Abstract: Aero-engine control systems generally adopt centralized or distributed control schemes, in which all or most of the tasks of the control system are mapped to a specific processor for processing. The performance and reliability of this processor have a significant impact on the control system. Based on the aero-engine distributed control system (DCS), we propose a decentralized controller scheme. The characteristic of this scheme is that a network composed of a group of nodes acts as the controller of the system, so that there is no core control processor in the system, and the computation is distributed throughout the entire network. An LQR output feedback control is constructed using system input and output, and the control tasks executed on each node in the decentralized controller are obtained. The constructed LQR output feedback is equivalent to the optimal LQR state feedback. The primal-dual principle is used to tune the parameters of each decentralized controller. The parameter tuning algorithm is simple to calculate, making it conducive for engineering applications. Finally, the proposed scheme was verified by simulation. The simulation results show that a high-precision feedback gain matrix can be obtained with a maximum of eight iterations. The parameter tuning algorithm proposed in this paper converges quickly during the calculation process, and the constructed output feedback scheme achieves equivalent performance to the state feedback scheme, demonstrating the effectiveness of the design scheme proposed in this paper.

Keywords: decentralized controller; aero-engine; linear quadratic regulation (LQR); Q-learning; output feedback control construction; primal-dual

1. Introduction

Early aero-engine control systems adopt a centralized control scheme, where the controller is connected to analog sensors and actuators via cables [1]. The controller executes tasks such as calculation of control law, analog-to-digital conversion (ADC), and digital-to-analog conversion (DAC), resulting in a large workload for the controller. Meanwhile, the centralized control scheme increases the difficulty and cost of upgrading and maintaining the aero-engine throughout its entire lifespan [2]. Moreover, the cables should be arranged around the rotating components. The huge quantity of cables reduces the thrust-to-weight ratio of the aero-engine. To overcome the drawbacks of the centralized control scheme, distributed control schemes for aero-engines have emerged [3].

A distributed control system (DCS) is a control architecture applied in process control or plant control [4–7]. In an aero-engine DCS, traditional analog sensors and actuators have been replaced by intelligent digital sensors and actuators. The components in the control system are connected through a universal and standardized communication interface [8,9]. Currently, aero-engine related enterprises and scientific research institutions have a basic consensus on the “functional decentralization” of the DCS, that is, some of the functions of the controller in the centralized control system, such as signal conditioning and status monitoring, are executed by intelligent sensors and actuators, while the central controller focuses on executing the core control functions of the system [10]. A simple and robust digital bus replaces the large number of cables connecting the aero-engine control components to the avionics [11,12]. Compared with centralized control, an aero-engine...
DCS possesses the following advantages [13]: (1) Digital intelligent sensors and actuators reduce the workload of the central controller; (2) The digital bus reduces the weight of the aero-engine control system; (3) The modular design scheme reduces the difficulty and cost of system maintenance and upgrades.

Although the aero-engine DCS has the above advantages, it still has some inherent disadvantages due to the existence of core nodes.

1. With the improvement of aero-engine performance, the function and complexity of control tasks have greatly increased, which has increased the workload of controller. The control system needs to use high-performance, multi-core processors as its controller, which in turn puts relatively high demands on the thermal management system of the aero-engine.

2. The amount of software code in aero-engine control systems is increasing rapidly, which significantly impacts the software reliability.

3. The core control tasks of the aero-engine, such as control-law calculation, are executed in the central controller. The central controller determines the performance of the entire aero-engine control system, and its failure or damage has significant impact on the aero-engine or even the aircraft.

Decentralizing the central controller is an effective way to overcome the shortcomings of the aero-engine DCS.

Output feedback and state feedback are both utilized in designing closed-loop control systems. Generally, output feedback is easier to implement in projects, but state feedback offers better performance. However, state feedback requires full system states. In the case of aero-engines, there are numerous state variables, some of which are hard to measure. Furthermore, some state variables may possess only mathematical meaning without a physical value. Therefore, the application of the aero-engine state feedback is restricted.

Linear quadratic regulation (LQR) can derive the optimal control law for linear state feedback and facilitate the implementation of optimal closed-loop control [14]. Optimum LQR control design is achieved by solving the LQR problem [15–17]. The conventional approach to the LQR problem involves solving the Bellman equation through dynamic programming. For linear control system design, it is transformed into the solution of the algebraic Riccati equation (ARE). Both the Bellman equation and ARE are challenging to solve [18,19], and it is difficult to use a conventional microprocessor for real-time calculation [20–24].

The contributions of this paper are as follows. Based on the aero-engine DCS, we propose a decentralized controller architecture. An LQR output feedback control was constructed through system input and output, and the control tasks executed on each node in the decentralized controller are obtained. The constructed LQR output feedback is equivalent to the optimal LQR state feedback. Finally, a simple linear iterative algorithm for solving the LQR problem is introduced. It is demonstrated that the calculation of the decentralized control tasks is simple and the LQR output feedback provides equivalent performance to the LQR optimal state feedback. Meanwhile, the LQR solving algorithm is simple to calculate and fast to converge.

The remainder of this paper is organized as follows. Section 2 introduces an aero-engine linear state space model. Section 3 introduces the output feedback scheme of the decentralized control system. Section 4 introduces an output feedback scheme based on the Q-Learning LQR, equivalent to the state feedback scheme. Section 5 introduces a primal-dual principle-based parameter-tuning algorithm. Section 6 presents the simulation of the scheme proposed in this paper. Finally, Section 7 is the last section of this paper, providing the summary and conclusion.

2. Aero-Engine Model

Turbofan engines are widely used in both military and civil aviation [25,26] due to their high efficiency. However, the technical difficulty is also high [27,28], and they require advanced control systems. Here, we consider the turbofan engine as an example to
study [29]. Figure 1 is the schematic diagram of a turbofan engine. We first introduced the nonlinear modeling of the turbofan engine. Then, we derived a linearized model based on the nonlinear model and obtained the structure of the linearized model. Finally, we introduced the identification of the model parameters based on the input and output data of the engine.

2.1. Nonlinear Modeling

Due to the strong nonlinear characteristics of aero-engines, the nonlinear model of the aero-engines is briefly analyzed in this section. Based on fundamental physical principles, we model the various components of the turbofan engine. Then, using aerothermodynamic principles among components and the common working conditions of associated components, we establish joint equations. The following is a brief introduction to the physical equations involved.

(1) Intake

The intake is the first part where the air flows through the aero-engine. The inlet temperature $T_2$ and the pressure $P_2$ of the aero-engine are:

\[
\begin{align*}
    T_2 &= T_1 \left( 1 + \frac{k_1 - 1}{2} \cdot Ma^2 \right) \\
    P_2 &= \sigma_1 P_1 \left( 1 + \frac{k_1 - 1}{2} \cdot Ma^2 \right)^{k_1 / (k_1 - 1)}
\end{align*}
\]  

(1)

where $T_1$ and $P_1$ are the total air temperature and the total pressure at the intake, respectively, $k_1$ is the air adiabatic index, $\sigma_1$ is the total pressure recovery coefficient of the intake, $Ma$ is Mach number.

(2) Fan

The fan outlet temperature $T_{21}$ and the pressure $P_{21}$ of the aero-engine are:

\[
\begin{align*}
    T_{21} &= T_2 \left( 1 + \frac{k_F - 1}{\pi_F \eta_F} \right) \\
    P_{21} &= \pi_F P_2
\end{align*}
\]  

(2)

where $k_F$ is the air adiabatic index, $\pi_F$ is the fan pressure ratio, $\eta_F$ is the efficiency of the fan.

(3) Compressor

The compressor outlet temperature $T_3$ and the pressure $P_3$ are:

\[
\begin{align*}
    T_3 &= T_{22} \left( 1 + \frac{k_C - 1}{\pi_C \eta_C} \right) \\
    P_3 &= \pi_C P_{22}
\end{align*}
\]  

(3)
where \(T_{22}\) and \(P_{22}\) are the temperature and pressure of the compressor inlet, respectively, \(k_C\) is the air adiabatic index, \(\pi_C\) is the compressor pressure ratio, \(\eta_C\) is the efficiency of the compressor.

