Reduced Order H_∞/LTR Method for Aeroengine Control System

YANG Gang, SUN Jian-guo

(College of Energy & Power engineering, Nanjing Univ. of Aeronautics & Astronautics, Nanjing 210016, China)

Abstract: This paper proposes a new loop recovery method to solve the reduced order problem of H_∞/LTR method. The resulted lower order controller shares almost the same performance and robustness as the original H_∞/LTR controller. Further more, this paper develops a new order reduction method: slow-fast mode order reduction (SFMOR) method. This order reduction method is particularly effective for those controllers whose modes can be divided into a slow part and a fast part according to their velocities. Application of these methods to a benchmark example and a certain turbofan engine is described.

Key words: aeroengine; H_∞/LTR; SFMOR; Schur; order reduction

航空发动机中的降阶 H_∞/LTR 设计方法. 杨 刚, 孙健国. 中国航空学报(英文版), 2004, 17(3): 129—135.

摘 要: 提出了一种新的, 可以解决航空发动机 H_∞/LTR 设计方法阶数过高问题的目标回路设计方法。这种方法设计出的低阶控制器保持了原 H_∞/LTR 方法设计出的控制器的性能和鲁棒性。此外, 还提出了一种新的降阶方法: 快、慢模态降阶法(SFMOR)。这种降阶方法对那些模态可以按其速度分为快、慢两部分的控制器特别适用。最后, 给出了设计实例。

关键词: 航空发动机; H_∞/LTR; SFMOR; Schur; 降阶

文章编号: 1000-9361(2004)03-0129-07 中图分类号: V233.7+1 文献标识码: A

Simple linear controllers are normally preferred to complex linear controllers. There are fewer things to go wrong in the hardware or bugs to fix in the software; they are easier to understand; and the computational requirements are less. But the robust multivariable methods often result in controllers with relatively high order. So the control engineers have to solve the practical problems of how to simplify these design procedures and how to effectively reduce the orders of the resulted controllers.

H_∞/LTR method[1] uses H_∞ optimization to recover the loop transfer function. It solves the problem of LQG/LTR controllers whose gain are usually unduly high and whose bandwidth unduly wide[2, 3]. With H_∞/LTR method, the resulted controller not only recovers LQR loop transfer function, but also has relatively low gain and narrow bandwidth. So the H_∞/LTR controller has superior noise rejection properties. However, like many other robust multivariable design methods, H_∞/LTR methods can only result in a high order controller. The order of the controller must be reduced before any practical implementation.

A new loop transfer recovery method is developed in this paper to solve the reduced order problem of H_∞/LTR method. The resulted controller has relatively low order, but it maintains the performance and robustness of the original system. Further more, a slow-fast mode order reduction (SFMOR) method is developed in this paper to further reduce the order of the above controller. SFMOR first divides the H_∞/LTR controller into two subsystems according to the velocities of the controller modes, then uses Schur[4, 5] method to reduce the orders of the two subsystems separately, and finally recombines the two reduced-order subsystems to form the ultimate reduced-order controller. Examples show that SFMOR achieves the expected goal.

Received date: 2003-07-25; Revision received date: 2004-06-07

Foundation item: Aeronautic Science Foundation of China(00C52030); National Doctoral Education Foundation(2000028701)

1 H_∞/LTR Method

Consider the unit feedback system as shown in Fig. 1. $\boldsymbol{G}(s)$ is the design plant model (DPM),

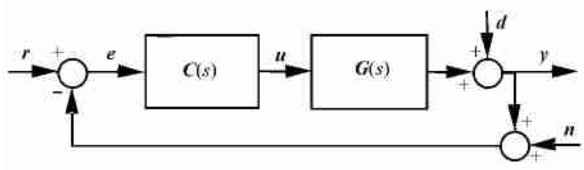

Fig. 1 Unit feedback system

and can be described by state space form as

$$\boldsymbol{G}(s):\begin{cases}\dot{\boldsymbol{x}}=\boldsymbol{A}\boldsymbol{x}+\boldsymbol{B}\boldsymbol{u}\\ \boldsymbol{y}=\boldsymbol{C}\boldsymbol{x}+\boldsymbol{D}\boldsymbol{u}\end{cases}\tag{1}$$

