Springer

KSME

Journal of Mechanical Science and Technology 33 (11) (2019) 5483~5498
www.springerlink.com/content/1738-494x(Print)/1976-3824(Online)
DOI 10.1007/s12206-019-0847-3

Comparison of model predictive controller and optimized min-max algorithm for turbofan engine fuel control †

Morteza Montazeri-Gh* and Ali Rasti

Systems Simulation and Control Laboratory, School of Mechanical Engineering, Iran University of Science and Technology (IUST), Tehran, Iran

(Manuscript Received October 5, 2018; Revised March 8, 2019; Accepted May 16, 2019)

Abstract

Min-max selector structure is traditionally used as the industrial control architecture of commercial turbofan engines. However, recent studies indicate that this structure with linear compensators suffers from lack of safety guarantee in fast demands. On the other hand, model predictive control (MPC) technique, which incorporates input/output constraints in its optimization process, has the potential to fulfill the control requirements of an aircraft engine. In this paper, a practical approach is performed for design and optimization of the turbofan engine controller through a comparative study where all control modes and requirements have been taken into account simultaneously. For this purpose, a thermodynamic nonlinear model is firstly developed for the turbofan engine. The linear regulators of min-max structure are then optimized via genetic algorithm (GA). The MPC technique is formulated based on the proper discrete-time linearized state-space models at desired operating points with real-time optimization, in which the MPC tuning horizons are obtained through GA optimization procedure. The both controllers are implemented on appropriate hardware taking the real-time aspects into account. Finally, a hardware in the loop (HIL) platform is developed for the turbofan engine electronic control unit (ECU) testing. The software and HIL simulation results confirm that MPC improves the response time of the system in comparison with min-max algorithm and guarantees the engine limit protection. This study demonstrates competitive advantages of MPC in terms of limit protection assurance and fast response, despite more computational burden.

Keywords: Turbofan engine; Thermodynamic model; Model predictive control; Optimal MPC horizons; Optimized min-max algorithm; HIL simulation

1. Introduction

Due to higher reliability and lower fuel consumption, turbofan engine is commonly used for commercial aircrafts [1]. Control system is one of the most critical and advanced technologies of turbofan engines. The objective of an aircraft engine control system is to provide the requested thrust and guarantee the safe operation of the engine. For this purpose, the aeroengine control system adjusts pressure ratio or fan speed as the thrust representative based on the requested throttle level, and also maintains the engine outputs within the permissible bounds at all times. In this way, fast engine response with limit protection assurance is the main challenge of engine control as this control system should fulfill fast thrust response in emergency maneuvers without limit violation [2].

A survey on aero-engine control techniques shows that various control methods have been used for engine fuel control [3-9]. Among them, min-max algorithm is traditionally used as the industrial control architecture of turbofan engines [10-13]. However, recent studies have shown that there is no guarantee for min-max algorithm with linear compensators to protect engine limits during transient state [14-16]. Imani and Montazeri-Gh investigated this issue in Ref. [16] and presented an analytical approach to guarantee the limit protection of min-max algorithm at one operating point. However, the controller provided a conservative response with a slow time response.

MPC is an advanced model-based controller which has attracted the attention of researchers in recent years for gas turbine engines [17, 18]. MPC considers input/output constraints in the production of control input signal while ensures the engine limit protection, which introduces it as a potential alternative approach for turbofan engines control [19]. Vroemen et al. [20] investigated the feasibility of MPC for a laboratory gas turbine installation. Mu et al. [21] designed an approximate model predictive control (AMPC), nonlinear MPC (NMPC) and PID gain-scheduling controller and compared the performance of these methods in engine fuel control. They designed MPC only with input and rate of input constraints and PID gain-scheduling controller with saturation constraints on the control input, without min-max struc-

*Corresponding author. Tel.: +98 9123071723, Fax.: +98 2177240258
E-mail address: montazeri@iust.ac.ir
† Recommended by Associate Editor Hanho Song

ture. DeCastro [22] developed MPC strategy for a fast response closed-loop control of turbine blade tip clearances to regulate them and optimally maintaining a predefined minimum clearance. Also, Kai et al. [23] utilized an active generalized predictive control (GPC) with auto-regressive (AR) error modification and fuzzy adjustment on control horizon for an aero-engine turbine blade tip clearance control. Richter [24] expressed the formulation of MPC based on a LTI state-space model of an aircraft engine, but only the limited number of required constraints of the engine were considered. Moreover, the optimization process was off-line and the MPC horizons were selected by trial and error. Richter et al. [25] proposed a multiplexed MPC approach using linear state-space model while focusing on the reduction of the computational burden of MPC.

Hardware implementation and HIL simulation are two main steps as a practical approach for the development of turbofan engine control system. There are several studies have been reported for the HIL simulation of the control system of gas turbine engines and aerospace vehicles [26-30]. Montazeri-Gh et al. [31] constructed a new HIL simulation for testing a single-spool turbojet engine fuel control system using multiple processors for multi-rate simulation. They set up an HIL framework consisting of an industrial PC for the engine model as the plant, commercial I/O board and an ECU with min-max control architecture. In another study, Montazeri-Gh et al. [32] presented an HIL simulation for a single-spool turbo-shaft engine's ECU using a simplified engine model. They utilized Wiener block structure method for modeling of the engine and the min-max fuel control algorithm was implemented on a PC/104 hardware. In these studies, min-max control algorithm was used for fuel control of single-spool turbojet and turbo-shaft engines. However, there is no published report about the comparison between these two controllers (MPC and min-max) both from design point of view and hardware implementation and real-time simulation.

In this paper, a practical approach is performed for design and optimization of the turbofan engine controller through a comparative study where all control modes and requirements have been taken into account simultaneously. For this purpose, an MPC with optimal tuning horizons and real-time optimization and a competitive min-max algorithm with optimized regulators are designed for a turbofan engine. In this way, all the challenging real-world constraints are translated into mathematical inequalities in the design process to ensure the safe and optimal operation for the engine. In addition, a nonlinear thermodynamic model of the engine is developed and its computer simulation is performed. To achieve an enhanced response of min-max controller, the linear compensators are obtained through genetic optimization algorithm. Moreover, MPC is designed based on the proper discrete-time linearized state-space model at several operating points in the flight envelope. Finally, hardware implementation and HIL testing are performed for both controllers, and the comparative study is carried out.

Table 1. Take-off specifications of the studied engine.

Parameter	Value
Bypass ratio	5.5
Overall pressure ratio	34
Max. thrust (kN)	134
Inlet air mass flow (kg/s)	427
Fuel mass flow (kg/s)	1.307
Inlet HPT temperature (K)	1580
LPS and HPS speeds (rpm)	5000, 14460

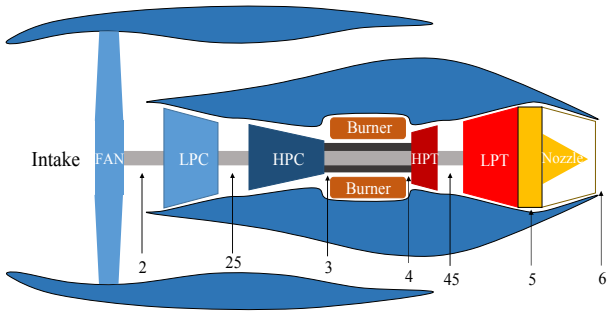


Fig. 1. Schematic of a two-spool high-bypass-ratio turbofan engine and its typical stations numbering.

2. The engine model

In this section, development of a real-time thermodynamic nonlinear turbofan engine model for the HIL test of the controllers is performed. Aircraft turbofan engines contain various main components characterized by nonlinear characteristic maps. These characteristic maps make the engine thermodynamic modeling complicated. In addition, engine computer simulation requires multiple iterative loops with heavy computational burden. Moreover, the thermodynamic model developed in this study is linearized for the MPC design.

A two-spool high-bypass unmixed flow turbofan engine is studied in this research. The components of the engine and the stations' numbering are defined in Fig. 1. Table 1 presents the take-off specifications of the engine. The engine transient model is a component level model (CLM) which is composed of two parts. The first part refers to a set of nonlinear algebraic equations that describe the thermodynamic relations in the engine components as follows:

Fan and compressors

$$\theta=\frac{T_{tin}}{T_{std}},\ \delta=\frac{P_{tin}}{P_{std}},\ N_c=\frac{N}{\sqrt{\theta}},\ w_{out}=\frac{w_{c,out}}{\sqrt{\theta}/\delta},\ BPR=\frac{w_{duct}}{w_{core}}$$
$$\left[w_{c,map},PR,\eta\right]=f_1\left(N_c,\beta line\right)$$
$$h_{tout}=\frac{h_{tout,ideal}-h_{tin}}{\eta}+h_{tin} \tag{1}$$
$$PR=\frac{P_{tout}}{P_{tin}},\ T_{tout}=f_2\left(P_{tout},h_{tout}\right)$$

$$PW = w_{out}\left(h_{tin} - h_{tout}\right).$$

Burner

$$\begin{aligned}
&\Delta P = P_{tout} - P_{tin}\\
&w_{out} = w_{in} + w_f\\
&h_{tout} = \frac{\eta_b w_f H + w_{in} h_{tin}}{w_{out}}.
\end{aligned} \tag{2}$$

Turbines

$$\begin{aligned}
&\theta = \frac{T_{tin}}{T_{std}},\ \delta = \frac{P_{tin}}{P_{std}},\ N_c = \frac{N}{\sqrt{\theta}},\ w_{out} = \frac{w_{c,out}}{\sqrt{\theta}/\delta}\\
&\left[w_{out}, PR, \eta\right] = f_3\left(N_c, \beta line\right)\\
&h_{tout_g} = h_{tin} - (h_{tin} - h_{tout_g,ideal})\eta\\
&PR = \frac{P_{tout}}{P_{tin}},\ T_{tout} = f_4\left(P_{tout}, h_{tout}\right)\\
&h_{tout} = \frac{1}{w_{out}}\left[h_{tin} w_{tin} + \eta(h_{tin} - h_{tout_g,ideal}) w_{in}\right]\\
&PW = w_{out}(h_{tin} - h_{tout_g}).
\end{aligned} \tag{3}$$

Convergence nozzles

$$\begin{aligned}
&PR = \frac{P_{tin}}{P_{amb}}\\
&P_s = g(h_{tin}, P_{tout}, M = 1)\\
&PR_{critical} = \frac{P_{tin}}{P_s}\\
&\text{If } PR > PR_{critical}\\
&\left[P_s, u\right] = g(h_{tin}, P_{tout}, M = 1)\\
&\text{else}\\
&\left[P_s, u\right] = g(h_{tin}, P_{tout}, P_s = P_{amb}).
\end{aligned} \tag{4}$$

The second part refers to nonlinear dynamic equations that describe transient process. Due to the complex geometry of the engine components and the complexity of gas flow, the dominant engine dynamics should be considered during controller design. The dynamics of the engine shafts are commonly considered as the necessary dynamics and play a key role in the engine transient behavior. The dynamic equations of the engine shafts are as follows:

$$\begin{aligned}
&\dot{N}_{HP} = \frac{PW_{HP}}{J_{HP} N_{HP}\left(\frac{2\pi}{60}\right)^2}\\
&\dot{N}_{LP} = \frac{PW_{LP}}{J_{LP} N_{LP}\left(\frac{2\pi}{60}\right)^2}.
\end{aligned} \tag{5}$$

The parameters used in Eqs. (1)-(5) are defined in the Nomenclature section.