4) Combustion chamber

The outlet pressure of the combustion chamber \(P_4\) can be expressed by the following equation:

\[
P_4 = \sigma_3 P_4 \tag{4}
\]

where \(\sigma_3\) represents the total pressure recovery coefficient of the combustion chamber.

5) High-pressure turbine

The outlet temperature \(T_{41}\) and pressure \(P_{41}\) of the high-pressure turbine can be obtained as:

\[
\begin{cases}
  T_{41} = q_{mg,TH} + C_{HPTCool}q_{ma,TotalTcool} + q_{ma,TotalTcool} + q_{ma,TH,Total} \\
  P_{41} = \frac{P_4}{\pi_{TH}}
\end{cases} \tag{5}
\]

where \(q_{mg,TH}\) is the gas flow converted by the high-pressure turbine, \(T_4\) is the outlet temperature of the combustion chamber, \(C_{HPTCool}\) is the proportion coefficient of the high-pressure compressor bleed air used to cool the high-pressure turbine, \(q_{ma,TotalTcool}\) is the total flow of added air, \(q_{ma,TH,Total}\) is the outlet air flow of the high-pressure turbine, \(\pi_{TH}\) is the high-pressure turbine pressure ratio, \(\eta_{TH}\) is the efficiency of the high-pressure turbine, \(k_{TH}\) is the gas adiabatic index.

6) Low-pressure turbine

The outlet temperature \(T_5\) and pressure \(P_5\) of the low-pressure turbine are calculated by the following equations:

\[
\begin{cases}
  T_5 = \frac{q_{mg,TL}T_{42} + C_{LPTCool}q_{ma,TotalTcool}}{q_{mg,TL,Total}} \left( 1 - \left( 1 - \frac{1}{\pi_{TL}} \right) \eta_{TL} \right) \\
  P_5 = \frac{P_4}{\pi_{TL}}
\end{cases} \tag{6}
\]

where \(q_{mg,TL}\) is the gas flow converted by the high-pressure turbine, \(T_{42}\) is the inlet temperature of the low-pressure turbine, \(C_{LPTCool}\) is the proportion coefficient of the high-pressure compressor bleed air used to cool the low-pressure turbine, \(q_{ma,TL,Total}\) is the outlet air flow of the low-pressure turbine, \(\pi_{TL}\) is the low-pressure turbine pressure ratio, \(\eta_{TL}\) is the efficiency of the low-pressure turbine, \(k_{TL}\) is the gas adiabatic index.

7) Bypass duct

The temperature \(T_6\) and the pressure \(P_6\) of the outlet of the bypass duct are:

\[
\begin{cases}
  T_6 = T_{21} \\
  P_6 = \sigma_4 P_{21}
\end{cases} \tag{7}
\]

where \(T_{21}\) and \(P_{21}\) are the outlet temperature and pressure of the fan, respectively, \(\sigma_4\) is the total pressure recovery coefficient of the bypass duct.

8) Mixer

The gas flow \(q_{mg,7}\) at the outlet of the mixer is the sum of the air flow \(q_{ma,6}\) at the outlet of the bypass duct and the gas flow \(q_{mg,5}\) at the outlet of the low-pressure turbine. Let \(\sigma_5\)
denotes the total pressure recovery coefficient of the mixer, then the physical parameters at the outlet of the mixer are:

\[
\begin{align*}
q_{mg,7} &= q_{mg,5} + q_{ma,6} \\
h_7 &= \frac{q_{mg,5} + h_{ma,6}}{q_{mg,7}} \\
P_7 &= \sigma_5 \frac{P_{mg,5} + P_{ma,6}}{q_{mg,7}}
\end{align*}
\]

(8)

where \( h_5, h_6, \) and \( h_7 \) are the specific enthalpy of the gas at the outlet of the low-pressure turbine, air at the outlet of the bypass duct, and gas at the outlet of the mixer, respectively. From \( h_7 \), it is easy to obtain the temperature \( T_7 \) at the outlet of the mixer.

(9) Nozzle

The gas flow \( q_{mg,8} \) at the nozzle outlet is:

\[
q_{mg,8} = \frac{P_8 A_8 q(\lambda_8)}{\sqrt{T_7}}
\]

(9)

where \( K_q \) is the state coefficient of the nozzle, \( q(\lambda_8) \) refers to the function related to the characteristics of the bypass duct and the core duct, \( A_8 \) is the sectional area of the nozzle, \( T_7 \) is the outlet temperature of the mixer.

The high-pressure turbine drives the high-pressure compressor of the turbofan engine to form the high-pressure rotor. The low-pressure turbine drives the fan to form the low-pressure rotor. After passing through the intake and fan, the gases enter the bypass duct and the core duct, respectively. The common working equations of the turbofan engine are obtained according to the flow, pressure, and power balance between the engine components.

(1) High-pressure rotor power balance

\[
P_H = P_{CH} + P_{ex,H} + D_H \left( \frac{dn_H}{dt} \right)
\]

(10)

where \( P_H \) denotes the high-pressure turbine power, \( P_{CH} \) denotes the high-pressure compressor power, \( P_{ex,H} \) denotes the power lost by transmission friction force, \( D_H \left( \frac{dn_H}{dt} \right) \) denotes the high-pressure rotor acceleration power. Furthermore, \( D_H = (\pi/30)^2 J_H n_H \), \( J_H \) denotes the moment of inertia of the high-pressure rotor, \( n_H \) denotes the speed of the high-pressure rotor.

When the aero-engine is stable, \( dn_H/dt = 0 \). Ignoring the power lost by transmission friction force \( P_{ex,H} \), Equation (10) can be simplified as:

\[
P_H = P_{CH}
\]

(11)

(2) Low-pressure rotor power balance

Similar to the high-pressure rotor, when the aero-engine is stable, ignoring the power lost by transmission friction force, we can obtain:

\[
P_L = P_{CL}
\]

(12)

where \( P_L \) denotes the low-pressure turbine power, \( P_{CL} \) denotes the low-pressure compressor power.

(3) Fan air flow balance

After flowing through the fan, the gas is divided into two parts, one part enters the bypass duct and the other enters the high-pressure compressor. Ignoring the gas flow loss, then:

\[
q_{maF} = q_{maC} + q_{ma6}
\]

(13)
where \( q_{maF} \) is the air flow from the fan, \( q_{maC} \) is the air flow into the high-pressure compressor, \( q_{ma6} \) is the air flow into the bypass duct.

4) High-pressure turbine gas flow balance

\[
q_{mgA} = q_{maC} + q_{mf}
\]

where \( q_{mgA} \) is the gas flow into the high-pressure turbine, \( q_{maC} \) is the gas flow from the high-pressure compressor, \( q_{mf} \) is the fuel flow.

5) Low-pressure turbine gas flow balance

The low-pressure turbine inlet gas flow is equal to the high-pressure turbine outlet gas flow \( q_{mgA2} \):

\[
q_{mgA2} = q_{mgA} + C_{HPTCool} q_{maC, totalTcool}
\]

where \( C_{HPTCool} \) is the proportion coefficient of the high-pressure compressor bleed air used to cool the high-pressure turbine, \( q_{maC, totalTcool} \) is the total air flow into the high-pressure turbine.

6) Nozzle gas flow balance

\[
q_{mg,8} = q_{mg,5} + q_{ma,16}
\]

where \( q_{mg,8} \) is the gas flow at the inlet of the nozzle, \( q_{mg,5} \) is the gas flow at the low-pressure turbine outlet, \( q_{ma,16} \) is the gas flow into the bypass duct.

