International Journal of Automotive Technology, Vol. 18, No. 2, pp. 195–207 (2017)
DOI 10.1007/s12239–017–0019–3

pISSN 1229–9138/ eISSN 1976–3832

GAIN SCHEDULED LINEAR QUADRATIC TRACKING SYSTEM TUNED OPTIMALLY BY COVARIANCE MATRIX ADAPTION EVOLUTIONARY STRATEGY FOR AUTOMOTIVE ENGINE COLDSTART CONTROL

N. L. AZAD[1)], A. MOZAFFARI[1)*] and J. K. HEDRICK[2)]

[1)]Systems Design Engineering Department, University of Waterloo, Waterloo N2L 3G1, Canada
[2)]Department of Mechanical Engineering, University of California Berkeley, Berkeley CA 94720, USA

(Received 20 May 2015; Revised 5 April 2016; Accepted 5 April 2016)

ABSTRACT–In this paper, a gain scheduled linear quadratic tracking system (LQTS) tuned optimally by an evolutionary strategy (ES) is devised to reduce the total tailpipe hydrocarbon (*HC*) emissions of an automotive engine over the coldstart period. As the engine's behavior during coldstart operations is nonlinear, the system dynamics is clearly analyzed and represented by a number of separate linear models generated based on a coldstart model verified by experimental data. An independent LQTS is then implemented for each of these linear models. In this way, several control laws are created, and the corresponding gains are calculated for each of the independent control laws. ES is then used to tune the adjustable parameters of LQTSs to calculate the control inputs, namely air/fuel ratio (*AFR*) and spark timing (Δ), such that the resulting exhaust gas temperature (T_{exh}) and engine-out *HC* emissions (HC_{raw}) be close to a set of optimum profiles. This enables the controller reduce the cumulative tailpipe hydrocarbon emissions (HC_{cum}) to the highest possible extent. To demonstrate the acceptable performance of the proposed controller, an optimal controller derived from the Pontryagin's minimum principle (PMP) is also taken into account. Based on the results of the conducted comparative study, it is shown that the proposed control technique has a very good performance, and also, can be easily used for real-time applications, as it consumes a remarkably trivial computational time for calculating the controlling commands.

KEY WORDS : Coldstart, Automotive engines, Gain scheduled linear quadratic tracking system, Evolutionary strategy

1. INTRODUCTION

Due to the existing environmental concerns and limited energy resources, in the recent years, an enormous trend in the automotive industry has emerged towards improving the quality of cars' subsystems, in particular internal combustions engines. One of the remarkable research tendencies in this area pertains to the reduction of fuel consumption and pollutant emission of spark-ignited (SI) engines (Rasul and Glasgow, 2005). The increasingly tight regulations for satisfying the global demands and pressures to overcome the environmental concerns have instigated the automotive engineering society and the related research communities to put more efforts into reducing the amount of tailpipe emissions from SI engines.

This problem entails several different aspects which are worth for further investigations. To have a significant contribution in reducing the emission rate of SI engines, precise analyses of various parts of the engine should be performed, and its performance over the working period should be carefully studied. It is also necessary to find out which emitted pollutants are contributing to the environmental contamination.

The outcomes of the conducted studies indicate that unburned hydrocarbons (*HC*s) are one of the most destructive pollutants emitted from SI engines. The emitted *HCs* can cause serious consequences for the environment, and also, result in some undesirable health problems. Such an observation has obliged automotive engineers to make significant efforts on reducing the rate of emitted *HCs*. In this context, numerous advanced technological tools as well as high-performance computational methods have been applied to improve the performance of SI engines to decrease their *HC* emissions. Among the existing technologies, the catalytic converter has been proven to be one of the most efficient tools for reducing *HCs* and also other pollutants form cars. Although the introduction of catalytic converters has made a profound impact on decreasing the rate of *HC* emissions from SI engines, it still has some performance flaws which should be further improved. To be more to the point, catalytic converters reach their nominal performances with conversion efficiencies of 95 % or higher only in certain temperature ranges, typically, when the engine is fully warmed-up. Before arriving at the warmed-up phase, the conversion efficiency of a catalytic converter is very low, which leads

**Corresponding author.* e-mail: amozaffa@uwaterloo.ca

to remarkable emissions of *HCs* from the engine (Sanketi, 2009). The time required for the warming-up of catalytic converter is around 1 ~ 2 minutes. This period is known as the coldstart period (Zavala, 2007). The conducted investigations indicate that, for a typical drive cycle, about 90 % of the emitted *HCs* are expelled over the coldstart period for a given SI engine.

Automotive system design engineers have exerted a considerable effort for reducing the time required for increasing the catalytic converters' temperatures in SI engines over the coldstart period, to enhance their conversion efficiency quickly. To do so, both advancing the components of the engine system and developing much more efficient controlling algorithms have been taken into account (Zavala, 2007). An active catalyst heating system is an example of the technological advancements to expedite the warming-up procedure. However, from a financial viewpoint, industrialists have reached the conclusion that investing on equipping an SI engine system with such additional instruments will result in considerable extra costs. Therefore, more attention has been paid on developing computationally-efficient control algorithms to optimally manage the performances of the engine and catalytic converter over the coldstart period to minimize the tailpipe *HC* emissions (Azad *et al*., 2012). Given the fact that an increasing interest from the automotive industry is emerging towards advanced model-based controllers, the development of such control algorithms for the coldstart problem in real-time applications can be pursued from two different aspects. In one hand, it is of high importance to seek for sufficiently accurate, but simple control-oriented models to properly represent the coldstart behavior of a given engine. On the other hand, investigations can be carried out by searching for computationally efficient controlling algorithms to optimally calculate the control commands. A comprehensive survey on developing both high-fidelity and low-order control-oriented models for the coldstart problem can be found in previous studies by the authors' research group as well as other research teams (Azad *et al*., 2012; Mozaffari and Azad, 2014; Aquino, 1981; Tseng and Cheng, 1999; Shaw and Hedrick, 2003), and thus, the authors avoid providing redundant descriptions here. By checking the provided references, one can easily observe that several high-fidelity as well as simplified control-oriented models have been successfully developed and verified in the literature, which can be used for designing and evaluating coldstart controllers. However, the development of more effective real-time controlling algorithms for the coldstart problem is still a hot topic of investigation and has actively been pursued by automotive control experts. Although the existing coldstart control algorithms have different structures and functionalities, they mostly pursue a similar strategy for reducing tailpipe *HC* emissions. To be more precise, the conducted researches have indicated that the cumulative tailpipe *HC* emission (HC_{cum}) of an SI engine during the coldstart operation will be decreased by increasing the exhaust gas temperature (T_{exh}) and reducing the engine-out *HC* emission (HC_{raw}). Therefore, an effective coldstart emission-reduction controller should calculate the controlling commands such that a logical trade-off between the above-mentioned effects is achieved (Zavala, 2007), to optimally reduce HC_{cum}. Among the reported control techniques, those implemented based on the concepts of hybrid switching control (HSC) (Sanketi *et al*., 2006; Salehi *et al*., 2014), sliding mode control (SMC) (Sanketi *et al*., 2007), nonlinear model predictive control (NMPC) (Mozaffari *et al*., 2016), and Pontryagin's minimum principle (PMP) (Azad *et al*., 2012) have received a great of attention.

The overall feedback of the conducted studies indicates that one of the best ways of the control of an automotive engine over the coldstart period is to define a set of desired trajectories and then try to track them to minimize the total amount of HC_{cum}. Also, each of the above-mentioned controllers has its own advantages and downsides. For example, the PMP method is an open-loop control strategy and only determines a set of offline controlling commands which are used to regulate the engine over the coldstart period, but it is difficult to make this method online. On the other hand, HSC and MPC controllers have a feedback type configuration based on the states of the real engine. Also, the calculation of the near-optimal control inputs for MPC involves solving an optimization problem in real-time which makes it a bit complex. Taking the above-mentioned issues into account, it is worth to develop a controller for the coldstart problem which can be easily implemented for real-time applications, and at the same time, has a closed-loop form, and most importantly, can yield, at least, near-optimal solutions. Such a fact has motivated the authors to seek for designing a coldstart controller for a given automotive engine, which has a simple architecture for the real-time calculation of the near-optimum control inputs. The resulting controller is known as gain scheduled linear quadratic tracking system (GS-LQTS) (Naidu, 2003), which is best suited for tracking a desired trajectory, and at the same time, calculates the control commands very quickly for real-time applications. The gain scheduled scheme enables us divide the considered nonlinear control-oriented coldstart model into a set of linear models, and then, apply LQTS for the calculation of controlling sequence for each linear model. Furthermore, in this study, an optimization method, called covariance matrix adaption evolutionary strategy (CMA-ES) (Hansen and Ostermeier, 2001), is used to determine the optimum values of a number of weighting coefficients in the formulation of LQTS. Based on some simulations, the authors indicate how well the proposed control algorithm can be employed for the coldstart problem.

