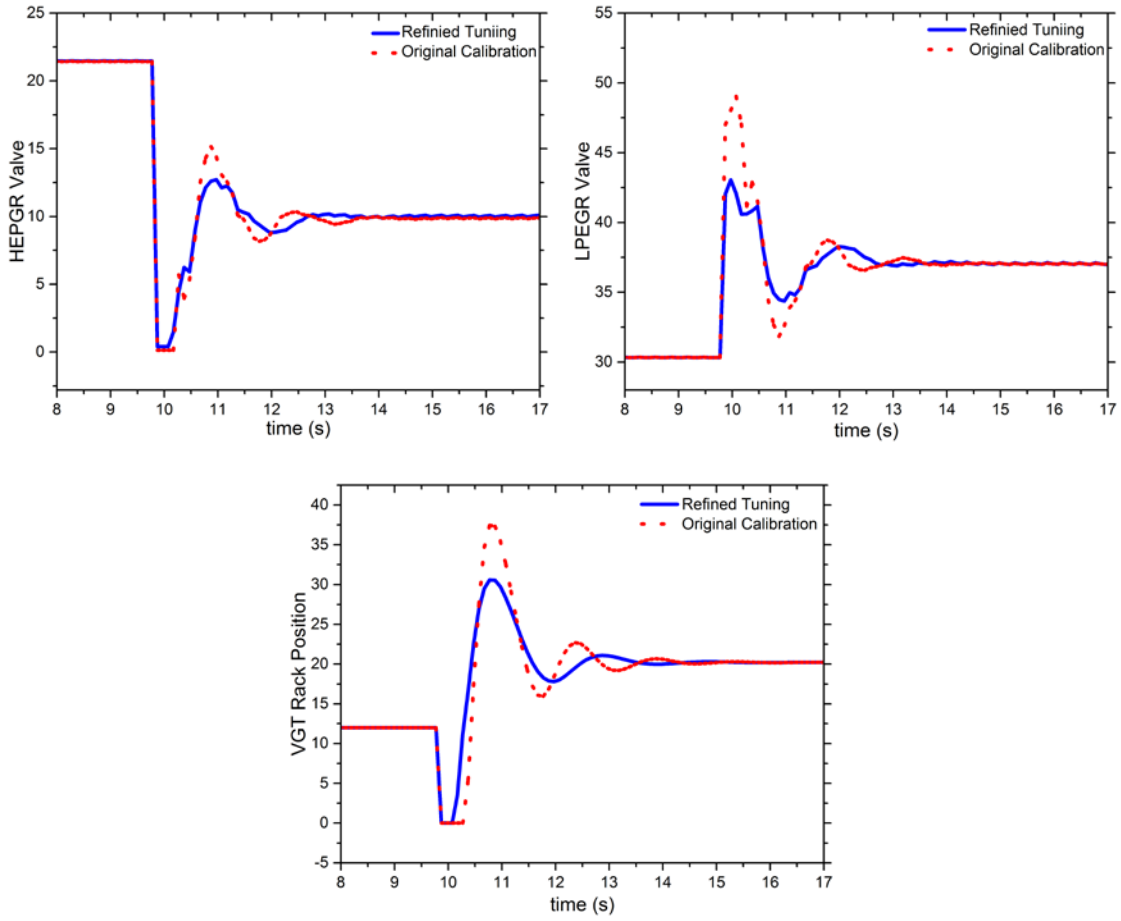



*Fig. 10 Trajectories of engine parameters using CAPSO calibration algorithm*

### 4.4. Case Study of Engine Transient Calibration

The capability of proposed CAPSO calibration algorithm has been proved by the figures shown in the previous paragraphs. It should be noticed that several other factors such as the actuators' delay would also affect the calibration results. Only one test sequence is not adequate. To test the transient calibration method in a wider range of the engine's working conditions, several other transient cases are selected and demonstrated, which involve various engine speed and mass of fuel injection. The designed test sequences are shown in Table 4.