2.2. Linear State Space Model

We derived the component-level nonlinear model of the aero-engine, along with the common working equations. However, the current control system is primarily designed based on a linear system. Therefore, a linear model of the aero-engine is needed. In this section, the system identification method is applied to obtain the model of the aero-engine. Firstly, the structural identification of the model is carried out.

The state equation and the output equation of the aero-engine nonlinear state space model are:

\[
\begin{align*}
\dot{x} &= f(x, u) \\
y &= g(x, u)
\end{align*}
\]

where \( x \) is the state vector, \( y \) is the output vector, \( u \) is the control input, and the dimensions of \( x, y, u \) are \( n, m, r \), respectively. It is assumed that the state equation and the output equation of the aero-engine are differentiable within the flight envelope.

Based on the principle of rotor dynamics, taking the aero-engine as a rigid body, let \( \dot{w} \) denote the angular acceleration, \( \Delta Q \) denote the torque difference between the two shafts, and \( I \) denote the mass moment of inertia. The acceleration of the system can be obtained from Newtonian mechanics as:

\[
\dot{w} = \frac{\Delta Q}{I}
\]

Let \( N \) denote the aero-engine shaft speed, \( f \) is the function between shaft speed and fuel quantity, the aero-engine shaft dynamic equation is:

\[
\dot{n} = \frac{f(N, W_f)}{I}
\]

Let \( J_H \) denote the moment of inertia of the high-pressure rotor, \( \Delta M_H \) the residual torque of the high-pressure rotor, \( J_L \) the moment of inertia of the low-pressure rotor, \( \Delta M_L \)
the residual torque of the low-pressure rotor, then, there are two rotor moment balance equations for the aero-engine:

\[
\begin{align*}
\frac{\partial}{\partial t} J_H \frac{dN_C}{dt} &= \Delta M_H \\
\frac{\partial}{\partial t} J_L \frac{dN_F}{dt} &= \Delta M_L 
\end{align*}
\]  
(20)

The residual torque of the high-pressure rotor and the residual torque of the low-pressure rotor are:

\[
\begin{align*}
\Delta M_H &= M_{TH} - M_{CH} = \Delta M_H \cdot \left( N_C, N_F, W_f \right) \\
\Delta M_L &= M_{TL} - M_{CL} = \Delta M_L \cdot \left( N_C, N_F, W_f \right)
\end{align*}
\]  
(21)

Linearization is carried out at a steady-state operating point \((x_0, u_0)\) of the aero-engine:

\[
\begin{align*}
\Delta M_H &= \frac{\partial \Delta M_H}{\partial N_C} \bigg|_{(x_0, u_0)} \Delta N_C + \frac{\partial \Delta M_H}{\partial N_F} \bigg|_{(x_0, u_0)} \Delta N_F + \frac{\partial \Delta M_H}{\partial W_f} \bigg|_{(x_0, u_0)} \Delta W_f \\
\Delta M_L &= \frac{\partial \Delta M_L}{\partial N_C} \bigg|_{(x_0, u_0)} \Delta N_C + \frac{\partial \Delta M_L}{\partial N_F} \bigg|_{(x_0, u_0)} \Delta N_F + \frac{\partial \Delta M_L}{\partial W_f} \bigg|_{(x_0, u_0)} \Delta W_f 
\end{align*}
\]  
(22)

From Equations (20) and (22), the linearized shaft dynamic equation is:

\[
\begin{align*}
\dot{N}_C &= a_{11} \cdot \Delta N_C + a_{12} \cdot \Delta N_F + b_{11} \cdot \Delta W_f \\
\dot{N}_F &= a_{21} \cdot \Delta N_C + a_{22} \cdot \Delta N_F + b_{21} \cdot \Delta W_f
\end{align*}
\]  
(23)

where

\[
\begin{align*}
a_{11} &= \left( \frac{30}{\pi f_H} \right) \cdot \frac{\partial \Delta M_H}{\partial N_C} \bigg|_{(x_0, u_0)} \\
a_{12} &= \left( \frac{30}{\pi f_H} \right) \cdot \frac{\partial \Delta M_H}{\partial N_F} \bigg|_{(x_0, u_0)} \\
a_{21} &= \left( \frac{30}{\pi f_L} \right) \cdot \frac{\partial \Delta M_L}{\partial N_C} \bigg|_{(x_0, u_0)} \\
a_{22} &= \left( \frac{30}{\pi f_L} \right) \cdot \frac{\partial \Delta M_L}{\partial N_F} \bigg|_{(x_0, u_0)} \\
b_{11} &= \left( \frac{30}{\pi f_H} \right) \cdot \frac{\partial \Delta M_L}{\partial W_f} \bigg|_{(x_0, u_0)} \\
b_{22} &= \left( \frac{30}{\pi f_L} \right) \cdot \frac{\partial \Delta M_L}{\partial W_f} \bigg|_{(x_0, u_0)}
\end{align*}
\]

Equation (23) can be written as:

\[
\begin{align*}
\dot{x} &= Ax + Bu \\
y &= Cx + Du
\end{align*}
\]  
(24)

where \(x, y, u\) are the state vector, the output vector, and the control input of the aero-engine state space model, respectively. The model described by Equation (24) is a second-order system, and the structural identification of the system model is completed. The linear model of the turbofan engine is represented as a second-order system. Next, we introduce the method for identifying the parameters of the system model.

MATLAB software was used for identification. By running the open-loop control system of the engine model and providing an input at a certain steady-state point, the input and output data of the model can be saved when the system stabilizes, thus obtaining the data from the engine at that steady-state point. Then, a small step signal based on the input data is added to each variable, and the input and output data are saved when the system reaches a steady state. By subtracting the input and output data of the steady-state
point from the input and output data of the system’s step response, incremental data can be obtained that can be used for identification. After preprocessing the data, it is imported into the system identification toolbox, and the N4SID method is used to identify the linear model of the system.

Based on the method of system identification described above, we provide two linear models of the turbofan engine.

The first is a single-variable model for the turbofan engine. Here, taking the high-pressure rotor-speed control as an example, the high-pressure rotor speed \( y = N_h \) is the controlled variable of the control system, and the main fuel flow \( u = W_f \) is the control input. The system identification in this section is based on the input and output data of the system, i.e., the model of the turbofan engine is a black-box model, and the specific meaning of the state variables in the model is not limited. At the steady-state point where the relative speed is 85\%, height \( H = 0 \), mach number \( M_a = 0 \), and the sample time is 20 ms, the discrete time linear state space model of the turbofan engine obtained through the system identification is:

\[
\begin{align*}
   x[k+1] &= G_S x[k] + H_S u[k] \\
   y[k] &= C_S x[k] + D_S u[k]
\end{align*}
\]

(25)

where \( x = [x_1 \ x_2]^T \) is the state variable, which has no physical meaning. Because this is not a physical variable, it cannot be directly measured by the sensors. Therefore, the model established by Equation (25) cannot theoretically realize state feedback control by conventional methods.