The rest of the paper is organized, as follows. Section 2 is devoted to the description of the coldstart control-oriented model developed in the Vehicle Dynamics and

Control Lab (VDL) at University of California (UC), Berkeley. The development of the gain scheduled linear quadratic tracking system (GS-LQTS) is described in Section 3. The algorithmic structure of covariance matrix adaption evolutionary strategy (CMA-ES) for optimizing the weighting coefficients of GS-LQTS is presented in Section 4. Section 5 is devoted to the description of results and discussions. Finally, the concluding remarks are given in Section 6.

2. CONTROL-ORIENTED MODEL

In this section, the authors explain the steps taken for developing the coldstart control-oriented model for the considered SI engine. Firstly, the existing coldstart experimental setup at UC, Berkeley is introduced, and thereafter, the steps required for building the control-oriented model as well as the validation procedure are given.

2.1. Experimental Setup

To generate the control-oriented model, the design of experiments (DoE) technique was taken into account to capture the required information for the performance of the engine over the coldstart period. For doing this procedure, an instrumented Toyota engine with several sensors and actuators was employed. All of the experimental work for collecting the required information was performed at the UC, Berkeley's coldstart research facility. The considered engine and the related instruments for coldstart experiments are shown in Figure 1. The depicted engine can generate a maximum power of 117 KW at 5,600 rpm. It is also coupled to a dynamometer to simulate various loading conditions. Moreover, a dyno-controller regulates the speed and torque of the dynamometer. The engine also hosts a number of sensors to measure and monitor the main engine variables, including an intake pressure sensor, several thermocouples, and a couple of air/fuel ratio (*AFR*) sensors. There is also an emission analyzer to measure the rate of *HC* emissions. The above-mentioned coldstart experimental platform was used to create the control-oriented model, as described in the next section.

Figure 1. Considered engine and the related instruments for coldstart experiments.

2.2. Modeling and Validation

Based on a comprehensive empirical study, the effects of different variables on the performance of the engine over the coldstart were analyzed. The coldstart experimental results indicated that the spark timing (Δ), air/fuel ratio (*AFR*), and the engine's speed (ϖ_e) had the highest impacts on the variations of T_{exh} and engine-out *HC* emission concentration ($HC_{raw\text{-}c}$), which are both crucial for the calculation of cumulative tailpipe *HC* emissions (HC_{cum}) as shown later. The coldstart measurements demonstrated that the dynamic behaviour of T_{exh} can be approximated by a first-order linear system for each following input: $u_1 = \Delta$ (deg. *ATDC*) + 50, $u_2 = AFR$, and $u_3 = \varpi_e$. The experiments also indicated that the dynamic behaviour of $HC_{raw\text{-}c}$ with respect to each above-mentioned input can be represented with a similar first-order linear system approximation (Sun and Sivashankar, 1999). However, as shown later, the resulting equations are not linear due to the considered saturation limits and excluding specific negative excitations. Moreover, the dynamic equations associated with T_{exh} and $HC_{raw\text{-}c}$ have different sets of the parameters. These parameters were estimated using the collected experimental data. As the considered GS-LQTS controller has a discrete-time formulation, the original ordinary differential equations (ODEs) for T_{exh} and $HC_{raw\text{-}c}$ are transformed into difference forms, making them applicable for GS-LQTS.

The difference forms of the coldstart engine system's equations are given below;

$$
\begin{aligned}
x_1(k) &= \delta t \cdot \frac{u_1(k-1)}{\tau_1} + \left(1 - \frac{\delta t \cdot k_1}{\tau_1}\right) x_1(k-1) \\
x_2(k) &= \delta t \cdot \frac{u_3(k-1)}{\tau_2} + \left(1 - \frac{\delta t \cdot k_2}{\tau_2}\right) x_2(k-1) \\
x_3(k) &= \delta t \cdot \frac{16 - u_2(k-1)}{\tau_3} + \left(1 - \frac{\delta t \cdot k_3}{\tau_3}\right) x_3(k-1) \\
x_4(k) &= \delta t \cdot \frac{u_3(k-1) - 800}{\tau_4} + \left(1 - \frac{\delta t \cdot k_4}{\tau_4}\right) x_4(k-1) \\
x_5(k) &= \delta t \cdot \frac{16 - u_2(k-1)}{\tau_5} + \left(1 - \frac{\delta t \cdot k_5}{\tau_5}\right) x_5(k-1) \\
x_6(k) &= \delta t \cdot \frac{|u_1(k-1) - 55| + (u_1(k-1) - 55)}{2\tau_6} + \left(1 - \frac{\delta t \cdot k_6}{\tau_6}\right) x_6(k-1)
\end{aligned}
\tag{1}
$$

where δt shows the time interval between two sequential values of the system's states x, and k represents the

Table 1. Parameters of the control-oriented model.

τ_1	τ_2	τ_3	τ_4	τ_5	τ_6
2.9629	156.2661	0.1800	1.1667	0.0002	0.0150
k_1	k_2	k_3	k_4	k_5	k_6
0.1997	5.2708	0.8527	0.1667	0.001	0.0075

corresponding time step for the above coldstart control-oriented model. The values of the model parameters are listed in Table 1.

The first three state equations are related to the calculation of T_{exh}, and the last three state equations are required for calculating $HC_{raw\text{-}c}$. The output variables, namely T_{exh} and $HC_{raw\text{-}c}$, are found by;

$$T_{exh}(k) = \max(x_1(k) + x_3(k), 0) + x_2(k) \quad (2)$$

$$HC_{raw-c}(k) = \max(4000 - x_4(k), 800) + \max(x_5(k) + x_6(k), 0) \quad (3)$$

where, $T_{exh}(k)$ and $HC_{raw\text{-}c}(k)$ are in ^{o}C and *ppm*, respectively.

Some validation tests have been carried out using a different set of the input signals to evaluate the accuracy of the original ODEs representing the coldstart behaviour. The output profiles obtained by the control-oriented model and those resulting from the experiments are indicated in Figure 2. It can be seen that the output signals calculated by the control-oriented model are in a good correlation with the experimental data, and thus, the presented model is appropriate to be used for designing GS-LQTS.

It was mentioned previously that the value of HC_{cum} is related to the efficiency of catalyst (η). The following formulation has been empirically derived for the calculation of η (Azad *et al*., 2012);

$$\eta = (1-\eta_1)(1-\eta_2) \quad (4)$$

where:

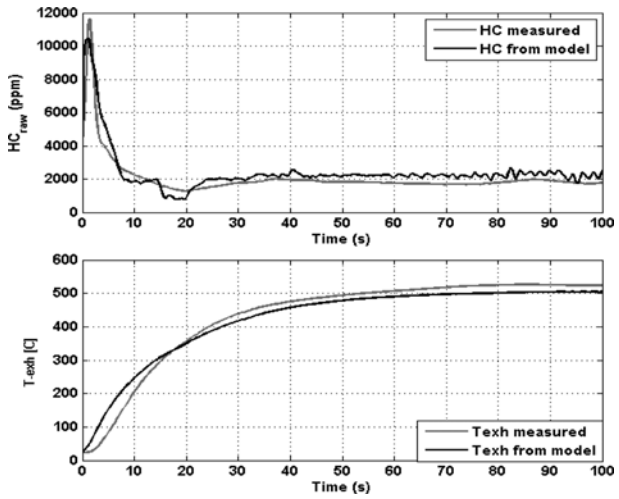

Figure 2. Validation tests for HC_{raw} and T_{exh}.

$$\eta_1(k) = \exp\left(-a_1\left(\frac{\max(u_2(k), 0) - \lambda_0}{\Delta\lambda}\right)^{m_1}\right) \quad (5)$$

$$\eta_2(k) = \exp\left(-a_2\left(\frac{\max(T_{cat}(k), T_{cat0}) - T_{cat0}}{\Delta T_{cat}}\right)^{m_2}\right) \quad (6)$$

The values of equation parameters, namely a_1, a_2, m_1, and m_2, have been identified using some experimental data (Sanketi, 2009). Also, the value of the catalyst temperature (T_{cat}) depends primarily on T_{exh}. This is because most of the heating energy to warm up the catalytic converter comes from the exhaust gas temperature. The equation below has been proposed for the calculation of T_{cat};

$$T_{cat}(k) = k_{cat} \cdot T_{exh}(k) \quad (7)$$

Also, Azad *et al*. (2012) has proposed the following formulation to calculate HC_{cum};

$$HC_{cum} = \int_0^T (1-\eta)\dot{HC}_{raw} \cdot dt = \int_0^T (1-\eta)\dot{m}_{exh}\left(\frac{16}{28.5}\times 10^{-6}\right) HC_{raw-c} \cdot dt \quad (8)$$

where, $\dot{m}_{exh}$ can be calculated by the following equation;

$$\dot{m}_{exh} = b_1 u_3 + b_2 \quad (9)$$

In this work, the third control command (engine speed) is considered as a predefined input, and thus, only four states and two inputs should be controlled by the controller.