Thus, aircraft engine dynamic model is based on the characteristic curves of the engine components and contain multiple loops with iterative numerical solutions [33, 34]. Generally, this procedure is achieved via matrix solution procedure and is commonly enhanced by Newton-Raphson iterative solving technique [35]. This procedure is simulated in Simulink/MATLAB environment using MATLAB functions. Therefore, the state equations of engine dynamics suitable for controller design are not available in closed form and the dynamic model is commonly linearized numerically. Therefore, the primary problem in controller design is to have a linear state-space model at considered operating points of the engine [36]. One of the approaches of extracting an accurate linear model from the thermodynamic nonlinear model is the application of system identification (SID) methods [37, 38]. The linearization is carried out using system identification toolbox of MATLAB software [37]. For this purpose, the input signal should be a persistent excitation in order to encompass all the input frequencies and amplitudes in the desired neighborhood of the operating point. The quasi amplitude-modulated pseudo random binary sequence (QAPRBS) signal is one of excitation signals suitable for identification of nonlinear systems. The generated QAPRBS signal is shown in Fig. 2. The hold time [39] is considered between 6 to 10 seconds and the signal amplitude range is in the interval of ±6 % of the mean value of the fuel flow about the desired operating point. Moreover, the ramp time is considered as 0.04 seconds for a signal from the minimum to the maximum range. This signal is applied to the thermodynamic model as input and the required outputs are collected.

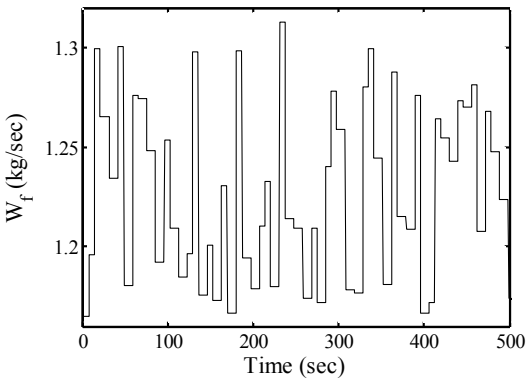


Fig. 2. The QAPRBS signal of the fuel input.

3. Turbofan engine control requirements

As mentioned earlier, the objective of the engine control system is to provide the requested thrust according to the pilot demand while protecting the engine from exceeding its physical bounds. Limit protection assurance of jet engines is one of the main challenges in the controller design.

A turbofan engine encounters several limits during its operation which should be protected with the control system. Turbofan engine stall or surge is a physical instability which may occur in a compressor due to a sudden fuel increase. An

abrupt fuel decrease may cause combustor flame-out and over-pressurization of the compressor discharge causes combustor blow-out. Other structural limitations including rotor over-speed and turbine over-temperature may occur due to excessive fuel injection. These detrimental factors create limitations for engine operation.

Each of these constraints can be satisfied by controlling one or more parameters. The maximum static pressure of HPC discharge (*Ps3*) limiter is used to prevent combustor over-pressurization. The *Ps3* minimum limit is used to provide stable engine operation at idle power. The fuel flow rate divided by static pressure of HPC discharge ($W_f/Ps3$) is commonly used as a popular control parameter which is called "ratio unit (*RU*)" limiter. The minimum ratio unit limiter protects the engine against combustor lean blow-out and LPC stall. A safe margin from HPC surge line is considered to prevent HPC stall. Although HPC stall margin has a decreasing trend during acceleration, it should be considered as a maximum limit. The maximum speed limiter of fan and core shafts are applied to ensure that over-speed of the engine shafts do not occur. Since HPT inlet temperature is not measurable, the over-temperature of the turbines is controlled via LPT exit temperature limiter. The safe operation of the engine depends on the protection of all these constraints at all times.

4. Engine fuel control system design

4.1 Min-max algorithm

The most common fuel control structure for aircraft engines is min-max selector scheme. The basis of this algorithm is to control a main output using a single control input while maintaining the other intended outputs within their limits. So, this structure contains multiple control loops. The accomplishment of this method is based on a selection strategy between the various control loops. This selection logic decides which control loop should be made active at any given time. Several linear control theories have been applied to design the single feedback control loop such as PID regulators, LQG and H_∞. This algorithm does not require state-space model and can easily be implemented on thermodynamic model.

Because the major part of commercial turbofan engine thrust is due to the fan bypass flow, the fan speed is a good indicative of thrust to regulate. In this study, the min-max control structure contains 7 loops. The power lever angle (PLA) command is the main control loop and the other loops are related to min and max limiters.

The min-max selection strategy can be explained as follows:

$$W_{f-transient} = \max\left(\min\left(\begin{matrix} W_{f-PLA}, W_{f-Ps3\max}, W_{f-T45\max} \\ , W_{f-NHP\max}, W_{f-HPC-SM\min}, \\ W_{f-Ps3\min}, W_{f-RU\min} \end{matrix}\right),\ldots\right) \quad (6)$$

where W_{f-PLA} represent the fuel flow rate calculated by the PLA loop, $W_{f-Ps3\max}$, $W_{f-T45\max}$, $W_{f-HPC-SM\min}$ and $W_{f-NHP\max}$ demonstrate the fuel flow rate computed by the *Ps3*-max, *T45*-max, *HPC-SM*-min and N_{HP}-max limiter loops, respectively, and $W_{f-Ps3\min}$ and $W_{f-RU\min}$ are the fuel flow rate obtained by the *Ps3*-min and *RU*-min limiter loops, respectively. Thus, this selection strategy determines the final value of the engine transient fuel flow rate at each time instance. Finally, the transient fuel flow rate is added to the steady-state fuel flow and the sum is injected to the fuel actuation system. Another approach in applying the steady-state fuel flow rate is the use of integral control action. The ability of the control system to track varying reference commands and reject disturbances can be enhanced by including integral control action [24].

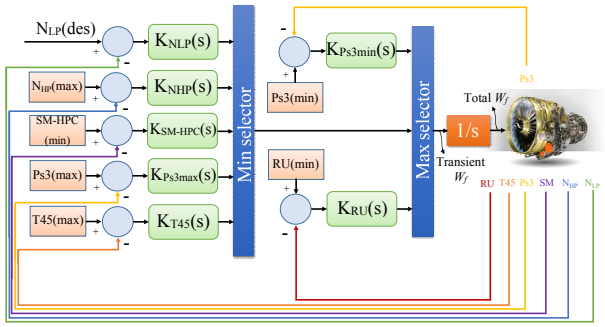


Fig. 3. Min-max controller structure with linear regulators and input integration.

The performance of the min-max algorithm strongly depends on the loops' controller types. If only proportional control is used, high frequency oscillations will occur in the limited outputs responses and also there is always an error between the desired input and actual response [10]. So, the use of integral and derivative control is effective or in some cases inevitable. Thus, the min-max algorithm is designed using linear compensators, as illustrated in Fig. 3. Linear control theories can be used to design the controller gains, but in this case the min-max algorithm provides a conservative engine response. Because the objective of this study is to achieve an enhanced response, an optimization algorithm is used to obtain the controller gains. Due to the nonlinear and switching nature of the min-max algorithm, a non-gradient optimization approach on the basis of the evolutionary algorithms is used. The application of evolutionary algorithms has been investigated for the optimization of fuel controller parameters in turbojet engines [40-43], however, they are not employed for turbofan engines, so far.

4.1.1 Min-max controller tuning using GA

Genetic algorithms (GAs) are numerical optimization algorithms based on natural genetics and natural selection, thus, they are great methods to apply to a wide range of problems [44]. In advanced turbofan engine control system, the control objective should be achieved by minimum fuel consumption as well as the shortest possible engine response time. In this paper, this work is accomplished using GA optimization

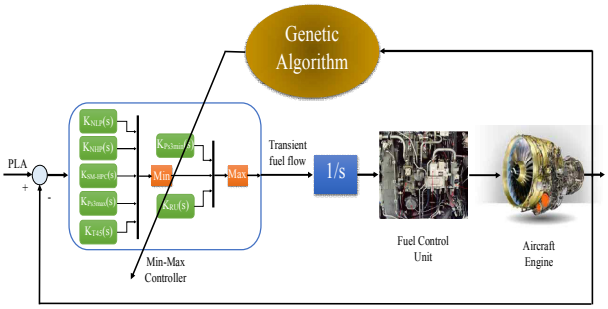


Fig. 4. Optimization procedure using genetic algorithm.

method, as depicted in Fig. 4. The arrow in this figure illustrates that the linear regulators of the min-max structure are obtained using GA optimization algorithm.

For this purpose, a cost function should be defined. The goal of this optimization problem is minimization of the cost function.

In order to minimize the engine fuel consumption and response time, the cost function for the min-max controller optimization is defined as follows:

$$J = w_1 \int_0^{sim-time} \frac{w_f}{\left(w_f\right)_{\max} \times \dfrac{sim-time}{sample\,time}} dt + w_2 \left(\frac{t_{acc} + t_{dec}}{sim-time} \right) \quad (7)$$

where w_f is the instantaneous fuel flow, $\left(w_f\right)_{\max}$ is the maximum total fuel flow, *sim-time* is the simulation time that is the time interval in which the simulation is performed, *sample time* is the simulation time step and t is the time index. In addition, t_{acc} and t_{dec} are the acceleration and deceleration times required by the engine to follow the PLA command [43]. In this study, the simulations are performed in 25 seconds, as shown in Figs. 9-26. Moreover, the $\left(w_f\right)_{\max}$ parameter is the fuel flow value in which the spool speeds of the engine reach to their maximum limits.