**TABLE 4**
Design of Test Sequence

| ENGINE SPEED | MASS OF FUEL INJECTION | TRANSIENT PERIOD |
|---|---|---|
| 1500 rpm | 15-20 mg/stroke | 0.1s |
| | 15-40 mg/stroke | 0.1s |
| 2000 rpm | 15-20 mg/stroke | 0.1s |
| | 15-40 mg/stroke | 0.1s |

Fig.11 shows that the refined tuning of the controller parameters using the CAPSO calibration algorithm could reduce the system response time, overshoot and settling time among all cases. The averaged reductions on system overshoot and settling time are 59.9% and 35.4%. In the control

point of view, the dynamic performance of the diesel air-path is improved. With the help of the proper controlled engine's air- path, more optimal fuel economy is also acquired. The averaged reduction of fuel consumption is 0.91%. The trajectories of the engine torque are shown in Fig.12.

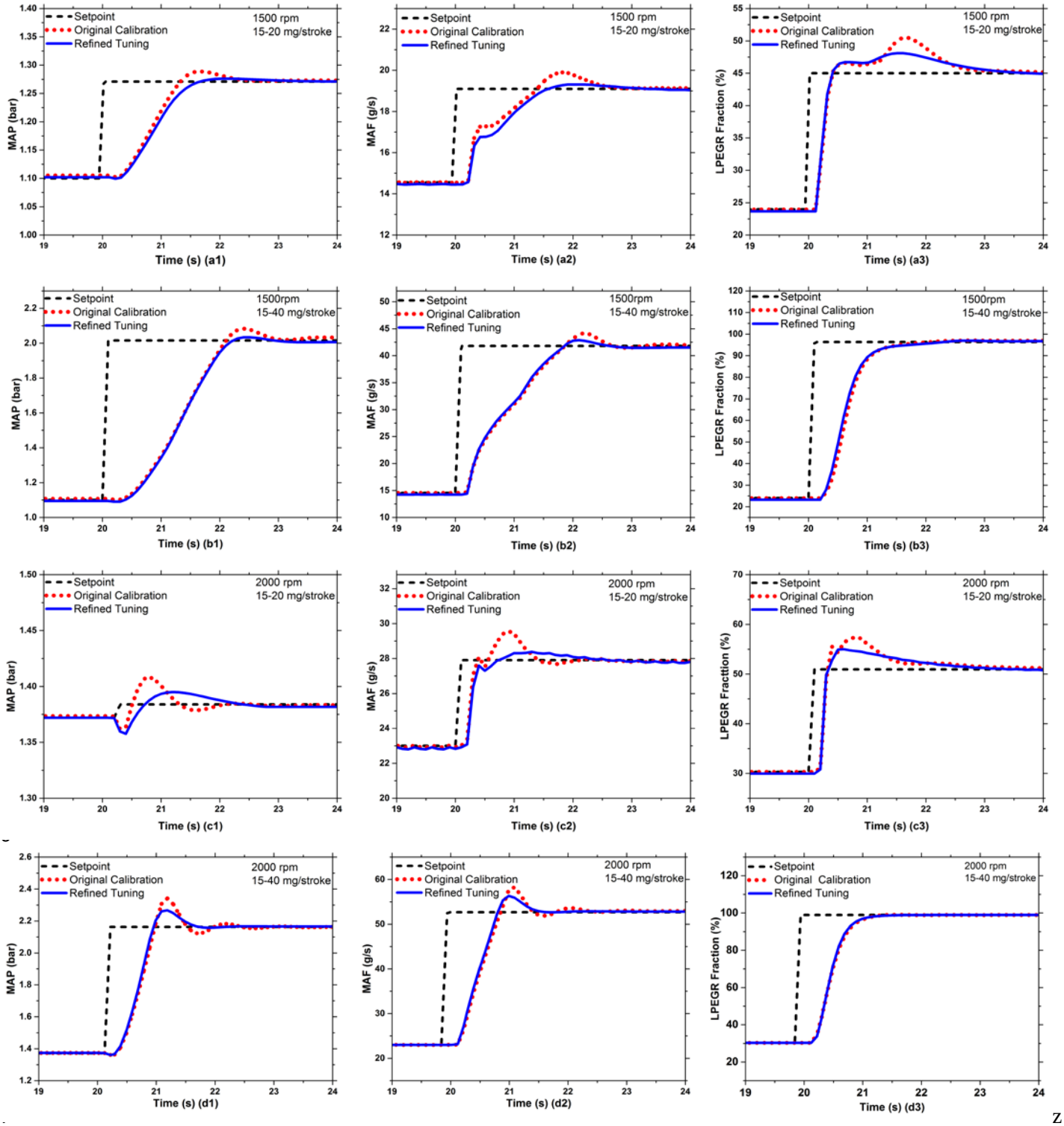



***Fig. 11*** *Trajectories of Engine Parameters under Various Transient Scenarios using CAPSO Calibration Algorithm and Original Calibration: (a) 1500rpm 15-20 mg/stroke; (b) 1500rpm 15-40 mg/stroke; (c) 2000rpm 15-20 mg/stroke; (d) 2000rpm 15-40 mg/stroke*

## 5. Conclusions

A new engine transient calibration method based on the CAPSO algorithm has been developed. The unique part of the algorithm is the chaotic mapping strategy, which contributes to finding the global optimum controller parameters. The engine performance with different tuning results has been obtained via case study. The repeatability test is also conducted to evaluate the proposed CAPSO algorithm. The conclusions are summarized as follows:

1. The proposed intelligent calibration algorithm could optimize the transient behaviour of the engine's air-path under various working conditions. The overshoot and the settling time of the controller objects are reduced 59.9% and 35.4% respectively, compared with the baseline engine.
2. With the more optimal calibration, an improved engine fuel economy is obtained. The case studies of engine indicate that the transient calibration method obtains decrement of the accumulated fuel consumption by at least 0.83%, which is a significant reduction for internal combustion engines.
3. The proposed CAPSO algorithm outperforms the conventional APSO algorithm. The Monte Carlo analysis shows that the CAPSO algorithm achieves a 4.1 % lower value of the optimization object than the conventional APSO algorithm. It is found in the repeatability test that the standard deviation of the optimization object using CAPSO algorithm is also 65.2% smaller than that using conventional APSO algorithm.

The real-time model of the diesel engine's air path in this research provides sufficient interfaces for further optimization via control strategy design. In terms of the multiple-objective optimization issue, the proposed method could also contribute to the calibration of controller with other strategies such as model predictive control and fuzzy logic control.