The coefficient matrix of Equation (25) is:

\[
\begin{align*}
   G_S &= \begin{bmatrix} \xi_{11} & \xi_{12} \\ \xi_{21} & \xi_{22} \end{bmatrix} = \begin{bmatrix} 0.9854 & 0.0156 \\ -1.3474 & 0.5935 \end{bmatrix} \\
   H_S &= \begin{bmatrix} h_{11} \\ h_{21} \end{bmatrix} = \begin{bmatrix} 0.1875 \\ 19.3024 \end{bmatrix} \\
   C_S &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
   D_S &= 0
\end{align*}
\]

The second presented system is a multi-variable model for the turbofan engine. Here, the target of the control system is set to achieve a change in the thrust of the turbofan engine according to the angle of the throttle lever. As thrust cannot be measured through the sensors, parameters that are highly correlated with thrust are selected to represent the thrust characteristics. Here, according to the aero-engine principle, the controlled variable can be selected as high-pressure rotor speed \( n_H \) and turbine pressure ratio \( P_iT = \pi_{TL} / \pi_{TH} \), the reference input is selected as fuel flow \( W_f \) and nozzle area \( A_8 \). At the steady-state point with the power lever angle \( PLA = 35^\circ \), height \( H = 0 \), mach number \( M_a = 0 \), sample time 20 ms, the discrete time linear state space model of the turbofan engine obtained through the system identification is:

\[
\begin{align*}
   x[k+1] &= G_M x[k] + H_M u[k] \\
   y[k] &= C_M x[k] + D_M u[k]
\end{align*}
\]

(26)

where \( u = [W_f \ A_8]^T \) is the reference input, \( y = [n_H \ P_iT]^T \) is the controlled variable, \( x = [x_1 \ x_2]^T \) is the state variable, which has no physical meaning.
The coefficient matrix of Equation (26) is:

\[
\begin{aligned}
G_M &= \begin{bmatrix}
\xi_{11} & \xi_{12} \\
\xi_{21} & \xi_{22}
\end{bmatrix} = \begin{bmatrix}
0.9609 & -0.0679 \\
-0.0028 & 0.9098
\end{bmatrix} \\
H_M &= \begin{bmatrix}
h_{11} & h_{12} \\
h_{21} & h_{22}
\end{bmatrix} = \begin{bmatrix}
0.3305 & 0.1688 \\
0.0758 & 0.2097
\end{bmatrix} \times 10^{-3} \\
C_M &= \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix} \\
D_M &= 0
\end{aligned}
\]

In this section, we report the single-variable and multi-variable linear models of the turbofan engine, respectively. Their state variables have no practical physical meaning. Because the value of the state variables cannot be measured using sensors, the state feedback scheme that relies on sensor measurements is unavailable.

3. Output Feedback of Decentralized Control System

The decentralized control system is proposed based on the development of communication technology, computer technology, data storage technology, sensor technology, and other supporting technologies. The traditional aero-engine DCS architecture is shown in Figure 2 [30]. In the DCS, the sensors and actuators are typically digital and smart, and are connected by bus to the controller. Signal conditioning, ADC, DAC, etc., are executed by smart sensors and smart actuators. Meanwhile, the controller focuses on core control tasks such as control-law calculation. With the development of the aero-engine, the tasks of the controller become more complex, and new-high performance microprocessors are needed to act as controllers. Therefore, the performance and reliability of the microprocessor acting as the controller have significant impact on the entire aero-engine control system.

![Figure 2. Overview of a traditional aero-engine DCS.](image)

The design scheme of the decentralized controller proposed in this paper uses a group of low-price, high-reliability, but low-performance microprocessor nodes to form a decentralized network, as shown in Figure 3. Each microprocessor node is equipped with computing, memory, and communication capabilities and can transmit data to each other, as represented by the circles in Figure 3. Using this decentralized network to replace the DCS controller, as shown in Figure 2, the network acts as the controller of the control system, representing a decentralized aero-engine control scheme. Some microprocessors in the network possess the capability of bus communication, enabling the entire decentralized...
network to communicate with the smart sensors and actuators in the aero-engine. In this paper, we refer to this kind of control system as the software-defined control system (SDCS). In Figure 3, the green circles represent the regular nodes, and the blue circles represent the nodes with bus-communication capability.

![Figure 3. Overview of the decentralized controller.](image)

The system control tasks are virtualized into multiple sub-tasks, named the virtual control tasks (VCTs). Each VCT establishes a mapping relationship with the microprocessors in the network. Since each VCT is part of the system control task, the task of each decentralized microprocessor is part of the controller task in DCS, which greatly reduces the workload of each microprocessor, and allows the use of low-price, high-reliability, low-performance microprocessors. At the same time, the amount of software code in each microprocessor is greatly reduced, which can increase the software reliability of the microprocessors.

Despite typically having inferior performance compared with state feedback, output feedback has been widely utilized due to its good economy and feasibility. Figure 4 is the structural block diagram of the output feedback system.

![Figure 4. Output feedback control.](image)

The dynamic equation of the feedback system is:

\[
\begin{align*}
    x[k+1] &= Gx[k] + Hu[k] \\
    y[k] &= Cx[k] \\
    u[k] &= Cy[k]
\end{align*}
\]  

where \( x[k] \in \mathbb{R}^n \) denotes the internal states, \( u[k] \in \mathbb{R}^m \) denotes the input of the system, \( y[k] \in \mathbb{R}^p \) denotes the output of the system, \( G \in \mathbb{R}^{n \times n} \) is the system matrix, \( H \in \mathbb{R}^{n \times m} \) is the control matrix, \( C \in \mathbb{R}^{p \times n} \) is the output matrix, \( F \in \mathbb{R}^{m \times p} \) is the feedback gain matrix.
In this section, \( \gamma = \{v_1, v_2, \ldots, v_n\} \) denotes all of the nodes in the network, \( \gamma_A = \{va_1, va_2, \ldots, va_m\} \) is the set of the nodes that send data to the actuators, \( \gamma_S = \{vs_1, vs_2, \ldots, vs_p\} \) is the set of the nodes that receive sensor measurements.

As shown in Figure 5, the network acts as the system output feedback controller, i.e., the feedback \( F \) in Figure 4, \( sy_j \) denotes the sensor measuring the output, \( y_j \) is its measured value, i.e., the system output. Upon receiving sensor measurements, the network transmits the data to the system input after calculation. The system description of the nodes in the network during the running process is:

\[
u_i[k] = \sum_{y_j \in \gamma_S} f_{ij} y_j[k]
\]

(28)

![Figure 5](image)

Figure 5. The network acts as the output feedback controller.

The linear weighting process of the entire network can be expressed as:

\[
u[k] = \begin{bmatrix} f_{11} & f_{12} & \cdots & f_{1p} \\ f_{21} & f_{22} & \cdots & f_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ f_{m1} & f_{m2} & \cdots & f_{mp} \end{bmatrix} y[k] = Fy[k]
\]

(29)

In Figure 5, the state space of the plant is:

\[
x[k+1] = Gx[k] + Hu[k] \\
y[k] = Cx[k]
\]

(30)

Then, the state space description of the control system in Figure 5 is:

\[
x[k+1] = (G + HFC)x[k] \\
y[k] = Cx[k]
\]

(31)

In this section, we have briefly introduced the SDCS, and derived the model of the SDCS output feedback. In the next section, based on the LQR output feedback, we obtain the VCTs executed by the microprocessor.
4. Q-Learning Based LQR Output Feedback Control

The characteristic of the SDCS architecture is that the controller is decentralized. Therefore, it is required that the tasks executed on each node, i.e., VCTs, are also decentralized. The objective of this section is to obtain the VCTs executed on these decentralized nodes.

4.1. General LQR Problem Solving

The objective of the control system’s state equation derived in Section 3 is to determine the optimal control sequence \( u[k] \). When controllability and observability conditions of the system are satisfied, the following cost function can be minimized:

\[
J = \sum_{i=0}^{+\infty} j(x[i], u[i])
\]

where \( j \) denotes the one-step cost of executing control \( u \) in the state \( x \), and \( J \) denotes the long-term cost with an expected optimal cost as:

\[
J^* = \min \sum_{i=0}^{+\infty} j(x[i], u[i])
\]

The optimal control sequence is:

\[
u^*[k] = \arg \min \sum_{i=0}^{+\infty} j(x[i], u[i])
\]

For the LQR problem:

\[ j(x[k], u[k]) = x^T[k]Qx[k] + u^T[k]Ru[k] \]

where \( Q \in \mathbb{R}^{p \times p}, R \in \mathbb{R}^{m \times m} \), and meet \( Q = Q^T \geq 0, R = R^T > 0 \).