The settling time is an acceptable representative of the response speed of the system. Each term in Eq. (7) is weighted according to their importance by the coefficients of w_i, where the sum of w_i $(i = 1,2)$ should be equal to 1.

The physical limitations should also be satisfied for safe engine operation. The penalty technique is the most common method in constrained GA optimization problems to handle constraints. Various methods are used to apply the penalty function to the optimization problem. In this research, Taguchi & Yakota method [45] is used to define the penalty function. According to the requirements of Sec. 3, the penalty factor is formulated as follows:

$$PF = 1 + \frac{1}{8}\sum_{i=1}^{8} \left(\begin{array}{l} \left| \dfrac{\Delta b_{N_{LP}\max}}{5000} \right| + \left| \dfrac{\Delta b_{N_{LP}\min}}{4800} \right| + \left| \dfrac{\Delta b_{N_{HP}}}{14460} \right| + \ldots \\ \left| \dfrac{\Delta b_{Ps3\max}}{35.1} \right| + \left| \dfrac{\Delta b_{Ps3\min}}{32.6} \right| + \left| \dfrac{\Delta b_{T45\max}}{1155} \right| + \ldots \\ \left| \dfrac{\Delta b_{HPC-SM}}{22.9} \right| + \left| \dfrac{\Delta b_{RU}}{0.0355} \right| \end{array} \right), \quad (8)$$

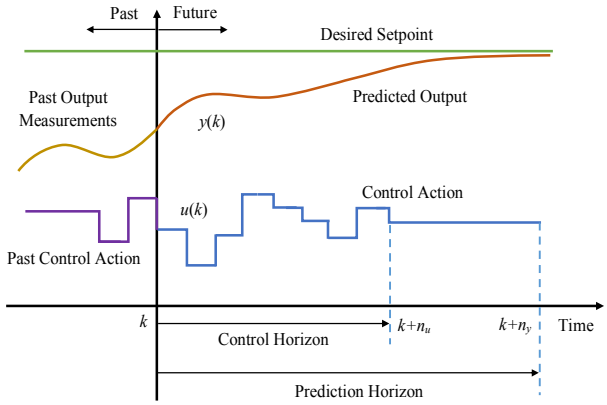


Fig. 5. Model predictive control strategy.

$$\begin{aligned}
\Delta b_{N_{LP}\max} &= \max\left(0, \max\left(N_{LP}(t) - 5000\right)\right) \\
\Delta b_{N_{LP}\min} &= \min\left(0, \min\left(N_{LP}(t) - 4800\right)\right) \\
\Delta b_{N_{HP}} &= \max\left(0, \max\left(N_{HP}(t) - 14460\right)\right) \\
\Delta b_{Ps3\max} &= \max\left(0, \max\left(Ps3(t) - 35.1\right)\right) \\
\Delta b_{Ps3\min} &= \min\left(0, \min\left(Ps3(t) - 32.6\right)\right) \\
\Delta b_{T45\max} &= \max\left(0, \max\left(T45(t) - 1155\right)\right) \\
\Delta b_{HPC-SM} &= \min\left(0, \min\left(HPC_{SM}(t) - 22.9\right)\right) \\
\Delta b_{RU} &= \min\left(0, \min\left(RU(t) - 0.0355\right)\right)
\end{aligned} \quad (8)$$

where Δb is the maximum value of violation for the constraints.

Finally, the penalty function of Eq. (8) is multiplied by the cost function (Eq. (7)), and thus, the final objective function is constructed.

4.2 Model predictive control

Based on the predictive control strategy, future behavior of a process is predicted over a determined prediction horizon using a model of the process [46]. Fig. 5 represents the MPC strategy. The predicted control signal is calculated by minimizing an objective function to keep the predicted output as close as possible to the reference trajectory. The first value of the control signal is applied to the process while the remaining values are rejected due to the receding horizon concept [47]. Fig. 6 illustrates the components of general MPC structure which is applied to a turbofan engine.

Consider a proper discrete-time linear model of the turbofan engine which is augmented to control update equation to form the incremental discrete state-space system as [48]:

$$\begin{aligned}
\begin{bmatrix} \mathbf{x}(k+1) \\ \Delta W_f(k) \end{bmatrix} &= \begin{bmatrix} \mathbf{A}_d & \mathbf{B}_d \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ \Delta W_f(k-1) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_d \\ \mathbf{I} \end{bmatrix} \Delta\left(\Delta W_f\right)(k), \\
\mathbf{y}(k) &= \begin{bmatrix} \mathbf{C}_d & \mathbf{D}_d \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ \Delta W_f(k-1) \end{bmatrix} + \mathbf{D}_d \Delta\left(\Delta W_f\right)(k)
\end{aligned} \quad (9)$$

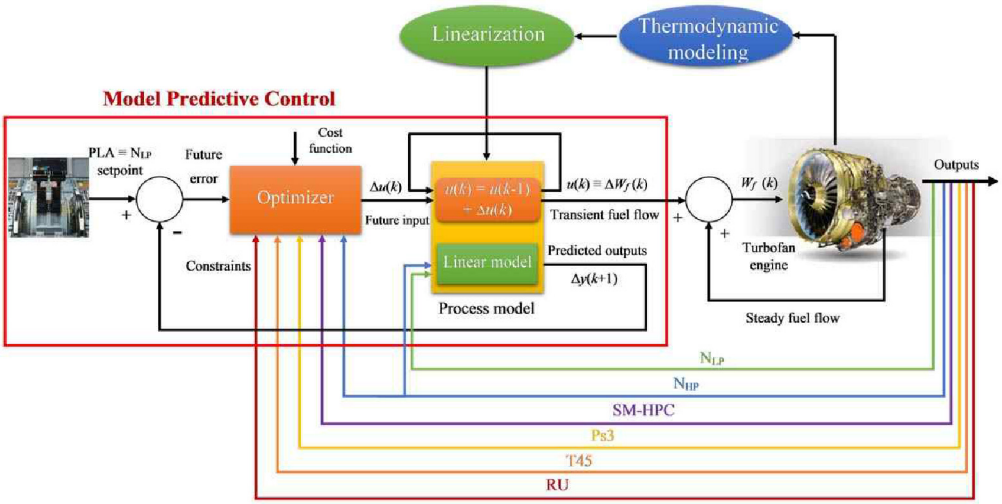


Fig. 6. Model predictive structure for turbofan engine fuel control.

where $\Delta\left(\Delta W_f\right)(k)$ is the incremental control input and the state and output variables are defined as follows:

$$\begin{aligned}\mathbf{x}^T(k)&=\left[\Delta N_{LP}(k)\quad \Delta N_{HP}(k)\right]\\ \mathbf{y}^T(k)&=\left[\Delta N_{LP}(k)\quad \Delta N_{HP}(k)\quad \Delta HPC-SM(k)\right.\\ &\quad\left.\Delta Ps3(k)\quad \Delta T45(k)\quad \Delta W_f/Ps3(k)\right].\end{aligned}\tag{10}$$

It is worth mentioning that matrix $\boldsymbol{D_d}$ is not zero and the size of its elements depends on the direct effect of the change in fuel flow on the desired outputs. As shown in Fig. 6, the linear model obtained from the thermodynamic model of the engine is augmented with the control update equation to attain an offset-free response. So, the augmented model in general form is:

$$\begin{aligned}\mathbf{x}_a(k+1)&=\mathbf{M}\mathbf{x}_a(k)+\mathbf{N}\Delta\left(\Delta W_f\right)(k),\\ \mathbf{y}(k)&=\mathbf{Q}\mathbf{x}_a(k)+\mathbf{D}_{\mathrm{d}}\Delta\left(\Delta W_f\right)(k)\end{aligned}\tag{11}$$

where the augmented state vector is given as:

$$\mathbf{x}_a^T(k)=\left[\Delta N_{LP}(k)\quad \Delta N_{HP}(k)\quad \Delta W_f{}^T(k-1)\right].\tag{12}$$

and the definition of matrices **M**, **N** and **Q** is obvious. As shown in Fig. 6, MPC structure contains an optimizer to produce an optimal control input. The objective function and the engine limits are the requirements of the optimizer. Thus, the objective function simply can be defined as [48]:

$$\begin{aligned}J'=&\left(\Delta\left(\Delta\hat{W}_f\right)\right)^T\left[\mathbf{\Phi}^T\mathbf{\Phi}+\lambda\mathbf{I}\right]\Delta\left(\Delta\hat{W}_f\right)\\ &+2\left[\mathbf{x}_a^T\mathbf{F}^T\mathbf{\Phi}-\mathbf{r}^T\mathbf{\Phi}\right]\Delta\left(\Delta\hat{W}_f\right)\end{aligned}\tag{13}$$

where λ is a scalar weighting factor and **r** is the reference trajectory and matrices **F** and $\mathbf{\Phi}$ are described as:

$$\mathbf{F}=\begin{bmatrix}\mathbf{QM}\\ \mathbf{QM}^2\\ \vdots\\ \mathbf{QM}^{n_y}\end{bmatrix},\quad \Phi=\begin{bmatrix}\mathbf{QN} & \mathbf{0} & \mathbf{0} & \cdots\\ \mathbf{QMN} & \mathbf{QN} & \mathbf{0} & \cdots\\ \vdots & \vdots & \vdots & \vdots\\ \mathbf{QM}^{n_y}\mathbf{N} & \mathbf{QM}^{n_y-2}\mathbf{N} & \mathbf{QM}^{n_y-3}\mathbf{N} & \cdots\end{bmatrix}.\tag{14}$$

Depending on the system constraints, the objective function should be minimized with respect to the following constraints:

$$\begin{aligned}&\Delta W_{f\min}\le\Delta W_f(k+i)\le\Delta W_{f\max}\\ &\quad\text{for } i=0,1,\cdots,n_u-1\ \ (\text{input constraints})\\ &\mathbf{Y}_{\min}\le\mathbf{y}(k+i)\le\mathbf{Y}_{\max}\\ &\quad\text{for } i=1,\cdots,n_y\ \ (\text{output constraints})\end{aligned}\tag{15}$$

where $\mathbf{U}_{\min}$, $\mathbf{U}_{\max}$, $W_{f\min}$ and $W_{f\max}$ are upper and lower bounds of input and outputs, n_u is control horizon and n_y is prediction horizon.