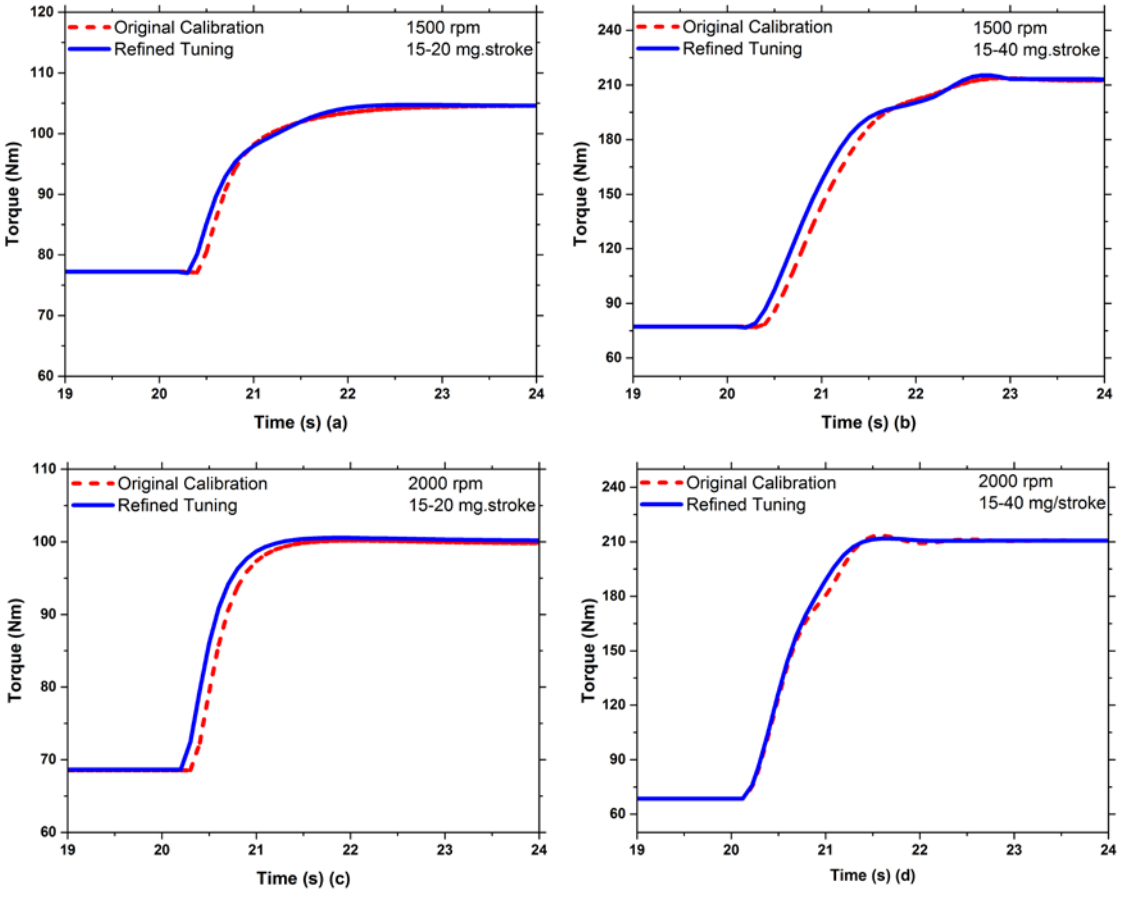


***Fig. 12*** *Trajectories of Engine Torque under Various Transient Scenarios using CAPSO Calibration Algorithm and Original Calibration: (a) 1500rpm 15-20 mg/stroke; (b) 1500rpm 15-40 mg/stroke; (c) 2000rpm 15-20 mg/stroke; (d) 2000rpm 15-40 mg/stroke*

# Reference


[1] H. Borhan, G. Kothandaraman, B. Pattel, Air handling control of a diesel engine with a complex dual-loop EGR and VGT air system using MPC, Proc. Am. Control Conf. 2015–July (2015) 4509–4516. doi:10.1109/ACC.2015.7172039.

[2] W. Zhang, Z. Chen, W. Li, G. Shu, B. Xu, Y. Shen, Influence of EGR and oxygen-enriched air on diesel engine NO-Smoke emission and combustion characteristic, Appl. Energy. 107 (2013) 304–314. doi:10.1016/j.apenergy.2013.02.024.

[3] M. Zheng, G.T. Reader, J.G. Hawley, Diesel engine exhaust gas recirculation - A review on advanced and novel concepts, Energy Convers. Manag. 45 (2004) 883–900. doi:10.1016/S0196-8904(03)00194-8.

[4] A. Maiboom, X. Tauzia, J.-F. Hétet, Experimental study of various effects of exhaust gas recirculation (EGR) on combustion and emissions of an automotive direct injection diesel engine, Energy. 33 (2008) 22–34. doi:10.1016/j.energy.2007.08.010.

[5] S. Reifarth, H.-E. Ångström, Transient EGR in a High-Speed DI Diesel Engine for a set of different EGR-routings, SAE Int. J. Engines. 3 (2010) 1071–1078. doi:10.4271/2010-01-1271.

[6] G. Zamboni, M. Capobianco, Experimental study on the effects of HP and LP EGR in an automotive turbocharged diesel engine, Appl. Energy. 94 (2012) 117–128. doi:10.1016/j.apenergy.2012.01.046.