3. GAINED SCHEDULED LINEAR QUADRATIC TRACKING SYSTEM

In this section, the authors explain the steps required for the implementation of GS-LQTS. The implementation of the controller includes a set of steps. Firstly, the considered nonlinear control-oriented model is divided into a number of linear models. Then, each of these linear models are used to design an independent LQTS, and the corresponding Riccati matrixes are calculated offline for each of those controllers. By implementing a hyper-level rule-base, a recommender is provided to activate the appropriate control in a real-time fashion so that the proper controlling commands be dispatched to the engine for appropriate performance (Naidu, 2003). In this way, the hyper-level rule-base considers as a schedule for determining the correct gain at any time during the control process.

At the first step, the authors separate the nonlinear model into a set of linear models, and based on that derive the conditions required for scheduled the feed-back gains. Thereafter, independent LQTSs are implemented for each of the conditions and the corresponding Riccati matrixes are derived to be used during the coldstart period.

To better understand the required strategy for the implementation of GS-LQTS, firstly, the authors present the standard formulation of linear quadratic tracking system.

3.1. Linear Quadratic Tracking System

In this investigation, the authors consider a discrete-time variant of LQTS which is fit for the state-space formulation of the nonlinear control-oriented model. This variant of linear quadratic regulator (LQR) is originally proposed for tracking a desired trajectory over a given time interval for a linear system. Assume that the algebraic formulation of a linear, time-invariant system is as below;

$$\mathbf{x}(k+1)=\mathbf{A}\,\mathbf{x}(k)+\mathbf{B}\,\mathbf{u}(k) \tag{10}$$

$$\mathbf{y}(k+1)=\mathbf{C}\,\mathbf{x}(k) \tag{11}$$

where **A**, **B**, and **C** are parameters of the state-space formulation, $\mathbf{x} = [x_1, x_3, x_5, x_6]$ is the state vector including a set of system states, $\mathbf{u} = [u_1, u_2]$ is the controlling effort, and **y** is the output vector, containing the outputs of the system.

Then, the general convex quadratic objective formulation can be presented as:

$$\begin{aligned} J(k_0)=&\frac{1}{2}[\mathbf{C}\,\mathbf{x}(k_f)-\mathbf{y}_d(k_f)]^T\mathbf{F}[\mathbf{C}\,\mathbf{x}(k_f)-\mathbf{y}_d(k_f)]\\ &+\frac{1}{2}\sum_{k=k_0}^{k_f-1}\{[\mathbf{C}\,\mathbf{x}(k)-\mathbf{y}_d(k)]^T\mathbf{Q}[\mathbf{C}\,\mathbf{x}(k)-\mathbf{y}_d(k)]+\mathbf{u}^T(k)\mathbf{R}\mathbf{u}(k)\} \end{aligned} \tag{12}$$

where k_0 and k_f are the initial and final time of the procedure, and $\mathbf{x}(k_0)$ and $\mathbf{x}(k_f)$ are the initial and final states which can be predefined or free.

As the state-space formulation is convex, the above quadratic convex objective function has an analytical solution which has been derived in Naidu (2003). Based on the solving procedure presented in Naidu (2003), the matrix difference Riccati equation (**P**) and vector difference equation (**g**) should be defined to calculate the optimal controlling efforts;

$$\mathbf{P}(k)=\mathbf{A}^T\mathbf{P}(k+1)\left[\mathbf{I}+\mathbf{E}\mathbf{P}(k+1)\right]^{-1}\mathbf{A}+\mathbf{V} \tag{13}$$

where $\mathbf{P}(k_f)=\mathbf{C}^T\mathbf{F}\mathbf{C}$, $\mathbf{V}=\mathbf{C}^T\mathbf{Q}\mathbf{C}$, and $\mathbf{E}=\mathbf{B}^T\mathbf{R}^{-1}\mathbf{B}$.

$$\mathbf{g}(k)=\mathbf{A}^T\{\mathbf{I}-[\mathbf{P}(k+1)+\mathbf{E}]^{-1}\mathbf{E}\}\mathbf{g}(k+1)+\mathbf{W}\mathbf{y}_d(k) \tag{14}$$

where $\mathbf{g}(k_f)=\mathbf{C}^T\mathbf{F}\mathbf{y}_d(k_f)$, $\mathbf{y}_d(k)$ is the desired trajectory to be tracked, and $\mathbf{W}=\mathbf{C}^T\mathbf{Q}$.

Then, the optimal states $\mathbf{x}^*$ can be calculated, as follows;

$$\mathbf{x}^*(k+1)=\left[\mathbf{A}-\mathbf{B}\mathbf{L}(k)\right]\mathbf{x}^*(k)+\mathbf{B}\mathbf{L}_g(k)g(k+1) \tag{15}$$

where $\mathbf{L}(k)=[\mathbf{R}+\mathbf{B}^T\mathbf{P}(k+1)\mathbf{B}]^{-1}\mathbf{B}^T\mathbf{P}(k+1)\mathbf{A}$ and $\mathbf{L}_g(k)=[\mathbf{R}+\mathbf{B}^T\mathbf{P}(k+1)\mathbf{B}]^{-1}\mathbf{B}^T$.

Finally, the optimal controlling commands ($\mathbf{u}^*$) can be calculated in a real-time fashion using the feedback of the system's states, as given below;

$$\mathbf{u}^*(k+1)=-\mathbf{L}(k)\mathbf{x}^*(k)+\mathbf{L}_g(k)\mathbf{g}(k+1) \tag{16}$$

The above-mentioned mathematical formulations are used to implement LQTS. In the rest of this section, the authors use LQTS to design different control laws to cover various operating conditions which can be experienced over the controlling process.

3.2. Scheduled for Different Operating Conditions

By checking the formulation of the control-oriented model presented in Equation (1), one can easily infer that some modifications should be made to turn this coldstart model into the form of standard linear model used for LQTS. This can be done by using a change of variables for the states. Meanwhile, the considered control problem involves a specific range for the controlling commands. In a previous study by the authors' research group, it was demonstrated that due to some practical considerations, the controlling commands had to lie within the following ranges (Azad *et al*., 2012);

$$\begin{cases} 40^\circ \le u_1 \le 60^\circ \\ 10 \le u_2 \le 16 \end{cases} \tag{17}$$

By defining two revised input variables, it can be ensured that LQTS tries to maintain the actual controlling commands in the above-mentioned ranges by minimizing the objective function. The modified inputs are given by;

$$\begin{cases} \nu_1(k)=u_1(k)-\nu_{ref\,1}(k) \\ \nu_2(k)=u_2(k)-\nu_{ref\,2}(k) \end{cases} \tag{18}$$

The stoichiometric air to fuel ratio is used as the reference value for *AFR* (14.7) or u_2. Moreover, the average amount, namely 50, is considered as the reference value for u_1.

Referring to the 6th state equation of the control-oriented model, to create a set of linear systems, two cases, namely $\nu_1 \ge 5$ and $\nu_1 < 5$, should be considered. Now, let us define the following new states after a change of variables;

$$\begin{cases} x_1(k)=z_1(k)+\psi_1 \\ x_3(k)=z_3(k)+\psi_2 \\ x_5(k)=z_5(k)+\psi_3 \\ x_6(k)=z_6(k)+\begin{cases}\psi_4 & if\ \nu_1 \ge 5\\ \psi_5 & if\ \nu_1 < 5\end{cases} \end{cases} \tag{19}$$

where

$$\begin{bmatrix}\psi_1\\ \psi_2\\ \psi_3\\ \psi_4\\ \psi_5\end{bmatrix}=\begin{bmatrix}50/k_1\\ 1.3/k_3\\ 1.3/k_5\\ -5/k_6\\ 0\end{bmatrix} \tag{20}$$

Then, a new set of the state-space equations are created, as given below;

$$
\begin{aligned}
z_1(k) &= \left(1-\frac{\delta t\cdot k_1}{\tau_1}\right)z_1(k-1)+\frac{\delta t}{\tau_1}\nu_1(k-1)\\
z_3(k) &= \left(1-\frac{\delta t\cdot k_3}{\tau_3}\right)z_3(k-1)-\frac{\delta t}{\tau_3}\nu_2(k-1)\\
z_5(k) &= \left(1-\frac{\delta t\cdot k_5}{\tau_5}\right)z_5(k-1)+\frac{\delta t}{\tau_5}\nu_2(k-1)
\end{aligned}
\qquad (21)
$$

$$
\begin{cases}
z_6(k) = \left(1-\dfrac{\delta t\cdot k_6}{\tau_6}\right)z_6(k-1)+\dfrac{\delta t}{\tau_6}\nu_1(k-1) & \text{if } \nu_1 \ge 5\\
z_6(k) = \left(1-\dfrac{\delta t\cdot k_6}{\tau_6}\right)z_6(k-1) & \text{if } \nu_1 < 5
\end{cases}
$$