In this research, the MPC has been programmed in C++ language. The solution to this problem relies on application of a quadratic programming procedure [49]. In this study, in order to solve this problem, Hildreth's quadratic programming procedure is utilized which offers simplicity and reliability in real-time implementation [49].

4.2.1 Optimization of MPC horizons

The final goal of a new control algorithm design is its implementation on a real hardware. So, it must have real-time simulation capability while providing a satisfactory performance. Therefore, the MPC control and prediction horizons should be selected reasonably. In order to have lower computational burden, the horizons selected should be as small as possible. But, in order to have a satisfactory performance, it is

better to have an overshoot/undershoot free response which is provided by larger values of the horizons. Thus, an optimization procedure can be applied to find trade-off values of the horizons.

For this purpose, the cost function is composed of the controller run-time and the engine response time. Therefore it is defined as follows:

$$J_h = w_1\left(\frac{run-time}{sim-time}\right) + w_2\left(\frac{t_{acc}+t_{dec}}{sim-time}\right) \tag{16}$$

where *run-time* is the time it takes to run the MPC program once. Moreover, the penalty factor should penalize the overshoot/undershoot percentage of the controlled output (N_{LP}). Therefore it is formulated using Taguchi & Yakota method as follows:

$$PF_h(t) = 1 + \frac{1}{2}\left(\begin{array}{l}\dfrac{\max\left(0,\max\left(y_{con}(t) - PLA_{\max}\right)\right)}{PLA_{\max}} \\ +\dfrac{\min\left(0,\min\left(y_{con}(t) - PLA_{\min}\right)\right)}{PLA_{\min}}\end{array}\right) \tag{17}$$

where $y_{con}(t)$ is the controlled output (N_{LP}) time response. Multiplication of Eq. (16) by Eq. (17) forms the final objective function.

5. Hardware-in-the-loop setup

The implementation of the controller on a hardware is a main challenge toward the development of the ECU. In order to confirm the performance accuracy of the controller, multiple tests should be performed because the faultless operation of the controller is essential for the efficient performance of the engine at any operating point.

To implement the designed control algorithms, the appropriate hardware should be selected due to the computational burden of the controller. There are numerous commercial I/O boards with different specifications including processing power, RAM, clock speed, I/O number, and etc. One of the most widely used commercial boards is the Arduino microcontrollers that initially among them, the Arduino DUE microcontroller board, with a processing speed of 84 MHz is selected for implementation of the min-max algorithm. Specifications of this microcontroller are listed in Table 2. To evaluate the feasibility of a real-time simulation of the MPC algorithm with this hardware, the runtime of one time step of the algorithm is calculated. Performing each time step of the MPC algorithm takes 1.78 seconds using Arduino DUE. Since each sample time of Simulink solver of the thermodynamic model for generating smooth responses should be less than 0.04 seconds, so, it is not possible to run a real-time simulation of the MPC using the Arduino DUE board. Therefore, real-time testing of the MPC algorithm requires a microcontroller with a stronger CPU. In this regard, one of the most suitable and reliable options is Intel board.

Table 2. The specifications of Arduino Due and Intel-Edison boards.

Component	Specification of Intel Edison	Specification of Arduino DUE
SoC	Dual-core, dual-threaded Intel® Atom™ CPU and a 32-bit Intel® Quark™ microcontroller	AT91SAM3X8E
Digital I/O pins	20 (of which 6 provide analog input and 4 provide PWM output)	54 (of which 12 provide analog input, 12 provide PWM output and 2 provide analog output)
Input voltage	7-15 V	7-12 V
Operating voltage	3.3 V / 5 V	3.3 V
Flash storage	4 GB eMMC	512 KB
RAM	1 GB LPDDR3 POP	96 KB
Clock speed	500 MHz (Intel® Atom™ CPU) 100 MHz (Quark™ microcontroller)	84 MHz

SoC: System on chip; PWM: Pulse width modulation; RAM: Random access memory

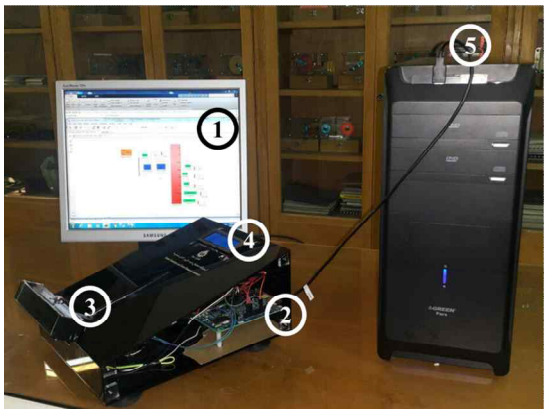


Fig. 7. The HIL platform.

The Intel Edison board is selected which has a CPU of 600 MHz processing speed. The specifications of this microcontroller have been listed in Table 2. This hardware is a development board which uses the Intel processor. The runtime of one time step of the MPC algorithm is again calculated using this electronic board. The runtime of 0.02 seconds for each time step of the MPC algorithm is obtained which indicates that the Intel Edison board is suitable for the real-time HIL implementation of the MPC algorithm for the sampling times of more than 0.02 seconds.

As shown in Fig. 7, an HIL test bench is prepared. In this figure, the thermodynamic model of the turbofan engine on a personal computer (PC), the built-in box including the Intel Edison board, a lever as the PLA and an input/output data display and serial communication between the PC and the Intel board have been presented. The power lever angle is

Table 3. The state variables, control input, outputs and incremental output limits.

State variables	Input variables	Output variables	Output limits
Fan speed, ΔN_{LP}	Fuel flow, ΔW_f	Fan speed, ΔN_{LP}	----
		Core speed, ΔN_{HP}	≤195 rpm
		HPC stall margin, $\Delta HPC - SM$	≥ -1 %
Core speed, ΔN_{HP}		HPC discharge static pressure, $\Delta Ps3$	≤ 1.3 bar, ≥ −1.3 bar
		HPT exit total temperature, $\Delta T45$	≤ 27 °K
		Ratio unit, $\Delta W_f / Ps3$	≥ -0.001 kg / bar.sec

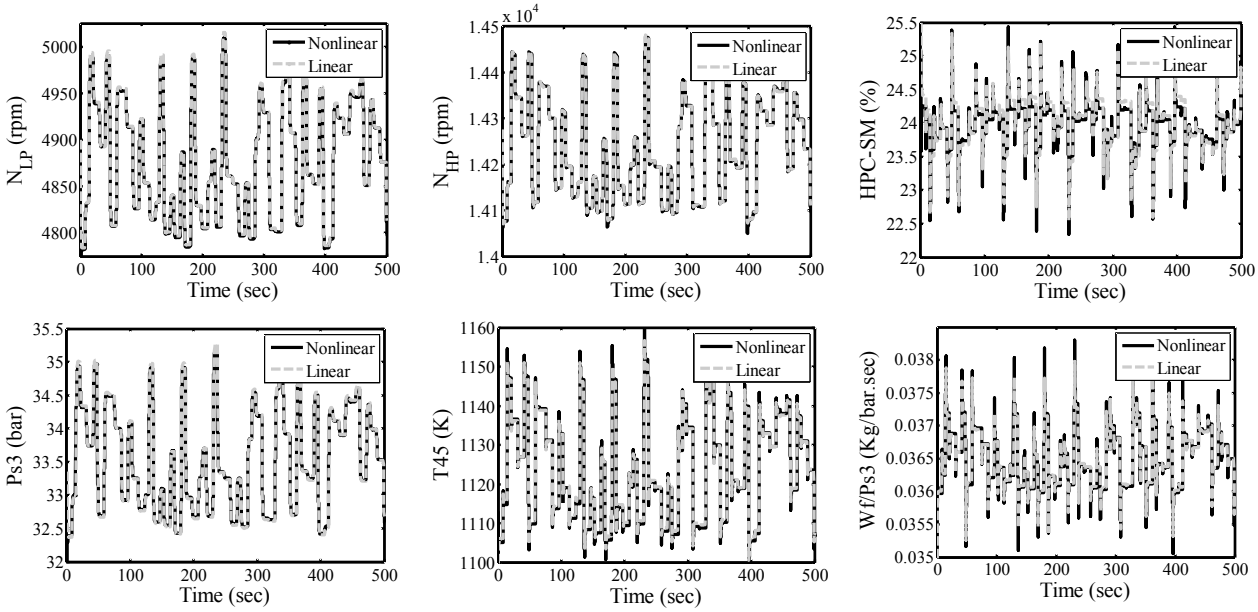


Fig. 8. Comparison of the outputs of the linear and nonlinear models.

converted to a voltage signal using a potentiometer and it is transmitted to the Intel board via the analog input pin.

As shown in Fig. 7, the nonlinear thermodynamic model of the turbofan engine which has been simulated in a PC is the software section of this test and represents the real engine. The model uses the Simulink real-time environment. Moreover, the control strategy is implemented on a hardware as the engine ECU, so, it is the hardware component of the HIL simulation. To run the simulation in real-time the "real time windows target" and "stream input, stream output" blocks of the Simulink are utilized.

6. Results and discussion

In this section, an operating point in sea level static (SLS) and international standard atmosphere (ISA) conditions in 98 % of the fan speed are considered for controller design. The thermodynamic model is linearized about this operating point. In addition, the input delay is fixed to zero and the noise matrix is omitted. The input and state variables as well as the outputs and their incremental bounds are listed in Table 3. The fit to estimation data of the linearization results are obtained [98.04;98.41;95.38;98.59;98.11;98.07] for the various outputs, respectively. The outputs of the linear and thermodynamic models due to the QAPRBS signal are compared in Fig. 8. In order to make the outputs of the two models comparable, the steady value of each variable is added to the output of the linear model. Fig. 8 demonstrates the excellent compliance of the outputs, which indicates the high accuracy of the linear model.