According to the infinite time Bellman equation:

\[
I(x[k]) = \min \{ j(x[k], u[k]) + I(x[k+1]) \}
\]

Suppose the Bellman Equation (36) has the form of \( I(x[k]) = x[k]^TPx[k] \), and satisfies \( P = P^T > 0 \):

\[
I(x[k+1]) = (Gx[k] + Hu[k])^TP(Gx[k] + Hu[k])
\]

From Equations (36) and (37):

\[
x^T P x = \min_{r} \left[ x^T \left( Q + G^T P G \right) x + 2x^T G^T P H u + u^T R u + u^T H^T P H u \right]
\]

When the system meets the controllability condition \((G, H)\) and observability condition \((G, \sqrt{Q})\), then \( \sqrt{Q^T} \sqrt{Q} = Q \), \( Q = C^T Q C \). Taking the partial derivative of the expression inside the square brackets in Equation (38) \( \partial / \partial r = 0 \), the optimal state feedback of the system is:

\[
u^* = - \left( R + H^T P H \right)^{-1} H^T P G x
\]

It can be obtained from Equations (38) and (39):

\[
x^T P x = x^T \left( Q + G^T P G - G^T P H \left( R + H^T P H \right)^{-1} H^T P G \right) x
\]

Then,

\[
P = Q + G^T P G - G^T P H \left( R + H^T P H \right)^{-1} H^T P G
\]
The above method is a conventional LQR problem-solving method based on the Bellman equation. Equation (41) is a discrete-time ARE. It can be concluded that the solved $P$ is unique.

4.2. Construction of the Output Feedback Control

The cost function of Equation (33) is rewritten into the form of the Bellman equation,

$$I(x[k]) = f(x[k], u[k]) + I(x[k+1])$$

(42)

We use the same Q function in [18]

$$Q(x[k], r[k]) = f(x[k], u[k]) + I(x[k+1])$$

(43)

In the LQR problem, the Q-function above can be written as,

$$Q(x[k], u[k]) = x^T[k]Qx[k] + u^T[k]Ru[k] + x^T[k+1]P_{x}[x[k+1]$$

(44)

Let the $I^*$ denote the optimal cost function, $F^*$ denote the optimal feedback strategy, then the Q-function can be written as:

$$Q^*(x[k], u[k]) = f(x[k], u[k]) + I^*(x[k+1])$$

(45)

The optimal strategy is:

$$F^* = \arg \min_x Q^*(x[k], u[k])$$

(46)

The optimal LQR controller $u^*$ is obtained by solving $(\partial / \partial r) Q^* = 0$, and the result is consistent with the result obtained based on the Bellman equation.

In order to develop a new output feedback Q-function, the internal state of the system is parameterized and constructed. Based on the state space description of Equation (30), and the recursive equation of the discrete-time equation:

$$x[k - N + 1] = Gx[k - N] + Hu[k - N]$$


$$\vdots$$

$$x[k - N + N] = x[k] = G^N x[k - N] + G^{N-1}Hu[k - N] + \cdots + G^2Hu[k - 3] + GHu[k - 2] + Hu[k - 1]$$

(47)

It is easy to obtain:

$$\begin{bmatrix}
  y[k-N] \\
  y[k-N+1] \\
  \vdots \\
  y[k-2] \\
  y[k-1]
\end{bmatrix}
= \begin{bmatrix} C \\ CG \end{bmatrix} x[k-N] + \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 & 0^T \\ CH & 0 & \cdots & 0 & 0 & 0 \\
  \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
 CG^{N-2} & CG^{N-2} & \cdots & CH & 0 & 0 \\
 CG^{N-1} & CG^{N-1} & \cdots & CGH & CH & 0 \\
\end{bmatrix} \begin{bmatrix} u[k-N] \\ u[k-N+1] \\ \vdots \\ u[k-2] \\ u[k-1]
\end{bmatrix}$$

(48)
Let
\[
V_N \triangleq \begin{bmatrix} C^T & (CG)^T & \cdots & (CG^{N-2})^T & (CG^{N-1})^T \end{bmatrix}^T
\]
\[
U_N = \begin{bmatrix} G^{N-1}H & G^{N-2}H & \cdots & GH & H \end{bmatrix}
\]
\[
T_N = \begin{bmatrix}
0 & 0 & \cdots & 0 & 0 & 0 \\
CH & 0 & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
CG^{N-3}H & CG^{N-2}H & \cdots & CH & 0 & 0 \\
CG^{N-2}H & CG^{N-3}H & \cdots & CGH & CH & 0
\end{bmatrix}
\]
\[
\bar{y}[k-N,k-1] = \begin{bmatrix}
y[k-N] \\
y[k-N+1] \\
\vdots \\
y[k-2] \\
y[k-1]
\end{bmatrix}
\]
\[
\bar{u}[k-N,k-1] = \begin{bmatrix}
u[k-N] \\
u[k-N+1] \\
\vdots \\
u[k-2] \\
u[k-1]
\end{bmatrix}
\]

Then, the Equations (47) and (48) can be written as:
\[
x[k] = G^N x[k-N] + U_N \bar{u}[k-N,k-1] \tag{49}
\]
\[
\bar{y}[k-N,k-1] = V_N^T \bar{y}[k-N,k-1] + U_N - C^N \left( V_N^T V_N \right)^{-1} V_N^T T_N \bar{u}[k-N,k-1] = M_y \bar{y}[k-N,k-1] 
\]
\[
+ M_u \bar{u}[k-N,k-1] = x[k] \tag{50}
\]

The parameters in the above equation are determined by the system coefficients, which require complete system information. However, it is often difficult to obtain the complete system information. Therefore, we designed an optimal output feedback control strategy based on Q-learning, which does not require the complete system information. From the Equations (44) and (51):
\[
Q(x[k], u[k]) = \begin{bmatrix}
\bar{u}[k-N,k-1] \\
\bar{y}[k-N,k-1] \\
\end{bmatrix}^T \begin{bmatrix}
L_{11} & L_{12} & L_{13} \\
L_{21} & L_{22} & L_{23} \\
L_{31} & L_{32} & L_{33}
\end{bmatrix} \begin{bmatrix}
\bar{u}[k-N,k-1] \\
\bar{y}[k-N,k-1] \\
\end{bmatrix}
\]
\[
\tilde{z}[k] = \begin{bmatrix}
\bar{u}^T [k-N,k-1] \\
\bar{y}^T [k-N,k-1] \\
u^T [k]
\end{bmatrix}^T 
\]
\[
L = \begin{bmatrix}
L_{11} & L_{12} & L_{13} \\
L_{21} & L_{22} & L_{23} \\
L_{31} & L_{32} & L_{33}
\end{bmatrix}
\]
The Equation (52) can be written as:

\[ Q(x[k], u[k]) = z^T[k]Lz[k] \]  

\[ (53) \]

where

\[ L_{11} = M_u^T(Q + G^T PG)M_u \]
\[ L_{12} = M_u^T(Q + G^T PG)M_y \]
\[ L_{13} = M_u^T G^T PH \]
\[ L_{21} = M_y^T(Q + G^T PG)M_u \]
\[ L_{22} = M_y^T(Q + G^T PG)M_y \]
\[ L_{23} = M_y^T G^T PH \]
\[ L_{31} = H^T PG M_u \]
\[ L_{32} = H^T PG M_y \]
\[ L_{33} = R + H^T PH \]

Since \( Q = Q^T \), \( R = R^T \), \( P = P^T \), then, \( L = L^T \)

Equation (53) is a new Q-function relating to the LQR problem of system input and output data. Minimize the \( Q(x[k], u[k]) \) in Equation (53) by solving \( \partial Q / \partial r = 0 \). Then, the desired control law can be obtained as follows:

\[ u^* = -(L_{33})^{-1} \left( L_{31} \bar{u}[k - N, k - 1] + L_{32} \bar{y}[k - N, k - 1] \right) \]  

\[ (54) \]

From \( L_{31}, L_{32}, L_{33}, \) and Equation (54):

\[ u^* = -(R + H^T PH)^{-1} \left( H^T PG \bar{u}[k - N, k - 1] + H^T PG \bar{y}[k - N, k - 1] \right) = -(R + H^T PH)^{-1} H^T PG \left( M_u \bar{u}[k - N, k - 1] + M_y \bar{y}[k - N, k - 1] \right) \]  

\[ (55) \]

From Equation (51):

\[ u^* = -(R + H^T PH)^{-1} H^T PG \bar{x}[k] \]  

\[ (56) \]

It can be seen that the obtained optimal output feedback controller described in this paper is equivalent to the optimal state feedback controller, without requiring all of the internal state variables. This overcomes the limitation of acquiring all internal state variables of the system in practical engineering applications of state feedback control.