For the first case $\nu_1 \ge 5$, the system matrixes associated with the LQTS design are as given below;

$$
\mathbf{A}=\begin{bmatrix}
\left(1-\dfrac{\delta t\cdot k_1}{\tau_1}\right) & 0 & 0 & 0\\
0 & \left(1-\dfrac{\delta t\cdot k_3}{\tau_3}\right) & 0 & 0\\
0 & 0 & \left(1-\dfrac{\delta t\cdot k_5}{\tau_5}\right) & 0\\
0 & 0 & 0 & \left(1-\dfrac{\delta t\cdot k_6}{\tau_6}\right)
\end{bmatrix}
$$

$$
\mathbf{B}=\begin{bmatrix}
\dfrac{\delta t}{\tau_1} & 0 & 0 & \dfrac{\delta t}{\tau_6}\\
0 & \dfrac{-\delta t}{\tau_3} & \dfrac{-\delta t}{\tau_5} & 0
\end{bmatrix}^{\mathrm{T}}
\qquad (22)
$$

$$
\mathbf{C}=\begin{bmatrix}1&1&0&0\\0&0&1&1\end{bmatrix};\ \mathbf{Q}=\begin{bmatrix}q_1&0\\0&q_2\end{bmatrix};\ \mathbf{R}=\begin{bmatrix}r_1&0\\0&r_2\end{bmatrix};\ \mathbf{F}=\begin{bmatrix}f_1&0\\0&f_2\end{bmatrix}
\qquad (23)
$$

$$
\mathbf{E}=\begin{bmatrix}
\dfrac{(\delta t)^2}{r_1(\tau_1)^2} & 0 & 0 & \dfrac{(\delta t)^2}{r_1\tau_1\tau_6}\\
0 & \dfrac{(\delta t)^2}{r_2(\tau_3)^2} & \dfrac{(\delta t)^2}{r_2\tau_3\tau_5} & 0\\
0 & \dfrac{(\delta t)^2}{r_2\tau_3\tau_5} & \dfrac{(\delta t)^2}{r_2(\tau_5)^2} & 0\\
\dfrac{(\delta t)^2}{r_1\tau_1\tau_6} & 0 & 0 & \dfrac{(\delta t)^2}{r_1(\tau_6)^2}
\end{bmatrix};\ \mathbf{W}=\begin{bmatrix}q_1&0\\q_1&0\\0&q_2\\0&q_2\end{bmatrix};
$$

$$
\mathbf{V}=\begin{bmatrix}q_1&q_1&0&0\\q_1&q_1&0&0\\0&0&q_2&q_2\\0&0&q_2&q_2\end{bmatrix}
\qquad (24)
$$

For the second case $\nu_1 < 5$, the system matrixes are expressed by;

$$
\mathbf{B}=\begin{bmatrix}
\dfrac{\delta t}{\tau_1} & 0 & 0 & 0\\
0 & \dfrac{-\delta t}{\tau_3} & \dfrac{-\delta t}{\tau_5} & 0
\end{bmatrix}^{\mathrm{T}};\ \mathbf{Q}=\begin{bmatrix}q_3&0\\0&q_4\end{bmatrix};\ \mathbf{R}=\begin{bmatrix}r_3&0\\0&r_4\end{bmatrix};
$$

$$
\mathbf{F}=\begin{bmatrix}f_3&0\\0&f_4\end{bmatrix}
\qquad (25)
$$

$$
\mathbf{W}=\begin{bmatrix}q_3&0\\q_3&0\\0&q_4\\0&q_4\end{bmatrix};\ \mathbf{V}=\begin{bmatrix}q_3&q_3&0&0\\q_3&q_3&0&0\\0&0&q_4&q_4\\0&0&q_4&q_4\end{bmatrix}
$$

$$
\mathbf{E}=\begin{bmatrix}
\dfrac{(\delta t)^2}{r_1(\tau_1)^2} & 0 & 0 & 0\\
0 & \dfrac{(\delta t)^2}{r_2(\tau_3)^2} & \dfrac{(\delta t)^2}{r_2\tau_3\tau_5} & 0\\
0 & \dfrac{(\delta t)^2}{r_2\tau_3\tau_5} & \dfrac{(\delta t)^2}{r_2(\tau_5)^2} & 0\\
0 & 0 & 0 & 0
\end{bmatrix}
\qquad (26)
$$

It is worth pointing out that the formulation of **A**, **C**, and **V** are the same as those of the first case. Considering the obtained values of **A**, **B** and **C**, for the both cases, the resulting systems are both observable and controllable.

After the creation of the two sets of linear state-space equations, the outputs of the control-oriented model with respect to the new states should be formulated, as shown below;

$$
\begin{cases}
T_{\mathrm{exh}}(k) = \max(z_1(k)+z_3(k)+\psi_1+\psi_2,\ 0)+x_2(k)\\
HC_{\mathrm{raw-e}}(k) = \max(4000-x_4(k),\ 800)\\
\qquad\qquad +\max(z_5(k)+z_6(k)+\psi_3+\psi_4,\ 0)
\end{cases}
\qquad (27)
$$

It can be seen that the above formulations for T_{exh} and $HC_{\mathrm{raw\text{-}c}}$ have some additional terms other than $\mathbf{C}\,z(k)$ which makes them incompatible with the standard form. However, it is possible to get rid of these additional terms by revising the desired output matrix $\mathbf{y}_{\mathrm{d}}(k)=\begin{bmatrix}HC_{\mathrm{d}}(k)\\T_{\mathrm{d}}(k)\end{bmatrix}$, and turn the new output formulations into the standard form. Also, the output formulations include a set of "max" terms to represent the saturations effects. These notations result in the non-linearity of the output formulations, but they can be transformed into a set of linear formulations as a function of the operating conditions. To proceed with the implementation of LQTS, we need to check these conditions and determine the corresponding revised desired trajectory, and then by using the standard formulation given in the previous subsection, the calculation of controlling commands can be easily performed. The following eight operating conditions are considered for converting the existing nonlinear output formulations into a set of linear equations. During the control process, the controller can switch from one schedule to another one to calculate the appropriate inputs. The first four conditions are for $\nu_1(k-1)\ge 5$, and the last four schedules are for $\nu_1(k-1)<5$;

1st schedule: ($\nu_1(k-1)\ge 5$)

$$
\begin{cases}
z_1(k)+z_3(k)+\psi_1+\psi_2 \ge 0 \rightarrow T_{\mathrm{exh}}(k)\\
\qquad = z_1(k)+z_3(k)+\psi_1+\psi_2+x_2(k)\\
z_5(k)+z_6(k)+\psi_3+\psi_4 \ge 0 \rightarrow HC_{\mathrm{raw-c}}(k)\\
\qquad = z_5(k)+z_6(k)+\psi_3+\psi_4+\max(4000-x_4(k),\ 800)
\end{cases}
\qquad (28)
$$

$$\mathbf{y}_{\mathrm{d}}=\begin{bmatrix} T_{\mathrm{d}}(k)+\psi_1+\psi_2+x_2(k) \\ HC_{\mathrm{d}}(k)+\psi_3+\psi_4+\max(4000-x_4(k),\ 800) \end{bmatrix} \tag{29}$$

2nd schedule: ($\nu_1(k-1)\geq 5$)

$$\begin{cases} z_1(k)+z_3(k)+\psi_1+\psi_2\geq 0\rightarrow T_{\mathrm{exh}}(k) \\ \quad =z_1(k)+z_3(k)+\psi_1+\psi_2+x_2(k) \\ z_5(k)+z_6(k)+\psi_3+\psi_4<0\rightarrow HC_{\mathrm{raw-c}}(k) \\ \quad =\max(4000-x_4(k),\ 800) \end{cases} \tag{30}$$

$$\mathbf{y}_{\mathrm{d}}=\begin{bmatrix} T_{\mathrm{d}}(k)+\psi_1+\psi_2+x_2(k) \\ HC_{\mathrm{d}}(k)+\max(4000-x_4(k),\ 800) \end{bmatrix} \tag{31}$$

3rd schedule: ($\nu_1(k-1)\geq 5$)

$$\begin{cases} z_1(k)+z_3(k)+\psi_1+\psi_2<0\rightarrow T_{\mathrm{exh}}(k)=x_2(k) \\ z_5(k)+z_6(k)+\psi_3+\psi_4\geq 0\rightarrow HC_{\mathrm{raw-c}}(k) \\ \quad =z_5(k)+z_6(k)+\psi_3+\psi_4+\max(4000-x_4(k),\ 800) \end{cases} \tag{32}$$

$$\mathbf{y}_{\mathrm{d}}=\begin{bmatrix} T_{\mathrm{d}}(k)+x_2(k) \\ HC_{\mathrm{d}}(k)+\psi_3+\psi_4+\max(4000-x_4(k),\ 800) \end{bmatrix} \tag{33}$$

4th schedule: ($\nu_1(k-1)\geq 5$)

$$\begin{cases} z_1(k)+z_3(k)+\psi_1+\psi_2<0\rightarrow T_{\mathrm{exh}}(k)=x_2(k) \\ z_5(k)+z_6(k)+\psi_3+\psi_4<0\rightarrow HC_{\mathrm{raw-c}}(k) \\ \quad =\max(4000-x_4(k),\ 800) \end{cases} \tag{34}$$

$$\mathbf{y}_{\mathrm{d}}=\begin{bmatrix} T_{\mathrm{d}}(k)+x_2(k) \\ HC_{\mathrm{d}}(k)+\psi_3+\psi_4+\max(4000-x_4(k),\ 800) \end{bmatrix} \tag{35}$$