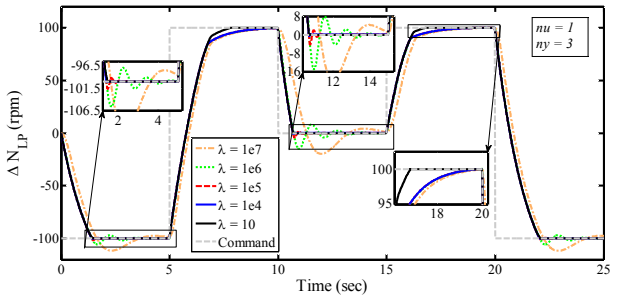
The model-in-the-loop (MIL) simulations are conducted on a PC with a Core2Due CPU of 2.5 GHz and 4 GB of RAM using MATLAB software. Moreover, in the HIL simulation, the thermodynamic model presented in Sec. 2 has been run on this computer. The run time of each time step of the thermodynamic model is about 0.0025 sec for the sample time of 0.01 sec, which is fast enough for HIL simulation and online optimization purposes.

6.1 Model-in-the-loop (MIL) simulation

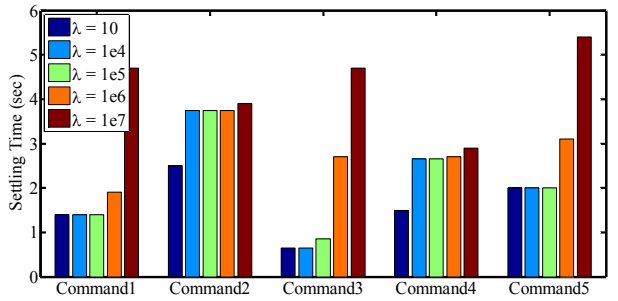
The parameters used for genetic algorithm for optimization of the min-max regulators' coefficients are presented in Table 4. These parameters are selected as common, and the population size and the generation increase do not change the results. Moreover, the objective function weight factors $w_1 = 0.5$ and

Table 4. Parameters used for genetic algorithm.

Parameter	Value
Population size	200
Percent of crossover	0.9
Percent of mutation	0.1
Generation	50



(a)



(b)

Fig. 9. Effect of weighting factor on fan speed response - using MPC with linear plant: (a) Fan speed response; (b) settling time.

$w_2 = 0.5$ are selected which means that the importance of the two objectives is equal in the optimization process.

Figs. 9 and 10 illustrate the effect of various weighting factors (λ) on the MPC performance. A fan speed scenario including 5 different set-points is defined. Fig. 9(a) shows the ΔN_{LP} response in following the desired profile for different values of λ. As shown in the figure, the maximum percent undershoot decreases with decreasing value of λ. Also, as depicted in Fig. 9(b), the settling time of the different commands decreases with the reduction of λ. Fig. 9(b) demonstrates that the low values provide faster fan speed response. This issue is confirmed according to Fig. 10. As indicated in this figure, low values of λ lead to a quick response and saturation of the actuator. Thus, λ is selected so that it provides a satisfactory response with respect to the actuator's limitation. On the other hand, the values of the prediction and control horizons are calculated via GA optimization. Thus, the optimal values of n_u and n_y are obtained as 2 and 6, respectively, and λ is selected as 10^4.

MPC is designed based on the linear model and the min-max regulators are optimized based on the thermodynamic model. Since the thermodynamic model is the representative

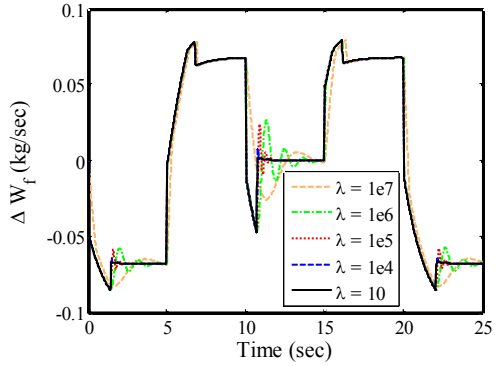


Fig. 10. Effect of weighting factor on fuel flow consumption - using MPC with linear plant.

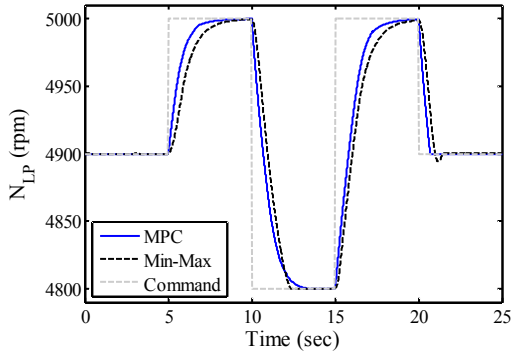


Fig. 11. Fan speed response - thermodynamic plant: MPC vs min-max.

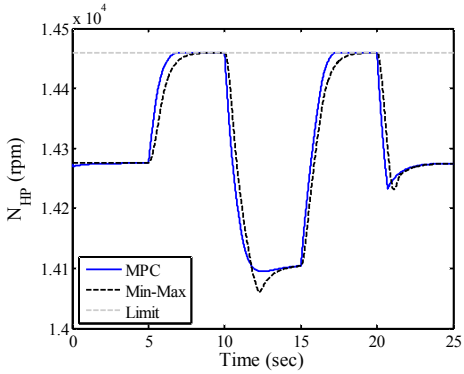


Fig. 12. Limited output N_{HP} response and its limit- thermodynamic plant: MPC vs min-max.

of the real engine, it should be used as the plant to properly compare MPC and min-max performance. The control objective is to follow the fan speed set-point as the desired reference while maintaining constrained outputs within bounds at all times. A fan speed scenario is defined as the reference trajectory of MPC. Different commands including acceleration and deceleration set-points of N_{LP} are defined in the demand profile. Fig. 11 demonstrates that N_{LP} response using both controllers follows the desired profile.

One of the requirements of the engine performance is the need to operate the engine as close as possible to its limits. So,

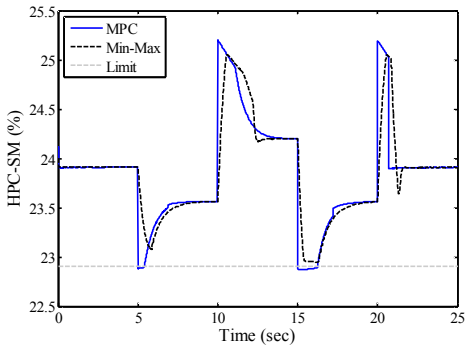


Fig. 13. Limited output $HPC-SM$ response and its limit - thermodynamic plant: MPC vs min-max.

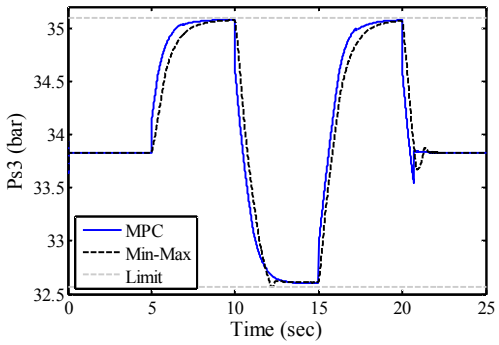


Fig. 14. Limited output $Ps3$ response and its limits - thermodynamic plant: MPC vs min-max.

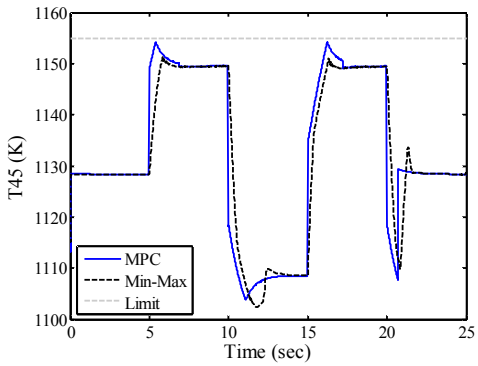


Fig. 15. Limited output $T45$ response and its limit - thermodynamic plant: MPC vs min-max.

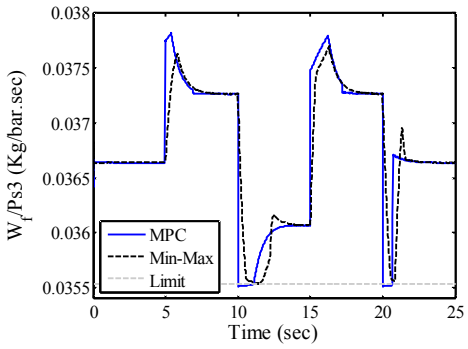


Fig. 16. Limited output $\left(W_f/Ps3\right)$ response and its limit - thermodynamic plant: MPC vs min-max.

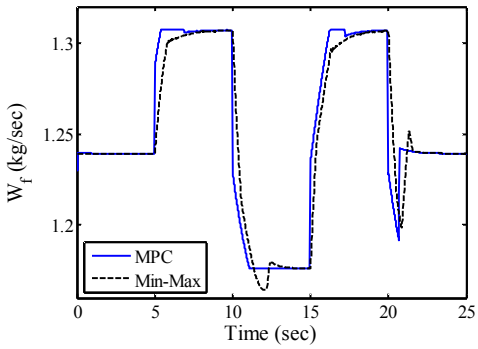


Fig. 17. Fuel control input - thermodynamic plant: MPC vs min-max.

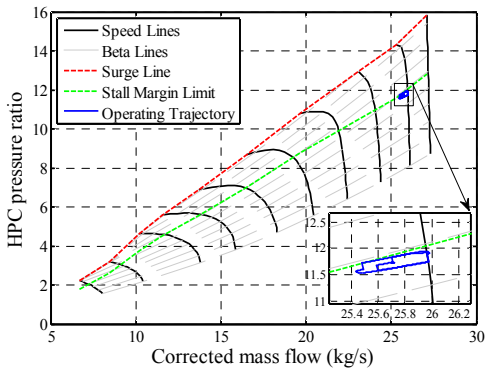


Fig. 18. Transient trajectory using MPC - Scaled HPC map.

the engine limit protection assurance is a serious challenge. As illustrated in Figs. 12-16, the limited outputs responses with MPC are completely close to their limits without exceeding the limits but min-max provides conservative responses. Fig. 18 depicts the characteristic curve of the HPC and the operating trajectory using MPC. During acceleration, the operating trajectory in the HPC characteristic curve moves towards the HPC stall margin limit.