4.3. Acquisition of the VCTs

The characteristic of the SDCS is that the controller is decentralized, so it is essential to obtain execution of the control tasks on each decentralized node, i.e., VCTs.

From Equation (54), the control task of the optimal output feedback LQR includes three parts: the construction of \( \bar{u} \), the construction of \( \bar{y} \), and the weighted sum of \( \bar{u} \) and \( \bar{y} \). It is easy to obtain two of the VCTs:

VCT\(_1\): from the sampled data, according to \( \bar{u} \) in Equation (49), to construct \( u \)-based \( \bar{u} \);

VCT\(_2\): from the sampled data, according to \( \bar{y} \) in Equation (50), to construct \( y \)-based \( \bar{y} \);

For the latter task, which is the weighted summing of \( \bar{u} \) and \( \bar{y} \), the number of virtualized VCTs depends on the dimensions of \( \bar{u} \) and \( \bar{y} \), as well as the microprocessor performance. Let VCT\(_3\) denote the VCT set of the virtualization of the weighted summation task. Some simple examples are as follows.
Example 1: when the dimensions of $\tilde{u}$ and $\tilde{y}$ are low, they are virtualized into a single VCT

$VCT_3$: The output feedback $-(L_{33})^{-1}(L_{31}\tilde{u} + L_{32}\tilde{y})$ is obtained;

Example 2: when the dimensions of $\tilde{u}$ and $\tilde{y}$ are high, they are virtualized into three VCTs

$VCT_{31}$: The output feedback $-(L_{33})^{-1}L_{31}\tilde{u}$ is obtained;

$VCT_{32}$: The output feedback $-(L_{33})^{-1}L_{32}\tilde{y}$ is obtained;

$VCT_{33}$: The total feedback $VCT_{31} + VCT_{32}$ is obtained.

In addition, virtualization can also be carried out by dividing $-(L_{33})^{-1}L_{31}$ and $-(L_{33})^{-1}L_{32}$ by rows; a detailed description is not provided here.

According to the description given above of the virtualization scheme for the LQR output feedback control task, for $VCT_1$ and $VCT_2$ we remove the earliest sampling data of $u$ and $y$ in each sampling period, obtain the latest $u$ and $y$, then construct $\tilde{u}$ and $\tilde{y}$ according to Equations (54) and (55). For $VCT_3$, the essence of the calculation is the weighted sum. The calculated amounts of the VCTs above are very small, and most low-performance microprocessors can be used for real-time processing.

5. Parameter Tuning Based on the Primal-Dual Method

We derived the VCTs of SDCS. In the method used in the current study, the parameter tuning of VCTs requires solving the LQR problem. Usually, dynamic programming (DP) is applied to solve the Bellman equation. In recent years, with the development of convex analysis and semi-definite programming (SDP) [31], SDP-based LQR-solving methods have emerged. In addition, research on LQR problems based on reinforcement learning (RL) has received widespread attention. Most of these solve the Bellman equation based on the DP of sampled data, meanwhile making use of the monotonicity of functions or the properties of contraction mappings to ensure convergence [32].

The solution of the primal problem can be achieved by solving the dual problem, making the problem easier to solve. Usually, the primal constrained problem is transformed into a dual unconstrained problem through the Lagrange function. When the described problem meets the Karush–Kuhn–Tucker (KKT) conditions, the above scheme can be adopted. The duality of RL for LQR is discussed and studied in [33].

In this section, we assume that the system $(G, H)$ is stable. The LQR problem described in Equation (32) can be expressed in the following form:

$$J = \sum_{i=0}^{+\infty} x[k]^T \left[ \begin{array}{cc} Q & 0 \\ 0 & R \end{array} \right] u[k] = \sum_{i=0}^{+\infty} x[k]^T L_{QR} u[k]$$

(57)

Let

$$L_F = \begin{bmatrix} G & H \\ FG & FH \end{bmatrix}, \Lambda = \begin{bmatrix} I_n & 0 \\ F & I_m \end{bmatrix}$$

Then,

$$\Lambda^{-1} = \begin{bmatrix} I_n & 0 \\ -F & I_m \end{bmatrix}$$

So, $\Lambda^{-1}L_F\Lambda = G + HF$, that is, $L_F \sim G + HF$.

Let

$$\begin{bmatrix} Q + G^T PG & G^T PH \\ H^T PG & R + H^T PH \end{bmatrix} = \tilde{P}$$

(in Equation (44)). $\tilde{P}$ is represented as

$$\tilde{P} = \begin{bmatrix} \tilde{P}_{11} & \tilde{P}_{12} \\ \tilde{P}_{12}^T & \tilde{P}_{22} \end{bmatrix}$$

(58)
where \( \tilde{P}_{11} = Q + G^T P G, \tilde{P}_{12} = G^T P H, \tilde{P}_{22} = R + H^T P H \).

The primal problem and dual problem introduced in [33] are:

Primal problem

\[
I_p = \min \text{Tr}(L_Q R S) \\
\text{s.t.} L_F S L_F^T + \Gamma = S
\]

where \( S \) and \( \Gamma \) are the symmetric positive definite matrix.

Dual problem

The Lagrange function is introduced:

\[
L(\hat{P}, \hat{F}, S) = \text{Tr}(L_Q R S) + \text{Tr}\left( \left( L_F S L_F^T + \Gamma - S \right) \hat{F} \right) + \text{Tr}(-S \hat{P})
\]

where \( \hat{P} \) is an arbitrarily fixed symmetric matrix, \( \hat{F} \) is an arbitrarily fixed symmetric positive semidefinite matrix cone. Then, the dual problem of the primal problem is:

\[
I_d = \sup_{\hat{P}, \hat{F}, S} L(\hat{P}, \hat{F}, S)
\]

Making use of theorem 1, attribute 7, and lemma 5 of [33], it is calculated that:

1. \( I_p = I_d \)
2. If \( (\hat{S}, \hat{F}) \) is the optimal point of the primal problem, \( (\hat{P}^*, \hat{P}^*) \) is the optimal point of the dual problem of the primal problem, then \( (\hat{S}, \hat{F}, \hat{P}^*) \) satisfies the KKT condition of \( (S, F, \hat{P}) \):

\[
L_F S L_F^T + \Gamma - S = 0
\]

\[
S > 0,
\]

\[
L_F^T \hat{P} L_F - \hat{F} + L_Q R = 0,
\]

\[
2 \left( \hat{P}_{12}^T + \hat{P}_{22} F \right) \begin{bmatrix} G & H \end{bmatrix} S \begin{bmatrix} G & H \end{bmatrix} = 0,
\]

3. Define \( Y(\hat{P}) = L_F^T \hat{P} L_F + L_Q R \), then, if \( \hat{P}_1 > \hat{P}_2, Y(\hat{P}_1) > Y(\hat{P}_2) \), and \( Y \) is a symmetric positive semi-definite cone. A matrix norm \( \| \cdot \| \) makes \( Y \) a contractive mapping. There is a unique symmetric matrix \( \tilde{P} \) that makes \( Y(\tilde{P}) = \tilde{P} \), that is, \( Y \) has a unique fixed point \( \tilde{P} \).
4. The \( \hat{P}_t \) in the following algorithm converges to the optimal solution \( \hat{P}^* \) of the dual problem, that is, the optimal solution \( \hat{P}^* \) of the matrix \( \hat{P} \) required in Equation (58).