5th schedule: ($\nu_1(k-1)<5$)

$$\begin{cases} z_1(k)+z_3(k)+\psi_1+\psi_2\geq 0\rightarrow T_{\mathrm{exh}}(k) \\ \quad =z_1(k)+z_3(k)+\psi_1+\psi_2+x_2(k) \\ z_5(k)+z_6(k)+\psi_3\geq 0\rightarrow HC_{\mathrm{raw-c}}(k) \\ \quad =z_5(k)+z_6(k)+\psi_3+\max(4000-x_4(k),\ 800) \end{cases} \tag{36}$$

$$\mathbf{y}_{\mathrm{d}}=\begin{bmatrix} T_{\mathrm{d}}(k)+\psi_1+\psi_2+x_2(k) \\ HC_{\mathrm{d}}(k)+\psi_3+\max(4000-x_4(k),\ 800) \end{bmatrix} \tag{37}$$

6th schedule: ($\nu_1(k-1)<5$)

$$\begin{cases} z_1(k)+z_3(k)+\psi_1+\psi_2\geq 0\rightarrow T_{\mathrm{exh}}(k) \\ \quad =z_1(k)+z_3(k)+\psi_1+\psi_2+x_2(k) \\ z_5(k)+z_6(k)+\psi_3<0\rightarrow HC_{\mathrm{raw-c}}(k) \\ \quad =\max(4000-x_4(k),\ 800) \end{cases} \tag{38}$$

$$\mathbf{y}_{\mathrm{d}}=\begin{bmatrix} T_{\mathrm{d}}(k)+\psi_1+\psi_2+x_2(k) \\ HC_{\mathrm{d}}(k)+\max(4000-x_4(k),\ 800) \end{bmatrix} \tag{39}$$

7th schedule: ($\nu_1(k-1)<5$)

$$\begin{cases} z_1(k)+z_3(k)+\psi_1+\psi_2<0\rightarrow T_{\mathrm{exh}}(k)=x_2(k) \\ z_5(k)+z_6(k)+\psi_3\geq 0\rightarrow HC_{\mathrm{raw-c}}(k) \\ \quad =z_5(k)+z_6(k)+\psi_3+\max(4000-x_4(k),\ 800) \end{cases} \tag{40}$$

$$\mathbf{y}_{\mathrm{d}}=\begin{bmatrix} T_{\mathrm{d}}(k)+x_2(k) \\ HC_{\mathrm{d}}(k)+\psi_3+\max(4000-x_4(k),\ 800) \end{bmatrix} \tag{41}$$

8th schedule: ($\nu_1(k-1)<5$)

$$\begin{cases} z_1(k)+z_3(k)+\psi_1+\psi_2<0\rightarrow T_{\mathrm{exh}}(k)=x_2(k) \\ z_5(k)+z_6(k)+\psi_3<0\rightarrow HC_{\mathrm{raw-c}}(k) \\ \quad =\max(4000-x_4(k),800) \end{cases} \tag{42}$$

$$\mathbf{y}_{\mathrm{d}}=\begin{bmatrix} T_{\mathrm{d}}(k)+x_2(k) \\ HC_{\mathrm{d}}(k)+\max(4000-x_4(k),\ 800) \end{bmatrix} \tag{43}$$

It is worth noting that the values of x_2 (k) and x_4 (k) are known at each time step as they depend on the predefined controlling input u_3 (k).

Having the above mathematical formulations enables us proceed with the step-wise implementation of GS-LQTS, which is presented in the next subsection.

3.3. GS-LQTS Implementation

The pseudo-code of the implemented GS-LQTS is given below;

***Step 1*:** Set the initial values, namely the number of updating points (N) of 200, time difference (δt) of 0.25 *sec*, and the initial values of control inputs and system's states;

$$\nu(0)=\begin{bmatrix}2\\2\end{bmatrix};\ \begin{bmatrix} z_1(0) \\ x_2(0) \\ z_3(0) \\ x_4(0) \\ z_5(0) \\ z_6(0) \end{bmatrix}=\begin{bmatrix} \frac{T_{\mathrm{cat0}}}{3k_{\mathrm{cat}}}-\frac{50}{k_1} & \frac{T_{\mathrm{cat0}}}{3k_{\mathrm{cat}}} & \frac{T_{\mathrm{cat0}}}{3k_{\mathrm{cat}}}-\frac{1.3}{k_1} & 0 & -\frac{1.3}{k_1} & -\frac{1.3}{k_1} \end{bmatrix}^{\mathrm{T}} \tag{44}$$

***Step 2*:** For each of the eight schedules, calculate **P**, **g**, **L**, and $\mathbf{L}_{\mathbf{g}}$ based on the equations given before. These values are determined in an offline fashion, and then, employed for generating the controlling commands in real-time.

***Step 3*:** At this stage, the gains derived from the offline calculations are used for the real-time control of the engine during the coldstart period. The main point is that, at each time step, using the feedback of the engine system's states, the operating conditions given in the previous subsection are checked, and then, the corresponding gains are picked for calculating the controlling commands.

The flowchart of GS-LQTS is presented in Figure 3.

4. CMA-ES FOR TUNING GS-LQTS

The covariance matrix adaption evolutionary strategy (CMA-ES) is a local search version of (m, λ) evolutionary strategy (Hansen and Ostermeier, 2001). CMA-ES uses a population of m parents to generate λ off-springs ($m < \lambda$). The algorithm proceeds by selecting the best m of these λ off-springs. The strategy transacts until a stopping criterion is satisfied. By exerting such a procedure, the user expects

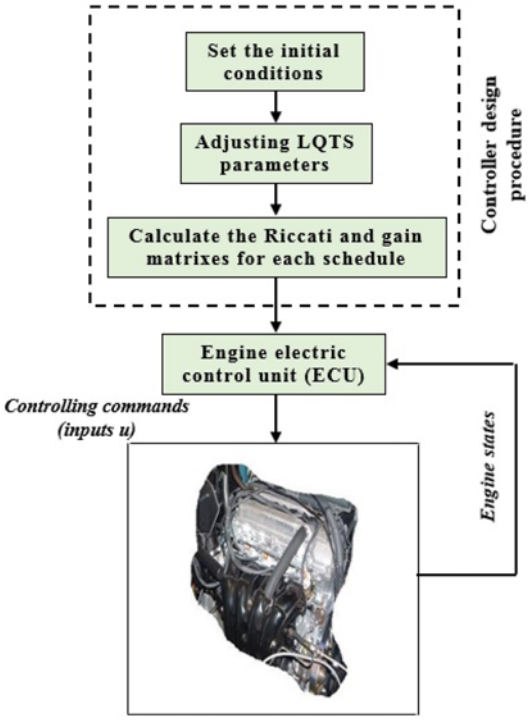

Figure 3. Flowchart of GS-LQTS.

to find a robust optimum solution within the solution domain. The optimization is fulfilled through the following steps.

Consider the objective function to be $f: S \subset R^n \rightarrow R$, generate a multivariate Gaussian distribution as;

$$\kappa(\theta; m, L) = \frac{1}{\sqrt{(2\pi)^n |\det L|}} \exp\left(-\frac{1}{2}(\theta - m)^T L^{-1}(\theta - m)\right) \quad (45)$$

where θ is the solution, $m \times R^n$, m is the mean of the distribution, and L represents a $n \times n$ covariance matrix.

Produce the parent m and off-spring population λ. To determine the optimum numbers of off-springs, set λ and m to be equal to [4+3 log n] +1 and [λ/2] +1, respectively. The operator [θ] yields the integer port of θ.

Choose a point $m = m^{(t)} \in S \subset R^n$ in such a way that $m^{(t)} \notin \partial S$. Select the diagonal covariance matrix $L = L^{(t)}$ in such a way that its diagonal elements $L_{ii} = s_i$. Considering the L matrix to be $n \times n$, if $S = [a_1, b_1] \times [a_2, b_2] \times \cdots \times [a_n, b_n]$, set the diagonal elements to be equal to $s_i = ((b - a)/3)^2$. It should be noted that t represents the iteration number here. Consider Θ to be the eigenvalue vector of L, and let D be a diagonal matrix where $\Delta_u = \delta_i^2$, and $\delta_i \neq 0$ are the eigenvalues of covariance matrix, then we have $L = \Theta\Delta^2\Theta^T$. Calculate the position of the off-springs by producing λ points $y_i^{(t+1)} \in S$, $i = 1, \ldots, \lambda$ with Gaussian distribution $\kappa(\theta; m, L)$. The mathematical expression of the semi-deterministic walk will be as given below;

$$\theta_i^{(t+1)} = m^{(t)} + \sigma^{(t)} y_i^{(t+1)} \quad (46)$$

where σ is equal to 1.

After producing the off-springs, select the best μ ones. By using the following equations, calculate the next mean value ($m^{(t+1)}$) and unbiased estimator ($L^{(t+1)}$) of the covariance matrix;

$$\langle y^{(t)} \rangle_W = \sum_{i=1}^{\mu} \omega_i y_{i\lambda}^{(t)} \quad (47)$$

Table 2. Mathematical formulations used within CMA-ES.