[7] D. Zhao, C. Liu, R. Stobart, J. Deng, S. Member, E. Winward, G. Dong, An Explicit Model Predictive Control Framework for Turbocharged Diesel Engines, IEEE Trans. Ind. Electron. 61 (2014) 3540–3552.

[8] H. Ma, H. Xu, J. Wang, T. Schnier, B. Neaves, C. Tan, Z. Wang, Model-based Multi-objective Evolutionary Algorithm Optimization for HCCI Engines, IEEE Trans. Veh. Technol. 9545 (2014) 1–1. doi:10.1109/TVT.2014.2362954.

[9] S. Park, Y. Kim, S. Woo, K. Lee, Optimization and calibration strategy using design of experiment for a diesel engine, Appl. Therm. Eng. 123 (2017) 917–928. doi:10.1016/j.applthermaleng.2017.05.171.

[10] C.K. Sanathanan, FREQUENCY DOMAIN METHOD FOR TUNING HYDRO GOVERNORS., IEEE Trans. Energy Convers. 3 (1988) 14–17. doi:10.1109/60.4193.

[11] K.J. Åström, T. Hägglund, The future of PID control, Control Eng. Pract. 9 (2001) 1163–1175. doi:10.1016/S0967-0661(01)00062-4.

[12] V. Pano, P.R. Ouyang, I. Member, PSO Gain Tuning for Position Domain PID Controller, 4th Annu. IEEE Int. Conf. Cyber Technol. Autom. Control Intell. (2014) 377–382.

[13] L. Guzzella, C.H. Onder, Introduction to modeling and control of internal combustion engine systems, Springer, 2010. doi:10.1007/978-3-642-10775-7.

[14] M.H.N. Tayarani, X. Yao, H. Xu, Meta-Heuristic Algorithms in Car Engine Design: A Literature Survey, IEEE Trans. Evol. Comput. 19 (2015) 609–629. doi:10.1109/TEVC.2014.2355174.

[15] Y. Ye, C.-B. Yin, Y. Gong, J. Zhou, Position control of nonlinear hydraulic system using an improved PSO based PID controller, Mech. Syst. Signal Process. 83 (2017) 241–259. doi:10.1016/j.ymssp.2016.06.010.

[16] X.D.Z.L.J. Zhang, PSO based on chaotic Map and Its Application to PID Controller Self-tuning, in: 16th Int. Conf. Electron. Packag. Technol., IEEE, 2015: pp. 1470–1476. doi:10.1109/ICEPT.2015.7236860.

[17] G. Reynoso-Meza, X. Blasco, J. Sanchis, M. Martínez, Controller tuning using evolutionary multi-objective optimisation: Current trends and applications, Control Eng. Pract. 28 (2014) 58–63. doi:10.1016/j.conengprac.2014.03.003.

[18] H. Fang, L. Chen, Z. Shen, Application of an improved PSO algorithm to optimal tuning of PID gains for water turbine governor, Energy Convers. Manag. 52 (2011) 1763–1770. doi:10.1016/j.enconman.2010.11.005.

[19] B. Bourouba, S. Ladaci, Comparative performance analysis of GA, PSO, CA and ABC algorithms for ractional PID controller tuning, in: 2016 8th Int. Conf. Model. Identif. Control, 2016: pp. 960–965. doi:10.1109/ICMIC.2016.7804253.

[20] S. Rogers, B. Birge, Swarm Optimization Applied to Engine RPM Control, SAE Int. (2004). doi:10.4271/2004-01-2669.

[21] H.C. Watson, A. Ratnawera, S. Halgamuge, Optimization of all SI engine combustion control and related events for efficiency, SAE Tech. Pap. (2006). doi:10.4271/2006-01-0045.

[22] N. Ben Guedria, Improved accelerated PSO algorithm for mechanical engineering optimization problems, Appl. Soft Comput. J. 40 (2016) 455–467. doi:10.1016/j.asoc.2015.10.048.