Algorithm steps:

1. Initialization of \( F_0, \epsilon > 0, \) and \( t = 0. \)
2. Calculate \( L_{F_0} = \begin{bmatrix} G & H \\ F_0 G & F_0 H \end{bmatrix} \).
3. Solve \( \hat{P}_0 \) from \( L_{F_0}^T \hat{P}_0 L_{F_0} + L_Q R = \hat{P}_0. \)
4. Repeat.
5. \( t = t + 1. \)
6. Dual update: \( \hat{P}_{t+1} = L_{F_t}^T \hat{P}_t L_{F_t} + L_Q R. \)
7. Primal update: \( F_{t+1} = -(\hat{P}_{22})^{-1} (\hat{P}_{12})_{t+1}^T \).
8. \( L_{F_{t+1}} = \begin{bmatrix} G & H \\ F_{t+1} G & F_{t+1} H \end{bmatrix}. \)
9. Until \( \| F_{t+1} - F_t \| \leq \epsilon. \)
The optimal solution \( \tilde{P}^* \) of \( \tilde{P} \) in Equation (58) can be obtained from the derived \( \tilde{P} \). By combining Equations (51)–(53) and (58), the parameters of each VCT of SDCS can be obtained.

6. Simulation Analyses

From Section 5, we can conclude that the parameter tuning algorithm is simple to calculate, and solving the algorithm process does not involve the complex ARE or Lyapunov equation. It can be calculated with high-reliability low-performance microprocessors.

The VCTs obtained in this paper do not involve complex calculations, making them suitable for meeting the requirements of the SDCS. Additionally, the iterative process of the VCTs’ parameter tuning algorithm also avoids complex calculations. The purposes of the simulation described here are: 1. to verify the convergence rate of the parameter tuning algorithm; and 2. to verify whether the output feedback control constructed in this study achieves equivalent performance to the conventional LQR state feedback control.

Based on the aero-engine model established in Section 2, the analysis in this section includes the following two simulations:

SIMULATION 1.

This simulation focuses on the single-variable aero-engine control. We used the high-pressure rotor speed-control model at the steady point of the relative speed of 83% obtained in Section 2. The eigenvalues of the system matrix of the model are 0.9231 and 0.6576, so it is asymptotically stable. Let \( \bar{Q} = 1 \), \( R = 0.49857 \), then \( Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0.49875 \end{bmatrix} \). The initial feedback gain matrix is \( F = \begin{bmatrix} 0 & 0 \end{bmatrix} \), and it is calculated that \( L_{QR} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0.49875 \end{bmatrix} \).

According to the scheme for solving ARE in Equation (41), the optimal feedback gain matrix is \( F^* = \begin{bmatrix} -0.9120 \\ -0.0252 \end{bmatrix} \). The convergence details during the iteration are reported in Table 1.

<table>
<thead>
<tr>
<th>( \epsilon )</th>
<th>Steps</th>
<th>( F )</th>
<th>( |F_t - F_{t+1}|)</th>
<th>( |F_t - F^*|)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>3</td>
<td>\begin{bmatrix} -0.8712 &amp; -0.0261 \end{bmatrix}</td>
<td>0.0659</td>
<td>0.0274</td>
</tr>
<tr>
<td>0.01</td>
<td>4</td>
<td>\begin{bmatrix} -0.9266 &amp; -0.0256 \end{bmatrix}</td>
<td>6.0304 \times 10^{-4}</td>
<td>0.0140</td>
</tr>
<tr>
<td>0.001</td>
<td>4</td>
<td>\begin{bmatrix} -0.9266 &amp; -0.0256 \end{bmatrix}</td>
<td>6.0304 \times 10^{-4}</td>
<td>0.0140</td>
</tr>
<tr>
<td>0.0007</td>
<td>4</td>
<td>\begin{bmatrix} -0.9266 &amp; -0.0256 \end{bmatrix}</td>
<td>6.0304 \times 10^{-4}</td>
<td>0.0140</td>
</tr>
<tr>
<td>0.0006</td>
<td>8</td>
<td>\begin{bmatrix} -0.9120 &amp; -0.0252 \end{bmatrix}</td>
<td>3.5857 \times 10^{-5}</td>
<td>4.6999 \times 10^{-5}</td>
</tr>
</tbody>
</table>

The calculation results of \( \check{P}, L_{31}, L_{32}, L_{33} \) are shown in Equation (64), and the parameters of the three VCTs are obtained. Figure 6 shows convergence process of the parameters during the iterative process:

\[
\check{P} = \begin{bmatrix} 3.2878 & 0.0530 & 0.9692 \\ 0.0530 & 0.0014 & 0.0267 \\ 0.9692 & 0.0267 & 1.0460 \end{bmatrix}
\]

\[
L_{31} = \begin{bmatrix} 0.3772 \\ 0.6980 \end{bmatrix}
\]

\[
L_{32} = \begin{bmatrix} -1.2037 \\ 2.0982 \end{bmatrix}
\]

\[
L_{33} = 1.4060
\]

From Table 1 and Figure 6, the feedback gain matrix \( F \) converges rapidly, and \( F_t \) converges to \( F^* \) within eight iterations. The simulation results show that the LQR parameter tuning algorithm proposed in this paper has the characteristics of fewer convergence steps and fast convergence speed. Due to its advantages of simple calculation and rapid convergence, microprocessors with low price, high reliability, and low performance can be utilized for real-time parameter tuning.
The following simulation was conducted to verify the performance of the decentralized LQR output feedback control proposed in this paper. Generally, the performance of optimal state feedback is superior to that of output feedback. Through the design described in Section 4, the LQR output feedback control proposed in this paper can theoretically achieve equivalent performance to the LQR state feedback. This scheme overcomes the shortcomings of state feedback while achieving comparable performance. This simulation compares the LQR output feedback control proposed in this paper with the conventional state feedback control, to validate the aforementioned conclusions.

Figure 7 shows the input and the response curves of the high-pressure rotor speed of the turbofan engine when the output feedback scheme proposed in this paper and the conventional state feedback scheme were adopted, respectively. It can be seen from Figure 7 that the high-pressure rotor speed response curves of the turbofan engine were almost identical when the two schemes were adopted. However, in reality, the two response curves did not completely overlap. As shown in Table 1, this is because there is a certain error between the feedback gain matrices calculated by the primal-dual principle (the output feedback scheme in this paper) and the ARE (the conventional state feedback scheme), respectively. According to the simulation results presented in Figure 7, the output feedback scheme proposed in this paper can achieve equivalent performance to the conventional state feedback, which is consistent with the theoretical analysis in this paper. However, because the state variables of the high-pressure rotor speed model have no physical meaning, the conventional LQR state feedback scheme cannot be used for system design in practical engineering applications. On the other hand, the scheme proposed in this paper can be applied in actual engineering applications, because of the output feedback.

SIMULATION 2.

This simulation focused on the multiple variables of aero-engine control. We used the dual-variable control model of the operating point at a PLA angle of 35°, obtained in Section 2.
The eigenvalues of the system matrix of the model are 0.9644 and 0.9063, so it is asymptotically stable. Let \( \hat{Q} = 1, R = [1 0; 0 1] \), then \( Q = [1 0 0; 0 1 0; 0 0 1] \). The initial feedback gain matrix is \( F = [0 0; 0 0] \), and it is calculated that \( L_{QR} = [1 0 0; 0 1 0; 0 0 1; 0 0 1] \). According to the scheme for solving ARE in Equation (41), the optimal feedback gain matrix is \( F^* = [-0.0038 0.0017; -7.2976 \times 10^{-4} - 0.0010] \). The convergence details obtained during the iteration are shown in Table 2.