(1): $\mu_{eff} = \left(\sum_{i=1}^{\mu} \omega_i^2\right)^{-1}$

(2): $c_c = \frac{4 + \frac{\mu_{eff}}{n}}{n + 4 + 2\frac{\mu_{eff}}{n}}$

(3): $c_\sigma = \frac{\mu_{eff} + 2}{n + \mu_{eff} + 5}$

(4): $d_\sigma = 1 + 2\max\left(0, \sqrt{\frac{\mu_{eff} - 1}{n+1}} - 1\right) + c_\sigma$

(5): $c_1 = \frac{2}{(n+1.3)^2 + \mu_{eff}}$

(6): $c_\mu = \min\left(1 - c_1, 2\frac{\mu_{eff} - 2 + 1/\mu_{eff}}{(n+2)^2 + \mu_{eff}}\right)$

(7): $\tilde{E} = E\|\kappa(0; 0, I_n)\| = 1.46\Gamma\left(\frac{n+1}{2}\right)/\Gamma\left(\frac{n}{2}\right)$

(8): $h_\sigma^{(k)} = \begin{cases} 1 & if \ \frac{\|p_\sigma^{(k)}\|}{\sqrt{1-(1-c)^{2(k+1)}}} < \left(1.4 + \frac{2}{n+1}\right)\tilde{E} \\ 0 & otherwise \end{cases}$

(9): $\delta(h_\sigma^{(k)}) = (1 - h_\sigma^{(k)}) c_c (2 - c_c)$

(10): $(L^{(k)})^{(-1/2)} = \Theta\Delta^2\Theta^T$

$$m^{(t)} = m^{(t+1)} + \sigma^{(t-1)} \left\langle y^{(t)} \right\rangle_W = \sum_{i=1}^{\mu} \omega_i \theta_{i:\lambda}^{(t)} \tag{48}$$

$$p_\sigma^{(t)} = (1-L_\sigma) p_\sigma^{(t-1)} + \sqrt{L_\sigma (2-L_\sigma) \mu_{eff}} (L^{(t-1)})^{-1/2} \left\langle y^{(t)} \right\rangle_W \tag{49}$$

$$\sigma^{(t)} = \sigma^{(t-1)} \times \exp\left(\frac{L_\sigma}{d_\sigma} \left(\frac{\left\| p_\sigma^{(t)} \right\|}{\tilde{E}} - 1 \right) \right) \tag{50}$$

$$p_\sigma^{(t)} = (1-L_\sigma) p_\sigma^{(t-1)} + h_\sigma^{(t)} \sqrt{L_c (2-L_c) \mu_{eff}} \left\langle y^{(t)} \right\rangle_W \tag{51}$$

$$\begin{aligned} L^{(t)} = {} & (1-c_1-c_\mu) L^{(t-1)} + c_1 (p_\sigma^{(t)} (p_\sigma^{(t)})^T + \delta(h_\sigma^{(t)}) L^{(t-1)}) \\ & + c_\mu \sum_{i=1}^{\mu} \omega_i y_{i:\lambda}^{(t)} (y_{i:\lambda}^{(t)})^T \end{aligned} \tag{52}$$

$$\omega_i = \frac{\log(0.5(\mu+1))) - \log(i)}{\sum_{i=1}^{\mu} (\log(0.5(\mu+1))) - \log(i))} \tag{53}$$

The formulations of the symbols used in the above equations are listed in Table 2.

In this study, CMA-ES is used to optimally tune the adjustable parameters of GS-LQTS. In general, the flexible values of GS-LQTS (which are often determined heuristically) have a remarkable impact on the performance of GS-LQTS. It is obvious that to form the formulation of GS-LQTS, the values of **Q** and **F** matrixes should be positive semi-definite, and the arrays of **R** matrix should be positive definite. By considering the following constraints for the elements of each of these matrixes;

$$\begin{cases} 0 \le r_1,\ r_2,\ r_3,\ r_4 \le 1 \\ 0 \le f_1,\ f_2,\ f_3,\ f_4 \le 10 \\ 0 \le q_1,\ q_2,\ q_3,\ q_4 \le 1 \end{cases} \tag{54}$$

CMA-ES determines the values of weighting matrixes **Q**, **F**, and **R** such that the best performance is achieved. The CMA-ES objective function used for determining the weighting matrixes of GS-LQTS is the amount of cumulative *HC* emissions over the coldstart period (which should be minimized), as follows;

$$Obj = \sum_{i=k_0}^{k_f} (1-\hat{\eta}(i)) \dot{m}_{exh}(i) \left(\frac{16}{28.5} \times 10^{-6} \right) \hat{HC}_{raw-c}(i) \tag{55}$$

5. RESULTS AND DISCUSSIONS

In this study, the concept of immune-inspired artificial systems is used for developing the real-time optimum SOC trajectory builder. In addition, the concept is considered for reducing the complexity of computations by clustering the possible speeds into a finite number of groups.

This section is given into two subsections. Firstly, the parameter settings required for the numerical experiments are presented. Thereafter, the simulation results are given.

5.1. Parameter Settings

To proceed with the simulations, a set of algorithmic parameters should be set. As it was mentioned previously, three different controllers, namely the standard gain scheduled linear quadratic tracking system (GS-LQTS) (Naidu, 2003), CMA-ES based GS-LQTS, and an optimal controller based on the Pontryagin's minimum principle (PMP) (Azad *et al*., 2012), are taken into account. For the standard GS-LQTS, the following values are considered for the weighting matrixes. For the first four schedules;

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 0.001 \end{bmatrix};\ \mathbf{F} = \begin{bmatrix} 4 & 0 \\ 0 & 0.01 \end{bmatrix};\ \mathbf{R} = \begin{bmatrix} 0.5 & 0 \\ 0 & 1 \end{bmatrix};$$

$$\mathbf{W} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 0.001 \\ 0 & 0.001 \end{bmatrix};\ \mathbf{V} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0.001 & 0.001 \\ 0 & 0 & 0.001 & 0.001 \end{bmatrix} \tag{56}$$

Also, for the last four schedules;

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 0.02 \end{bmatrix};\ \mathbf{F} = \begin{bmatrix} 4 & 0 \\ 0 & 0.01 \end{bmatrix};\ \mathbf{R} = \begin{bmatrix} 0.4 & 0 \\ 0 & 1 \end{bmatrix};\ \mathbf{W} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 0.02 \\ 0 & 0.02 \end{bmatrix};$$

$$\mathbf{V} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0.02 & 0.02 \\ 0 & 0 & 0.02 & 0.02 \end{bmatrix} \tag{57}$$

The mathematical formulation used for the implementation of PMP is given in Appendix A. To evaluate the efficiency of CMA-ES algorithm, two rival optimization techniques known as golden sectioning strategy (GSS) (Kiefer, 1953) and particle swarm optimization (PSO) (Van den Bergh and Engelbretch, 2006), are also taken into account. All of the three optimization techniques are encoded in the Matlab software. The stopping condition for all of the optimization methods is the number of iterations, which is equal to 100. For the CMA-ES algorithm, 10 chromosomes are taken into account. For the PSO algorithm, 10 particles are taken into account. The social and cognitive coefficients are set to be 2. The inertia weight parameter is equal to 1.3. To suppress the undesired effects of uncertainty associated with the stochastic instinct of the optimization algorithms, here, the hyper-level optimization of GS-LQTS is performed in 10 independent runs and the average results are reported.

To endorse the generalization capability of the proposed control strategy, three different working scenarios of the engine at high speed, medium speed, and low speed (Mozaffari and Azad, 2014) are considered. As mentioned before, the engine speed profile ($u_3 = \varpi_e$) is treated as a predefined input signal. The considered engine speed profiles are indicated in Figure 4.

The simulations are done using the Matlab software on a

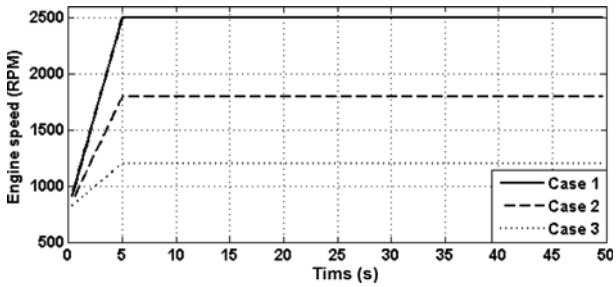

Figure 4. Predefined engine speed profiles during the coldstart operation.

Windows 7 operating system on a PC with Pentium IV, Intel core i7 CPU and GBs RAM.

5.2. Simulation Results

In this section, three important aspects are investigated. Firstly, the authors evaluate the efficacy of CMA-ES, GSS and PSO for optimizing the weighting matrixes of GS-LQTS. In the second stage, a comparative study is performed to endorse the efficacy of the proposed controller versus the PMP-based controller and the standard GS-LQTS controller. Finally, the impact of the proposed controller on the behavior of the engine during the coldstart period is studied.

CMA-ES is a population-based optimization algorithm which uses stochastic searching strategies for exploration/exploitation of the solution landscape. Therefore, it is important to evaluate the performance of the optimization algorithm over independent runs to find out whether the

Table 3. Performance of CMA-ES for the tuning of GS-LQTS over 10 independent runs.