[23] I. Rahman, P.M. Vasant, B.S.M. Singh, M. Abdullah-Al-Wadud, On the performance of accelerated particle swarm optimization for charging plug-in hybrid electric vehicles, Alexandria Eng. J. 55 (2016) 419–426. doi:10.1016/j.aej.2015.11.002.

[24] L. Shen, L. Xu, R. Wei, L. Cao, Multi-swarm Optimization with Chaotic Mapping for Dynamic Optimization Problems, in: Proc. - 2015 8th Int. Symp. Comput. Intell. Des. Isc. 2015, IEEE, 2016: pp. 132–137. doi:10.1109/ISCID.2015.173.

[25] D. Tan, Chaos Particle Swarm Optimization Algorithm for Multi-Objective Constrained Optimization Problems, in: Springer Berlin Heidelberg, 2012: pp. 469–476. doi:10.1007/978-3-642-27326-1_60.

[26] Z. Yang, Y. Yang, H. Yang, L. Zhang, An improved particle swarm optimization method based on chaos, in: 2014 10th Int. Conf. Nat. Comput., IEEE, 2014: pp. 209–213. doi:10.1109/ICNC.2014.6975836.

[27] B. Liu, L. Wang, Y.-H. Jin, F. Tang, D.-X. Huang, Improved particle swarm optimization combined with chaos, Chaos, Solitons & Fractals. 25 (2005) 1261–1271. doi:10.1016/j.chaos.2004.11.095.

[28] M. Pluhacek, R. Senkerik, I. Zelinka, D. Davendra, Designing PID Controllers by Means of PSO Algorithm Enhanced by Various Chaotic Maps, in: 2013 8th EUROSIM Congr. Model. Simul., IEEE, 2013: pp. 19–23. doi:10.1109/EUROSIM.2013.14.

[29] Universitat Politècnica de València, Development of methods for evaluating real time turbocharged engine models, 2015.

[30] Y. Zhang, G. Lu, H. Xu, Z. Li, Tuneable model predictive control of a turbocharged diesel engine with dual loop exhaust gas recirculation, Proc. Inst. Mech. Eng. Part D J. Automob. Eng. (2017) 095440701772694. doi:10.1177/0954407017726944.

[31] C. Tan, Model Based Control for a Modern Automotive, University of Birmingham, 2015.

[32] Z. Wang, Q. Su, X. Luo, A novel HTD-CS based PID controller tuning method for time delay continuous systems with multi-objective and multi-constraint optimization, Chem. Eng. Res. Des. 115 (2016) 98–106. doi:10.1016/j.cherd.2016.09.025.

[33] C.M. Fonseca, P.J. Fleming, Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization, Icga. 93 (1993) 416–423. doi:citeulike-article-id:2361311.

[34] X.-S. Yang, Nature-Inspired Optimization Algorithms, 2014. doi:http://dx.doi.org/10.1016/B978-0-12-416743-8.00017-8.

[35] A.H. Gandomi, G.J. Yun, X.S. Yang, S. Talatahari, Chaos-enhanced accelerated particle swarm optimization, Commun. Nonlinear Sci. Numer. Simul. 18 (2013) 327–340. doi:10.1016/j.cnsns.2012.07.017.

[36] Q. Zhou, W. Zhang, S. Cash, O. Olatunbosun, H. Xu, G. Lu, Intelligent sizing of a series hybrid electric power-train system based on Chaos-enhanced accelerated particle swarm optimization, Appl. Energy. 189 (2017) 588–601. doi:10.1016/j.apenergy.2016.12.074.

[37] J. Tian, H. Xu, R. Arumugam Sakunthalai, D. Liu, C. Tan, A. Ghafourian, Low Ambient Temperature Effects on a Modern Turbocharged Diesel engine running in a Driving Cycle, SAE Int. J. Fuels Lubr. 7 (2014) 2014-01-2713. doi:10.4271/2014-01-2713.

[38] T. V Johnson, Diesel emission control in review, Sae Trans. 110 (2001) 128–144. doi:doi:10.4271/2006-01-0030.