Time response characteristics of the controlled output, i.e. N_{LP}, using MPC and min-max in all 4 commands and the percentage difference of these characteristics are presented in Table 5. These characteristics are defined as [50]:

(1) Rise time: The time required for the response to rise from 10 % to 90 % of its final value.

(2) Settling time: The time required for the response to reach and remain within 5 % of its final value.

(3) Delay time: The time required for the response to reach 50 % of its final value.

Control and prediction horizons and scalar weight of the MPC are designed in such a way that the system response has no overshoot or undershoot, as indicated in Fig. 11. Therefore, the maximum percent overshoot of the system response using MPC and min-max are almost zero. The results of Table 5 represent that the rise time and settling time of the fan speed response in the second command with min-max is slightly less

Table 5. Comparison of time response characteristics - thermodynamic plant - MPC vs min-max.

	Command 1		Diff.	Command 2		Diff.	Command 3		Diff.	Command 4		Diff.
	MPC	Min-max	%	MPC	Min-max	%	MPC	Min-max	%	MPC	Min-max	%
Rise time (sec)	1.39	1.87	25.7	1.63	1.55	4.9	1.7	2.06	17.5	0.42	0.46	8.7
Overshoot (%)	0	0	0	0	0	0	0	0	0	0	-5.6	5.6
Settling time (sec)	1.85	2.8	34	2.12	2.06	2.8	2.24	2.98	24.8	0.61	0.88	30.7
Delay time (sec)	0.51	0.91	44	0.7	0.96	27	0.82	1.04	21	0.31	0.56	44.6

Table 6. Comparison of time response characteristics - HIL simulation - MPC vs min-max.

	Command 1		Diff.	Command 2		Diff.	Command 3		Diff.	Command 4		Diff.
	MPC	Min-max	%	MPC	Min-max	%	MPC	Min-max	%	MPC	Min-max	%
Rise time (sec)	1.6	1.62	1.25	1.54	1.66	6.6	1.72	1.90	9.5	0.56	0.65	15.4
Overshoot (%)	0	0	0	0	0	0	0	0	0	0	-3	3
Settling time (sec)	3.86	2.28	40	1.93	2.20	12.3	2.28	2.66	14.3	0.68	0.93	26.9
Delay time (sec)	0.54	0.89	39.3	0.72	0.96	25	0.85	1.01	15.8	0.34	0.55	38.2

Table 7. Comparison of time response characteristics in various flight conditions - MPC vs min-max.

Mach number	Altitude (m)	Setpoint		Rise time (sec)		Settling time (sec)		Delay time (sec)	
		From	To	MPC	Min-max	MPC	Min-max	MPC	Min-max
0	0	0	100	1.03	1.58	1.22	2.22	0.48	0.9
0	0	100	0	0.55	0.65	0.64	0.93	0.29	0.55
0.25	0	0	100	1	1.6	1.28	2.2	0.52	0.9
0.25	0	100	0	0.57	0.65	0.65	0.93	0.3	0.55
0.3	1500	0	100	1.26	1.67	1.48	2.32	0.62	0.9
0.3	1500	100	0	0.64	0.71	0.73	0.99	0.34	0.57
0.4	3000	0	100	1.65	1.74	1.99	2.43	0.71	0.95
0.4	3000	100	0	0.69	0.76	0.78	1.05	0.35	0.59
0.6	4500	0	100	2.41	2.48	3.23	3.48	0.87	1.08
0.6	4500	100	0	0.74	0.82	0.85	1.11	0.37	0.61

than MPC but the delay time of the response using min-max is higher than MPC. Moreover, the settling time and delay time of the system using MPC are significantly (21 % to 44.6 %) lower than min-max. According to the MIL simulation results, MPC has significantly improved the response time of the system in comparison with min-max algorithm.

Moreover, the computational time of min-max algorithm and MPC are 0.71 and 6.8 seconds, respectively. Although the real-time implementation of MPC is feasible using the mentioned computer, it requires a more expensive hardware than min-max for conducting a test-stand with actual engine and processor. Recent developments in distributed engine control architecture and processors power make it quite possible to implement MPC for turbofan engine control.

6.2 HIL simulation

After designing the MPC and min-max algorithms and performing MIL simulations, the controllers are implemented on the hardware by considering the thermodynamic model as the plant and HIL testing is carried out to verify the implementation and operation of the controllers. In HIL simulation of the MPC algorithm, the selection of sampling time is very important. In fact, the selection of appropriate sampling time for HIL testing has a significant effect on the accuracy of the results. In serial communication, there are three main factors to determine the speed required for data transfer, 1- The number of data sent on the serial port, 2- Maximum bit required to send data and 3- The sampling time. The minimum required data transfer speed determines by multiplying these three factors. Fig. 19 shows the fan speed response obtained from three HIL tests of the MPC algorithm with sampling times of 30, 50 and 100 milliseconds. As shown in this figure, the sampling time of 100 ms has the highest amplitude fluctuations and the amplitude of fluctuations decreases by reducing the sampling time. Thus, the sampling time of 30 ms is selected for the HIL test. The sampling time less than 20 ms will cause the selected hardware not to have the ability to run real-time and a method to reduce the computational burden of the MPC should be taken.

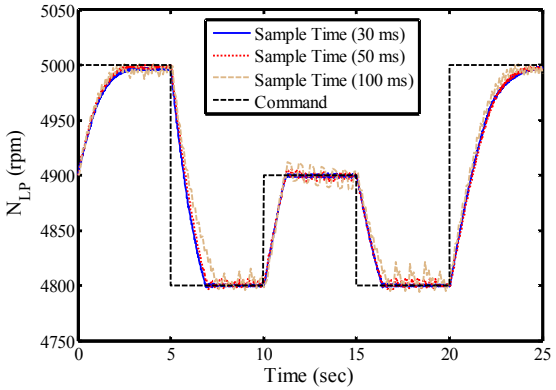


Fig. 19. Fan speed response for three different sampling times in the HIL simulation of the MPC algorithm.

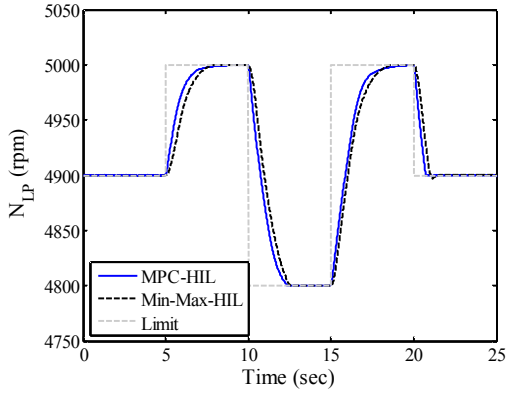


Fig. 20. Fan speed response to a general command - HIL simulation: MPC vs min-max.

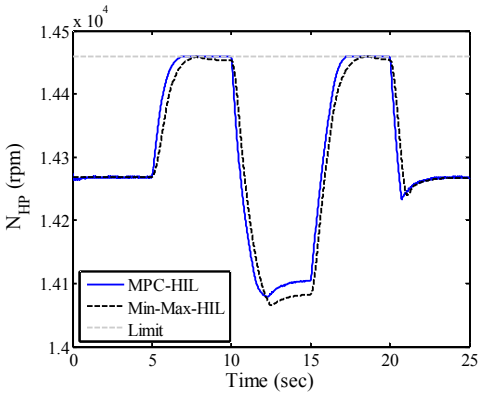


Fig. 21. Limited output N_{HP} response and its limit - HIL simulation: MPC vs min-max.

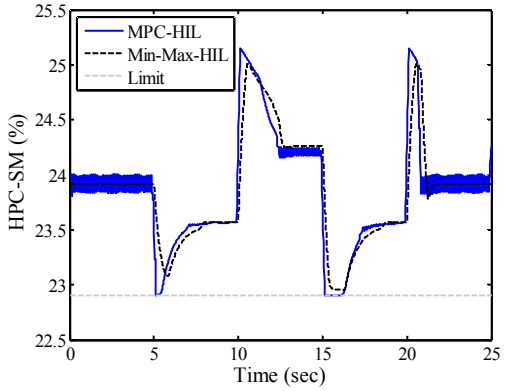


Fig. 22. Limited output $HPC-SM$ response and its limit - HIL simulation: MPC vs min-max.

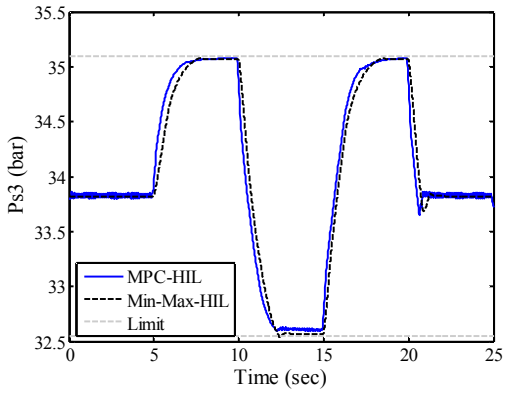


Fig. 23. Limited output $Ps3$ response and its limits - HIL simulation: MPC vs min-max.

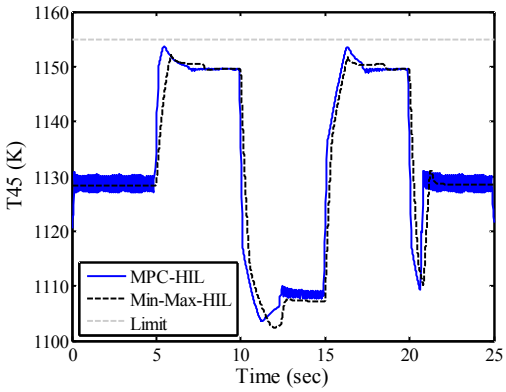


Fig. 24. Limited output $T45$ response and its limit - HIL simulation: MPC vs min-max.

A reference command is applied similar to the reference trajectory in the MIL simulation, as shown in Fig. 20. In the HIL simulation of the MPC algorithm, this command is entered by the user via the lever shown in Fig. 7. Fig. 20 also indicates the fan speed response in the HIL simulations of the both controllers in trajectory tracking.