Run	Case 1		Case 2		Case 3	
	Time (s)	HC_{cum} (g)	Time (s)	HC_{cum} (g)	Time (s)	HC_{cum} (g)
1	5.2434	0.1932	5.7463	0.1974	5.53653	0.2174
2	5.3126	0.1941	5.5126	0.1987	5.17354	0.2168
3	5.3196	0.1923	5.6623	0.1978	5.25367	0.2156
4	5.4423	0.1918	5.7237	0.1981	5.61523	0.2173
5	5.5152	0.1902	5.8843	0.1983	5.77465	0.2163
6	5.6473	0.1942	5.9283	0.1986	5.88273	0.2175
7	5.4983	0.1933	5.8847	0.1983	5.15325	0.2182
8	5.5663	0.1941	5.8646	0.1979	5.25434	0.2187
9	5.7744	0.1952	5.9948	0.1993	5.38847	0.2191
10	5.9384	0.1911	5.9126	0.1985	5.44736	0.2185

Table 4. Comparisons between the optimization algorithms for the tuning of GS-LQTS.

Algorithms		Case 1		Case 2		Case 3	
		Time (s)	HC_{cum} (g)	Time (s)	HC_{cum} (g)	Time (s)	HC_{cum} (g)
CMA-ES	Best	5.2434	0.1902	5.5126	0.1974	5.1532	0.2156
	Worst	5.9384	0.1952	5.9948	0.1993	5.8827	0.2191
	Mean	5.5258	0.1930	5.8114	0.1983	5.4480	0.2175
	Std.	0.2171	0.0016	0.1470	0.0005	0.2522	0.0011
GSS	Best	4.0172	0.1936	4.2322	0.1982	4.3524	0.2242
	Worst	4.5635	0.1948	4.6635	0.1991	4.8162	0.2283
	Mean	4.2994	0.1942	4.5190	0.1986	4.6048	0.2257
	Std.	0.2279	0.0007	0.1963	0.0004	0.2113	0.0018
PSO	Best	6.8836	0.1933	6.8837	0.1978	6.6114	0.2187
	Worst	7.3416	0.1942	7.1623	0.1989	7.3243	0.2213
	Mean	7.0563	0.1937	7.0005	0.1984	6.8991	0.2198
	Std.	0.1984	0.0004	0.1200	0.0005	0.3141	0.0011

obtained solutions are the same for independent optimization procedures. Table 3 lists the results obtained by CMA-ES over 10 independent runs with respect to the calculation time and HC_{cum} for all of the three case studies. By taking a precise look into the obtained results, it can be inferred that the performances of CMA-ES over the independent runs are close to each other. It seems that the computational time required for the calculations are around 5 ~ 6 seconds for all of the optimization cases. Furthermore, the results indicate that the minimum HC_{cum} is around 0.193, 0.198, and 0.217 g for *Case 1*, *Case 2*, and *Case 3*, respectively.

To further evaluate the performance of CMA-ES, a comparative study is carried out by using PSO and GSS for the same optimization problem. Table 4 lists the statistical results of the comparative simulations. As it can be seen, the mean value obtained by CMA-ES is better than the other rival optimization approaches. The results also indicate that PSO can provide acceptable solutions and outperform GSS. Also, the computational time of CMA-ES is a bit less than PSO, but slightly greater than GSS. One of the other considerations is the accuracy of the optimization algorithms. This can be evaluated by checking the obtained *std.* values for the three cases. Based on the obtained results, it can be inferred that the optimization techniques are close to each other in terms of this aspect. Table 5 lists the optimum values of GS-LQTS weighting coefficients.

After evaluating the performance of CMA-ES for the tuning of GS-LQTS weighting factors, the authors want to compare the performance of the proposed coldstart control method with the standard GS-LQTS and PMP-based controllers. Table 6 lists the minimum HC_{cum} obtained by the rival controlling methods for the three case studies. The results indicate that the GS-LQTS optimized by CMA-ES has a remarkably better performance compared to the standard GS-LQTS method. The simulations also indicate that PMP results in the smallest values of HC_{cum}, except for the third case that the performance of GS-LQTS optimized by CMA-ES and PMP are very close to each other. However, PMP is an open-loop controller and requires a significant amount of time to calculate the optimum controlling commands, which makes its real-time implementation very difficult. Given the fact that GS-LQTS is a real-time, closed-loop controller that employs the engine's states feedback for the calculation of control inputs, one can easily interpret that it is much more useful for the coldstart control process. This is mainly due to the fact that it can be easily implemented for real-time applications on the engine's control unit with very low processing power requirements, and at the same time, it can alleviate (up to some degree) the effects of model uncertainties and unknown disturbances because of using the feedback of the engine's states. Thus, the proposed controller has a higher level of robustness, as compared to the PMP algorithm.

Table 5. Optimum values of the GS-LQTS weighting coefficients derived by CMA-ES.

r_1	r_2	r_3	r_4
0.612	1	0.434	0.946
q_1	q_2	q_3	q_4
0.927	0.008	0.911	0.007
f_1	f_2	f_3	f_4
6	0.006	4	0.003

Table 6. Results obtained by the CMA-GS-LQTS, GS-LQTS and PMP-based controllers.

Controller	Case 1 HC_{cum} (g)	Case 2 HC_{cum} (g)	Case 3 HC_{cum} (g)
PMP	0.145	0.158	0.218
GS-LQTS	0.248	0.255	0.267
CMA-GS-LQTS	0.193	0.198	0.217

After evaluating the potential of GS-LQTS for the coldstart control problem, the authors would like to analyze its performance for controlling the considered SI engine during the coldstart process. The analytically derived control signals for Δ and *AFR* are depicted in Figure 5. The important point is that the both control commands have finally reached the reference values (namely, 50 for u_1 and 14.7 for *AFR*). This, in turn, implies that the remedy of changing variables from u to *v* is effective. Furthermore, it can be observed that the values of *AFR* signals are always confined within the considered range of 10 to 16, and therefore, no infeasible solution is achieved. Now, the authors intend to investigate the impact of the calculated commands on the physical behavior of the considered

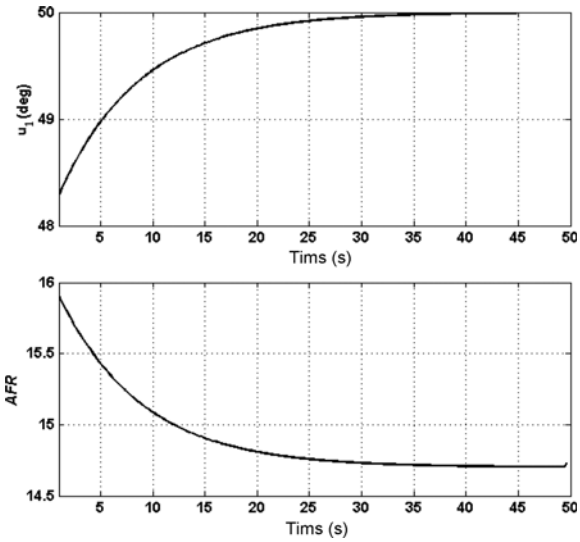

Figure 5. Calculated optimal control inputs.

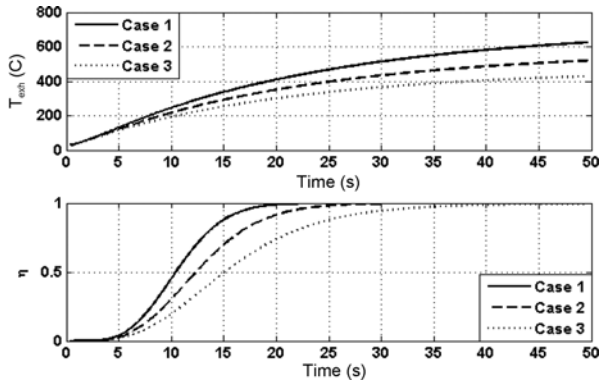

Figure 6. Optimum profiles for T_{exh}, HC_{raw}, and η for the three considered cases.

engine. Figure 6 depicts the profiles obtained for T_{exh}, and η. These results indicate that by increasing the engine speed, the final value of the exhaust gas temperature increases. This is equivalent to the fact that increasing the engine speed causes much more heat for warming up the catalyst. This, in turn, increases the conversion efficiency of the catalytic converter. Also, by observing the obtained η profiles, one can easily infer that the conversion efficiency of *Case 1* reaches the value of 1 faster than the other cases. To be more to the point, it can be seen that the time required for reaching the nominal efficiency values are 22, 28 and 38 seconds for *Case 1*, *Case 2*, and *Case 3*, respectively. Also, Figure 7 indicates the rate of engine-out *HC* emissions. The same peaks can be seen for these HC_{rate} profiles. A physical analysis shows this initial peak is, indeed, mandatory for decreasing the possibility of the engine stalling during the coldstart operation. Finally, the cumulative *HC* emission profiles are depicted in the lower sub-figure of Figure 7. The cumulating of emitted *HCs* occurs approximately within the first 10 to 15 seconds of