**Table 2.** Convergence details of Simulation 2.

<table>
<thead>
<tr>
<th>( \epsilon )</th>
<th>Steps</th>
<th>( F )</th>
<th>( |F_t - F_{t+1}| )</th>
<th>( F_t - F^* )</th>
</tr>
</thead>
</table>
| 0.1 | 1 | \[ -0.0038 0.0017 \\
-7.2977 \times 10^{-4} -0.001 \] | 0.0041 | 0.0041 |
| 0.01 | 1 | \[ -0.0038 0.0017 \\
-7.2977 \times 10^{-4} -0.001 \] | 0.0041 | 0.0042 |
| 0.001 | 2 | \[ -0.0038 0.0017 \\
-7.2976 \times 10^{-4} -0.001 \] | 4.8053 \times 10^{-9} | 6.9260 \times 10^{-8} |
| 0.0001 | 2 | \[ -0.0038 0.0017 \\
-7.2976 \times 10^{-4} -0.001 \] | 4.8053 \times 10^{-9} | 4.1779 \times 10^{-5} |
| 4.0 \times 10^{-9} | 5 | \[ -0.0038 0.0017 \\
-7.2976 \times 10^{-4} -0.001 \] | 3.8062 \times 10^{-9} | 5.6807 \times 10^{-8} |

The calculation result of \( \hat{P}, L_{31}, L_{32}, L_{33} \) is shown in Equation (65), and the parameters of the three VCTs were obtained. The convergence of the parameters during the iterative process is shown in Figure 8.

\[
\hat{P} = \begin{bmatrix}
13.5543 & -7.2729 & 0.0038 & 7.2976 \times 10^{-4} \\
-7.2729 & 11.3840 & -0.0017 & 0.0010 \\
0.0038 & -0.0017 & 1.0 & 3.4001 \times 10^{-7} \\
7.2976 \times 10^{-4} & 0.0010 & 3.4001 \times 10^{-7} & 1.0
\end{bmatrix}
\]

\[
L_{31} = 1.0 \times 10^{-5} \times \begin{bmatrix}
0.0545 & 0.0123 & 0.1123 & 0.0288 \\
0.0162 & 0.0166 & 0.0317 & 0.0334
\end{bmatrix}
\]

\[
L_{32} = \begin{bmatrix}
0.0018 & -0.0009 & 0.0018 & -0.0009 \\
0.0004 & 0.0004 & 0.0003 & 0.0004
\end{bmatrix}
\]

\[
L_{33} = \begin{bmatrix}
1.0 & 0.0 \\
0.0 & 1.0
\end{bmatrix}
\]
As shown in Table 2 and Figure 8, the feedback gain matrix $F$ converged rapidly, and $F_t$ converged to $F^*$ within five iterations. In fact, only one or two iterations were needed to obtain a feedback gain matrix $F$ with sufficient accuracy. The simulation results show that the LQR parameter tuning algorithm proposed in this paper has the characteristics of fewer convergence steps and fast convergence speed. Due to its advantages of simple calculation and rapid convergence, microprocessors with low price, high reliability, and low performance can be utilized for real-time parameter tuning.

![Convergence trajectory](image)

**Figure 8.** Convergence trajectory of $\|F_t - F_{t+1}\|$ and $\|F_t - F^*\|$ during the iterative process.

Figure 9 shows the input and the response curves of the high-pressure rotor speed of the turbofan engine when the output feedback scheme and the conventional state feedback scheme were adopted, respectively. Figure 10 shows the input and the response curves of the pressure ratio when the two feedback schemes were adopted, respectively. It can be seen from Figure 9 that the high-pressure rotor speed response curves of the turbofan engine were almost identical when the two schemes were adopted, and the same applies for the pressure ratio response curves in Figure 10. However, in both Figures 9 and 10, the system response of the two feedback schemes do not completely overlap. This is because there is a certain error between the feedback gain matrices calculated by the primal-dual principle and the ARE. According to the simulation results presented in Figures 9 and 10, the output feedback scheme proposed in this paper can achieve equivalent performance to conventional state feedback, which is consistent with the theoretical analysis in this paper. However, because the state variables of the dual-variable control model have no physical meaning, the conventional LQR state feedback scheme cannot be used for system design in practical engineering applications. On the other hand, the scheme proposed in this paper can be applied in actual engineering applications because of the output feedback.

According to the results of Simulations 1 and 2, the constructed output feedback scheme can achieve performance equivalent to that of the state feedback scheme. Furthermore, the proposed parameter tuning algorithm exhibits fast convergence speed and high accuracy. These simulation results show the effectiveness of the design scheme presented in this paper.

As shown in Table 2 and Figure 8, the feedback gain matrix $F$ converged rapidly, and $F_t$ converged to $F^*$ within five iterations. In fact, only one or two iterations were needed to obtain a feedback gain matrix $F$ with sufficient accuracy. The simulation results show that the LQR parameter tuning algorithm proposed in this paper has the characteristics of fewer convergence steps and fast convergence speed. Due to its advantages of simple calculation and rapid convergence, microprocessors with low price, high reliability, and low performance can be utilized for real-time parameter tuning.
A controller parameter tuning algorithm based on the primal-dual principle is constructed, and the VCTs of the SDCS are obtained. The constructed output feedback scheme is equivalent to the optimal state feedback control. A controller parameter tuning algorithm based on Q-Learning LQR is constructed, and the VCTs of the SDCS are obtained. The constructed output feedback scheme achieves control performance equivalent to that of the state feedback scheme. Furthermore, the proposed parameter tuning algorithm exhibits fast convergence speed and high convergence accuracy. The multi-variable model takes a maximum of eight iterations to obtain a high-precision feedback gain matrix, and the single-variable model takes only five iterations. In addition, the constructed output feedback scheme achieves control performance equivalent to the conventional state feedback, which is suitable for the actual engineering application of the aero-engine control system. The simulation results demonstrate the effectiveness of the design scheme presented in this paper.

**Figure 9.** High-pressure rotor speed response curve.

**Figure 10.** Pressure ratio response curve.

**7. Conclusions**

We propose a decentralized design scheme for controllers of aero-engines, the SDCS. In SDCS, a group of low-price, high-reliability, but low-performance microprocessors act as the controller of the aero-engine control system. An output feedback scheme based on Q-Learning LQR is constructed, and the VCTs of the SDCS are obtained. The constructed output feedback scheme is equivalent to the optimal state feedback control. A controller parameter tuning algorithm based on the primal-dual principle is introduced, which features simple calculation and can be implemented using low-cost, high-reliability, low-performance microprocessors for real-time calculation, suitable for the hardware characteristics of SDCS. Finally, we conducted simulation for verification. According to the simulation results, the controller parameter tuning algorithm proposed in this paper converges quickly and has high convergence accuracy. The single-variable model takes only a maximum of eight iterations to obtain a high-precision feedback gain matrix, and the multi-variable model takes a maximum of only five iterations. In addition, the constructed output feedback scheme achieves control performance equivalent to the conventional state feedback, which is suitable for the actual engineering application of the aero-engine control system. The simulation results demonstrate the effectiveness of the design scheme presented in this paper.
Author Contributions: Conceptualization, X.J. and J.L.; data curation, J.R.; investigation, X.J. and J.R.; methodology, X.J., J.L. and Y.W.; software, X.J. and J.R.; supervision, J.L. and Y.W.; validation, X.J. and J.R.; writing—original draft, X.J.; writing—review & editing, X.J., J.L., J.R. and Y.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data used to support the findings of this study are included within the article.

Conflicts of Interest: The authors declare that they have no competing financial and non-financial interest.

References


27. Pavlenko, D.; Dvirnyk, Y.; Przywoswa, R. Advanced Materials and Technologies for Compressor Blades of Small Turbofan Engine. *Aerospace* 2021, 8, 1. [CrossRef]


**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.