The response time specifications of the HIL simulation are presented in Table 6. As shown in this table, the rise time of the MPC is from 1.25 % in command 1 to 15.4 % in command 4 lower than min-max. In addition, the settling time of MPC is improved from 12.3 % in command 2 to 26.9 % in command 4, except in the first command. Moreover, the delay time of the system using MPC is significantly lower than min-max from 15.8 % in command 3 to 39.3 % in command 1. Therefore, the HIL simulation results approve that MPC has considerably improved the response time of the system which

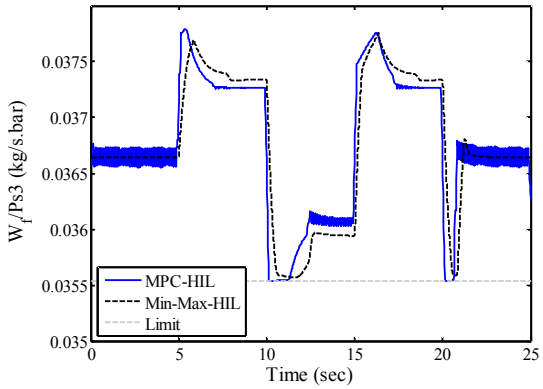


Fig. 25. Limited output $W_f/Ps3$ response and its limit - HIL simulation: MPC vs min-max.

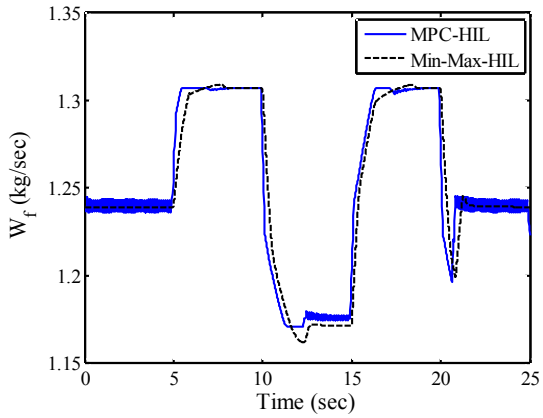


Fig. 26. Fuel control input - HIL simulation: MPC vs min-max.

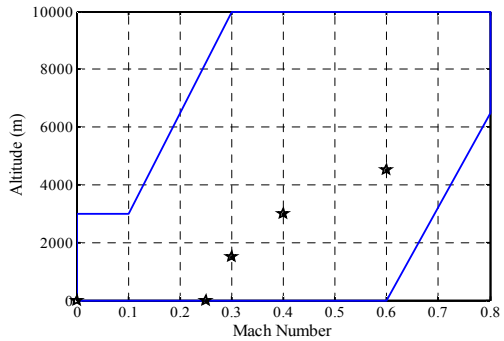


Fig. 27. The engine flight envelope and some flight conditions for the controller design.

also confirm the results of the MIL simulation.

Figs. 21-25 show the limited outputs responses in the HIL simulation confirm the MIL results and illustrate that the controllers in both MIL and HIL simulations have not violated the limits. Fig. 26 shows the rate of fuel flow calculated in the HIL simulations. By applying the input command by the user, the MPC quickly changes the fuel flow rate to achieve the desired set-point and when the controlled output (N_{LP}) is achieved to the final value, the fuel flow rate is reduced to a constant steady-state value.

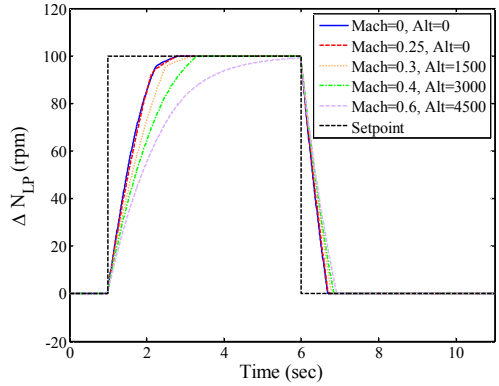


Fig. 28. Incremental fan speed responses using MPC in various flight conditions.

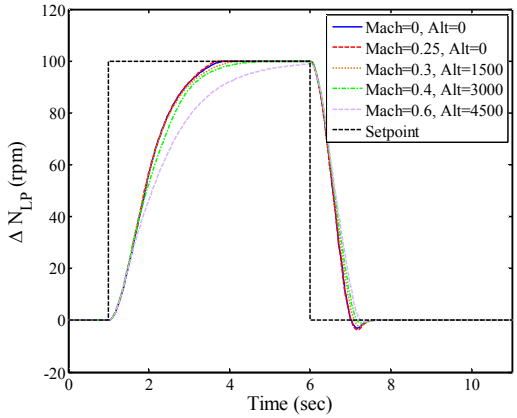


Fig. 29. Incremental fan speed responses using min-max controller in various flight conditions.

As shown in Figs. 20-26, the performance of the controllers implemented in both the steady state and the transient modes are very consistent with the MIL simulations which represents the successful implementation of the controllers. According to the comparison of the HIL and MIL simulation results of the MPC, the sampling time of 30 ms is acceptable and shows the accuracy of the hardware implementation and HIL test. However, oscillating behavior is observed in the HIL simulation results of the MPC. There are various reasons for these fluctuations. The discretization of the internal model of MPC and the conversion of the analog to digital signal through the sampling process lead to the loss of a part of data and create a delay between two consecutive signals. Moreover, the transmission of the analog voltage signal of the potentiometer to the hardware generates noise on the signal. Therefore, the noise of the output analog voltage signal of the potentiometer and the selected sampling time of the HIL test are the main factors which contributes to these fluctuations.

6.3 Analysis in the flight envelope

In this section, different operating points are assumed in flight conditions to investigate the engine response time using

MPC. The engine studied in this research is proper for the class of mid-range single aisle commercial aircrafts depicted in Fig. 27 as a typical flight envelope of this type of turbofan engines [35]. The operating points considered in the flight envelope are also shown in Fig. 27. The engine thermodynamic model is linearized at these points and model predictive and min-max controllers are designed based on the linear models. The controllers are then applied to the linear models. For this analysis, a new scenario is defined for ΔN_{LP} including one acceleration and one deceleration commands. Figs. 28 and 29 illustrate the incremental fan speed in tracking the desired reference using MPC and min-max, respectively. As expected, by increasing the altitude the engine responds much more slowly and it is independent of the controller effort. This can be explained by the reduction of air density at high altitudes. The time response data are presented in Table 7. The fan speed response time using MPC is significantly lower than min-max, at all operating points, as demonstrated in Table 7. As shown in this table, the rise time of the MPC is from minimum 2.8 % to maximum 37.5 % lower than min-max. Moreover, the settling time of MPC is from minimum 7.2 % to maximum 45 % lower than min-max which has significantly improved. In addition, the delay time of the system using MPC is considerably lower than min-max from minimum 19.4 % to maximum 47.2 %.

7. Conclusion

In this paper, a comparative study was carried out for comparison of the performance of MPC with optimized horizons and optimized min-max algorithm for two-spool turbofan engine fuel control, taking into account all the necessary constraints required for safe operation of the engine, as well as hardware implementation and HIL test of the both controllers. In order to achieve an improved transient performance as well as minimum fuel consumption, the linear regulators of min-max algorithm was acquired through GA optimization approach. On the other hand, MPC was formulated and designed based on the proper discrete-time linearized state-space model and programmed in C++ language with real-time optimization using Hildreth's procedure. The MPC horizons were also obtained through GA optimization procedure.

The MIL simulation results illustrated that MPC provided faster response than min-max whilst ensuring the limit protection. Also, the system response with MPC was as close as possible to the engine limits, which improved the performance of the engine, while min-max provided conservative response. Finally, an HIL test bench was developed to verify the MIL simulation results. MPC requires more expensive processor than min-max for real-time implementation. The HIL simulation results confirmed the MPC ability for application as the control architecture of commercial engines. Moreover, the analysis was performed at several operating points in the engine flight envelope of which the same results were achieved.

Nomenclature

$\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$	: Matrices of the linear state-space model
b_i	: The value of the ith limit
$\mathbf{e}$	: The white-noise disturbance
$\mathbf{F}$, $\mathbf{\Phi}$	: Matrices of the predicted outputs equation
f_1	: The look-up table of the compressor map
f_3	: The look-up table of the turbine map
f_2, f_4, g	: The tables of the thermodynamic characteristics
h	: Enthalpy
H	: Fuel heating value
HPC	: High pressure compressor
HPC-SM	: High pressure compressor stall margin
HPS	: High pressure spool
HPT	: High pressure turbine
$\mathbf{I}$	: Identity matrix
J	: Objective function
J_{LP}, J_{HP}	: Moment of inertia of low pressure and high pressure shafts
k	: Time step
K_i	: Min-max compensators
LPC	: Low pressure compressor
LPS	: Low pressure spool
LPT	: Low pressure turbine
M	: Mach number
M, N, Q	: Matrices of the augmented model of MPC
m	: The number of limits
N_{LP}, N_{HP}	: Speed of low pressure and high pressure shafts
$\dot{N}_{LP}$, $\dot{N}_{HP}$	: Angular acceleration of low pressure and high pressure shafts
n_u, n_y	: Control and prediction horizons
P	: Pressure
PF	: Penalty factor
PLA	: Power lever angle
PR	: Pressure ratio
$Ps3$	: High pressure compressor discharge static pressure
PW	: Power
$\mathbf{r}$	: Reference trajectory
t	: Time
T	: Temperature
$T45$	: High pressure turbine exit total temperature
u	: Velocity
$\mathbf{u}$	: Vector of control variables
$\hat{\mathbf{u}}$	: Vector of predicted inputs
$\mathbf{U_{min}}$, $\mathbf{U_{max}}$	: Vector of lower and upper bounds of the inputs
W_f	: Fuel flow rate
$W_f/Ps3$, RU	: Ratio unit limiter
w_i	: Weighting values
$\mathbf{x}$	: Vector of state variables
$\mathbf{y}$	: Vector of output variables
$\hat{\mathbf{y}}$	: Vector of predicted outputs
$\mathbf{Y}_{\min}$, $\mathbf{Y}_{\max}$	: Vector of lower and upper bounds of the outputs
η	: Efficiency
λ	: Scalar weighting factor

Subscripts

a	: Augmented matrix
acc	: Acceleration
amb	: Ambient
b	: Burner
c	: Corrected
con	: Controlled
d	: Discrete time
dec	: Deceleration
e	: Exit
f	: Fuel
_g	: Gas
HP	: High pressure
in	: Intlet
LP	: Low pressure
out	: Outlet
s	: Static
std	: Standard atmospheric condition
t	: Total

Superscripts

T	: Transpose

References

[1] J. D. Mattingly, W. H. Heiser and D. T. Pratt, *Aircraft Engine Design*, Second Ed., AIAA, Reston, USA (2002).