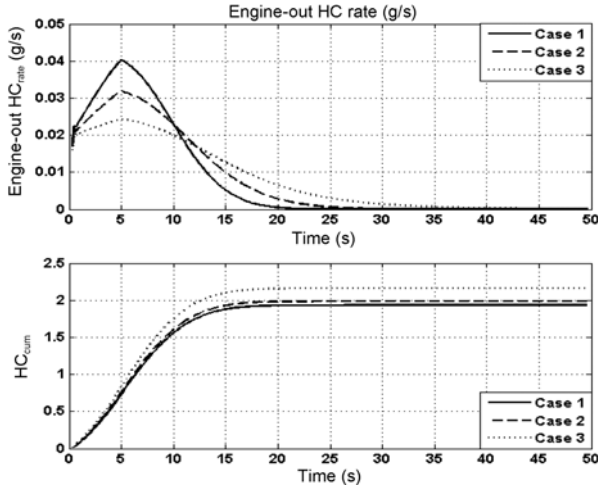

Figure 7. Optimum profiles for HC_{rate}, and also, HC_{cum} for the three considered cases.

the engine's operation. Furthermore, it is observed that HC_{cum} or the total tailpipe *HC* emissions (after passing through the catalytic converter) for Case 1 is the lowest (which has the highest engine speed).

All to all, the simulation results indicate that GS-LQTS shows a promising performance for the real-time control of the engine behavior over the coldstart period. Besides, the outcomes indicate that the use of the proposed higher level scheduled mechanism allows converting the nonlinear control-oriented model to a set of the linear models, which can be employed for the calculation of control inputs through some independent control laws.

6. CONCLUSION

In this study, a gain scheduled linear quadratic tracking system (GS-LQTS) was developed for reducing the cumulative hydrocarbon emissions (HC_{cum}) for an automotive engine over the coldstart period. To make sure that the controller offers its best performance for the considered problem, the covariance matrix adaption evolutionary strategy (CMA-ES) was adopted, and the weighting coefficients of GS-LQTS were tuned heuristically. The results of the conducted simulations indicated that the proposed controller shows an acceptable performance for controlling the engine over the coldstart operation. It was observed that the prominent asset of the devised controller refers to its very fast computational speed which facilitates its implementation in the engine's control units with lower processing powers, while it also considers the feedback from the engine's states to calculate the control commands which improves the level of robustness.

To endorse the acceptable performance of GS-LQTS, a set of comparative studies were carried out using the standard LQTS (without CMA-ES) and an optimal controller based on the Pontryagin's minimum principle (PMP). The results of the comparative studies indicate that the use of CMA-ES for an optimal tuning of the weighting coefficients of GS-LQTS improves its control performance. Furthermore, it was observed that the proposed control strategy shows acceptable results compared to PMP. Given the fact that the control commands of PMP are calculated in an offline fashion, and also it is an open-loop control approach, it can be concluded that the CMA-ES based GS-LQTS controller has an obvious advantage as its control commands can be easily calculated in real-time and it is a feedback control system, as well. Furthermore, the findings demonstrated that by taking advantage of a gain scheduled strategy, LQTS could cope with the nonlinearities associated with the engine behaviour over the coldstart period.

REFERENCES

Aquino, C. F. (1981). Transient A/F control characteristics of the 5 liter central fuel injection engine. *SAE Paper No*. 810494.

Azad, N. L., Sanketi, P. R. and Hedrick, J. K. (2012). Determining model accuracy requirements for automotive engine coldstart hydrocarbon emissions control. *ASME J. Dynamic Systems Measurement and Control* **134**, **5**, 051002.

Hansen, N. and Ostermeier, A. (2001). Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation* **9**, **2**, 159–195.

Kiefer, J. (1953). Sequntial minimax search for a maximum. *Proc. American Mathematics Society*, **4**, 502–506.

Mozaffari, A. and Azad, N. L. (2014). Optimally pruned extreme learning machine with ensemble of regularization techniques and negative correlation penalty applied to automotive engine coldstart hydrocarbon emission identification. *Neurocomputing*, **131**, 143–156.

Mozaffari, A., Azad, N. L. and Hedrick, J. K. (2015). A nonlinear model predictive controller with multiagent online optimizer for automotive coldstart hydrocarbon emissions reduction. *IEEE Trans. Vehicular Technology*, **99**, 1.

Naidu, D. S. (2003). *Optimal Control Systems*. CRC Press. Boca Raton, Florida, USA.

Rasul, M. and Glasgow, R. (2005). Performance improvement of an internal combustion engine. *6th Int. Conf. Mechanical Engineering*, Dhaka, Bangladesh.

Salehi, R., Shahbakhti, M. and Hedrick, J. K. (2014). Real -time hybrid switching control of automotive cold start hydrocarbon emission. *J. Dynamics Systems and Control*, **136**, 041002-1.

Sanketi, P. R., Zavala, J. C. and Hedrick, J. K. (2006). Automotive engine hybrid modeling and control for reduction of hydrocarbon emissions. *Int. J. Control* **79**, **5**, 449–464.

Sanketi, P. R., Zavala, J. C., Wilcutts, M., Kaga, T. and Hedrick, J. K. (2007). MIMO control for automotive coldstart. *5th IFAC Symp. Advances in Automotive Control*.

Sanketi, P. R. (2009). *Coldstart Modeling and Optimal Control Design for Automotive SI Engines*. Ph. D. Dissertation. University of California. Berkeley, USA.

Shaw, B. and Hedrick, J. K. (2003). Closed-loop engine coldstart control to reduce hydrocarbon emissions. *American Control Conf.*, 1392–1397.

Sun, J. and Sivashankar, N. (1999). Issues in cold start emission control for automotive ICE engines. *Proc. American Control Conf.*, 1372–1376.

Tseng, T. C. and Cheng, W. K. (1999). An adaptive air/fuel ratio controller for SI engine throttle transients. *SAE Paper No*. 1999-01-0552.

Van den Bergh, F. and Engelbretch, A. P. (2006). A study of particle swarm optimization particle trajectories. *Information Sciences* **176**, **8**, 937–971.

Zavala, J. C. (2007). *Engine Modeling and Control for Minimization of Hydrocarbon Coldstart Emissions in SI Engine*. Ph. D. Dissertation. University of California. Berkeley, USA.

APPENDIX

A. SOLUTION BASED ON PONTRYAGIN'S MINIMUM PRINCIPLE (PMP)

The considered classical optimal controller in this study employs the PMP algorithm to find the optimum control inputs (Azad *et al*., 2012). In this section, the authors provide the formulation used for the development of the PMP-based controller. Consider the state-space formulation given for the coldstart problem, together with the initial condition given in Equation (44). PMP calculates the optimized profiles of the input signals, that is, $\boldsymbol{\upsilon}^*$, such that the objective function below is minimized;

$$\hat{J} = \int_{t_0}^{t_f} \mathbf{L}(\chi(t),\ \upsilon(t),\ \mathrm{t}) \cdot dt = \int_{0}^{T} (1-\eta)\dot{m}_{\mathrm{exh}} \left(\frac{16}{28.5} \times 10^{-6} \right) HC_{\mathrm{raw-c}} \cdot dt \tag{A.1}$$

First, the Hamiltonian function $H(t)$ for the above objective function is defined by;

$$H(t) = \mathbf{L}(\chi(t),\ \upsilon(t),\ \mathrm{t}) + \mathbf{p}^{\mathrm{T}}(t)\mathbf{f}(\chi(t),\ \upsilon(t),\ \phi_\chi,\ t) \tag{A.2}$$

Then, the states of the system are given by;

$$\frac{\partial H}{\partial \mathbf{p}} = \dot{X} = \mathbf{f}(\chi(t),\ \upsilon(t),\ \phi_\chi,\ t), \quad \chi(t_0) = \chi_0 \tag{A.3}$$

The adjoint variables $\mathbf{p}(t)$ can be then determined through a backward integration, as follows;

$$\dot{\mathbf{p}}(t) = \frac{\partial H}{\partial X}, \quad \mathbf{p}(t_{\mathrm{f}}) = \mathbf{p}_{\mathrm{f}} \tag{A.4}$$

The optimal solution must satisfy the following criterion;

$$H[\chi(t),\ \upsilon^*(t),\ t] \le H[\chi(t),\ \upsilon(t),\ \mathbf{p}(t),\ t] \tag{A.5}$$

The steepest descend method is used to calculate the optimum inputs ($\boldsymbol{\upsilon}^*$) by means of the following updating rule;

$$\upsilon^{(\mathrm{i}+1)}(t) = \upsilon^{(\mathrm{i})}(t) - \mathbf{K}(t)\left(\frac{\partial H}{\partial \upsilon} \right) \tag{A.6}$$

If the change in the value of the objective function between two sequential iterations becomes very small ($\hat{J}^{(\mathrm{i}+1)} \approx \hat{J}^{(i)}$), then, the iterative procedure is terminated, and also: $\upsilon^*(t) = \upsilon^{(\mathrm{i})}(t)$.