[2] J. S. Litt, D. K. Frederick and T. H. Guo, The case for intelligent propulsion control for fast engine response, *NASA/TM-215668*, Cleveland, OH: Glenn Research Center, NASA (2009).

[3] C. D. Kong and S. C. Chung, Real time linear simulation and control for small aircraft turbojet engine, *KSME International Journal*, 13 (9) (1999) 656-666.

[4] Y. Gang and S. J. Guo, Reduced order H_∞/LTR method for aeroengine control system, *Chinease Journal of Aeronautics*, 17 (3) (2004) 129-135.

[5] H. Zhao, J. Liu and D. Yu, Approximate nonlinear modeling and feedback linearization control for aeroengines, *Journal of Engineering Gas Turbines and Power*, 133 (11) (2011) 21-31.

[6] L. Xiaofeng, S. Jing and Q. Yiwen, Design for aircraft engine multi-objective controllers with switching characteristics, *Chinease Journal of Aeronautics*, 27 (5) (2014) 1097-1110.

[7] J. Wang, G. Dimirovski and H. Yue, Finite-time regulation of two-spool turbofan engines with one shaft speed control, *Journal of Dynamic Systems, Measurement and Control*, 138 (8) (2016) 40-47.

[8] C. Chen and J. Zhao, A nonlinear switching control strategy of regulation and safety protection for aero-engines based on the equilibrium manifold expansion model, *Proc IMechE, Part I: Journal of Systems and Control Engineering* (2017) 1-14.

[9] I. Yazar, E. Kiyak, F. Kaliskan and T. H. Karakoc, Simulation-based dynamic model and speed controller design of a small-scale turbojet engine, *Aircraft Engineering and Aerospace Technology*, 90 (2) (2018) 351-358.

[10] H. A. Spang III and H. Brown, Control of jet engines, *Control Engineering Practice*, 7 (9) (1999) 1043-1059.

[11] J. Csank, R. D. May and J. S. Litt, Control design for a generic commercial aircraft engine, *NASA/TM-216811*, Cleveland, OH: Glenn Research Center, NASA (2010).

[12] S. Garg, Aircraft turbine engine control research at NASA glenn research center, *NASA/TM-217821*, Cleveland, OH: Glenn Research Center, NASA (2013).

[13] S. M. Tajalli and A. Tajalli, Thermodynamic simulation of two-shaft gas turbine to study invasive weeds optimization and min-max controller strategies considering air-cooled blades, *Journal of Mechanical Science and Technology*, 33 (2) (2019) 931-938.

[14] H. Richter, Multiple sliding modes with override logic: limit management in aircraft engine controls, *Journal of Guidance, Control and Dynamics*, 35 (4) (2012) 1132-1142.

[15] Y. Qi, W. Bao and C. Juntao, State-based switching control strategy with application to aeroengine safety protection, *ASCE Journal of Aerospace Engineering*, 28 (3) (2014) 1-11.

[16] A. Imani and M. Montazeri-Gh, Improvement of min-max limit protection in aircraft engine control: An LMI approach, *Journal of Aerospace Science and Technology*, 68 (2017) 214-222.

[17] J. Seok, I. Kolmanovsky and A. Girard, Coordinated model predictive control of aircraft gas turbine engine and power system, *Journal of Guidance, Control, and Dynamics*, 40 (10) (2017) 2538-2555.

[18] N. Gu and X. Wang, Model predictive controller design based on the linear parameter varying model method for a class of turboshaft engines, *2018 Joint Propulsion Conference, AIAA Propulsion and Energy Forum* (2018).

[19] M. V. Kothare, V. Balakrishnan and M. Morari, Robust constrained model predictive control using linear matrix inequalities, *Automatica*, 32 (10) (1996) 1361-1379.

[20] B. G. Vroemen, H. A. van Essen and A. A. van Steenhoven, Nonlinear model predictive control of a laboratory gas turbine installation, *Journal of Engineering Gas Turbines and Power*, 121 (4) (1999) 629-634.

[21] J. Mu, D. Rees and G. P. Liu, Advanced controller design for aircraft gas turbine engines, *Control Engineering Practice*, 13 (2005) 1001-1015.

[22] J. A. DeCastro, Rate-based model predictive control of turbofan engine clearance, *Journal of Propulsion and Power*, 23 (4) (2007) 804-813.

[23] P. Kai, F. Ding and Y. Fan, Active generalized predictive control of turbine tip clearance for aero-engines, *Chinease Journal of Aeronautics*, 26 (5) (2013) 1147-1155.

[24] H. Richter, *Advanced Control of Turbofan Engines*, First Ed., Springer, New York, USA (2012).

[25] H. Richter, A. Singaraju and J. S. Litt, Multiplexed predic-

tive control of a large commercial turbofan engine, *Journal of Propulsion and Power*, 31 (2) (2008) 273-281.
[26] T. Cheng, Hardware in the loop simulation of minitype turbojet engine digital control regulator, *Journal of Aerospace Power*, 19 (3) (2004) 383-386.
[27] W. Bao, Y. F. Sui and Z. M. Liu, Design and realization of hardware-in-the-loop simulation for turbofan-engine, *Journal of System Simulation*, 18 (6) (2006) 603-615.
[28] J. Lu, Y. Q. Guo and H. Q. Wang, Rapid prototyping real-time simulation platform for digital electronic engine control, *2nd International Symposium on. IEEE Systems and Control Aeronautics and Astronautics* (2008).
[29] M. Rezaei Darestani, M. Zareh, J. Roshanian and A. Khaki Sedigh, Verification of intelligent control of a launch vehicle with HILS, *Journal of Mechanical Science and Technology*, 25 (2) (2011) 523-536.
[30] X. Wang and L. Sun, Multi-loop controller design applied to the experiment of a load test bench, *Journal of Mechanical Science and Technology*, 29 (9) (2015) 3953-3960.
[31] M. Montazeri-Gh, M. Nasiri and S. Jafari, Real-time multi-rate HIL simulation platform for evaluation of a jet engine fuel controller, *Simulation Model Practice and Theory*, 19 (3) (2011) 996-1006.
[32] M. Montazeri-Gh, S. Abyaneh and S. Kazemnejad, Hardware-in-the-loop simulation of two-shaft gas turbine engine's electronic control unit, *Proc IMechE, Part I: Journal of Systems and Control Engineering*, 230 (6) (2016) 512-521.
[33] H. Rashidzadeh, S. M. Hosseinalipour and A. Mohammadzadeh, The SGT-600 industrial twin-shaft gas turbine modeling for mechanical drive applications at the steady state conditions, *Journal of Mechanical Science and Technology*, 29 (10) (2015) 4473-4481.
[34] Q. Yang, S. Li and Y. Cao, A new component map generation method for gas turbine adaptation performance simulation, *Journal of Mechanical Science and Technology*, 31 (4) (2017) 1947-1957.
[35] P. P. Walsh and P. Fletcher, *Gas Turbine Performance*, Blackwell Publishing Company (2004).
[36] M. Lichtsinder and Y. Levy, Jet engine model for control and real-time simulations, *Journal of Engineering Gas Turbines and Power*, 128 (4) (2006) 745-753.
[37] K. J. Keesman, *System Identification - An Introduction*, Springer-Verlag, London, UK (2011).
[38] D. K. Chaturvedi, *Modeling and Simulation of Systems using MATLAB and Simulink*, First Ed., CRC Press, Boca Raton, USA (2010).
[39] E. Mohammadi and M. Montazeri-Gh, A new approach to the gray-box identification of wiener models with the application of gas turbine engine modeling, *Journal of Engineering Gas Turbines and Power*, 137 (7) (2015) 11-23.
[40] A. J. Chipperfield, B. Bica and P. J. Fleming, Fuzzy scheduling control of a gas turbine aero-engine: A multiobjective approach, *IEEE Transactions on Industrial Electronics*, 49 (3) (2002) 536-548.
[41] M. Montazeri-Gh and A. Safari, Tuning of fuzzy fuel controller for aero-engine thrust regulation and safety considerations using genetic algorithm, *Journal of Aerospace Science and Technology*, 15 (2011) 183-192.
[42] M. Montazeri-Gh and S. Jafari, Evolutionary optimization for gain tuning of jet engine min-max fuel controller, *Journal of Propulsion and Power*, 27 (5) (2011) 1015-1023.
[43] M. Montazeri-Gh, E. Mohammadi and S. Jafari, Fuzzy-based gas turbine engine fuel controller design using particle swarm optimization, *Applied Mechanics and Materials*, 110-116 (2012) 3215-3222.
[44] M. Gen and R. Cheng, *Genetic Algorithms and Engineering Design*, First Ed., John Wiley and Sons, New York, USA (1997).
[45] M. Mitchell, *An Introduction to Genetic Algorithms*, Fifth Ed., Massachusetts Institute of Technology, Massachusetts, USA (1999).
[46] D. Jolevski and O. Bego, Model predictive control of gantry/bridge crane with anti-sway algorithm, *Journal of Mechanical Science and Technology*, 29 (2) (2015) 827-834.
[47] M. Abdullah and M. Idres, Constrained model predictive control of proton exchange membrane fuel cell, *Journal of Mechanical Science and Technology*, 28 (9) (2014) 3855-3862.
[48] E. F. Camacho and C. Bordons, *Model Predictive Control*, Second Ed., Springer-Verlag, London, UK (2007).
[49] L. Wang, *Model Predictive Control System Design and Implementation using MATLAB*, First Ed., Springer-Verlag, London, UK (2009).
[50] K. Ogata, *Modern Control Engineering*, Fifth Ed., Prentice Hall, New Jersey, USA (2010).

Morteza Montazeri-Gh is a Professor of Mechanical Engineering at Iran University of Science and Technology (IUST), in Tehran, Iran. He is also Director of the Systems Simulation and Control Laboratory at IUST since 1996. His current research is directed towards simulation, control and diagnosis of gas turbine engine as well as HIL test for performance evaluation of Engine ECU and FCU.

Ali Rasti is a Ph.D candidate in Mechanical Engineering Department of Iran University of Science and Technology (IUST), in Tehran, Iran. His research interests include model predictive control, HIL tests and gas turbine engine control system and fault diagnosis.