where $\boldsymbol{x}$, $\boldsymbol{u}$ and $\boldsymbol{y}$ belong to $\mathbf{R}^n$, $\mathbf{R}^m$ and $\mathbf{R}^p$, respectively; n, m and p represent system, input and output dimensions, respectively. Further more, it should be assumed that $(\boldsymbol{A}, \boldsymbol{B})$ is stabilizable and $(\boldsymbol{A}, \boldsymbol{C})$ detectable. Then the transfer function matrix of the DPM is

$$\boldsymbol{G}(s)=:\left[\begin{array}{c|c}\boldsymbol{A}&\boldsymbol{B}\\ \hline \boldsymbol{C}&\boldsymbol{D}\end{array}\right]=\boldsymbol{C}(s\boldsymbol{I}_n-\boldsymbol{A})^{-1}\boldsymbol{B}+\boldsymbol{D}\tag{2}$$

where s is Laplace operator and $\boldsymbol{I}_n$ is the unit matrix of dimension n.

Consider the case when the loop is broken at the input of the plant of Fig. 1 (the case when the breakpoint is at the output of the plant is similar). Assume that the sensitivity function of the closed loop is $\boldsymbol{S}_i$, and the LQR target loop sensitivity function is $\boldsymbol{S}_I$, then the target loop transfer function can be recovered by minimizing the following performance index (PI) with H_∞ optimization

$$\min_{Q\in\mathbf{R}_{H_\infty}}\|\boldsymbol{J}\|_\infty:=\min_{Q\in\mathbf{R}_{H_\infty}}\left\|(\boldsymbol{S}_\mathrm{i}-\boldsymbol{S}_\mathrm{I})\right\|_\infty\tag{3}$$

But it is pointed out by Han and Hsia[3] that this method can not reject measurement noises effectively. In order to maintain system performance as well as to reject measurement noises effectively, Ref. [1] proposes the following PI

$$\min_{Q\in\mathbf{R}_{H_\infty}}\|\boldsymbol{J}\|_\infty=\min_{Q\in\mathbf{R}_{H_\infty}}\left\|\begin{matrix}(\boldsymbol{S}_\mathrm{i}-\boldsymbol{S}_\mathrm{I})\boldsymbol{W}_S\\ \boldsymbol{U}\boldsymbol{W}_U\end{matrix}\right\|_\infty\tag{4}$$

where $\boldsymbol{U}=(\boldsymbol{I}+\boldsymbol{G}\boldsymbol{C})^{-1}\boldsymbol{C}$, representing the transfer function from $\boldsymbol{r}$ to $\boldsymbol{u}$ in Fig. 1, and $\boldsymbol{W}_S$ and $\boldsymbol{W}_U$ are weighting matrices. Then Eq. (4) can achieve the following simultaneously: (1) $\varepsilon_S(=(\boldsymbol{S}_\mathrm{i}-\boldsymbol{S}_\mathrm{I})\boldsymbol{W}_S)$ is small enough, to make sure that the controller can recover the performance of the target loop; and (2), $\varepsilon_U(=\boldsymbol{U}\boldsymbol{W}_U)$ is small enough, to make sure that the controller has satisfactory bandwidth.

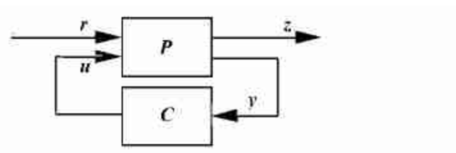

Fig. 2 Standard H_∞ problem

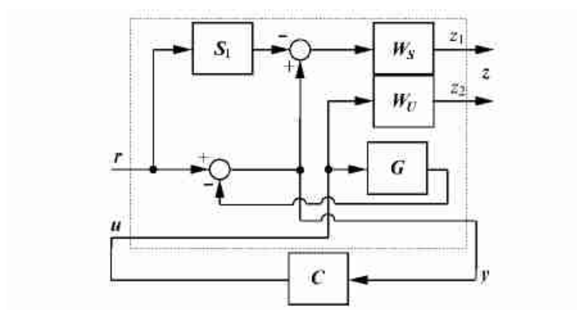

Fig. 3 Block diagram representation of performance index $\boldsymbol{J}$

In the sequel, discuss how to solve Eq. (4). Fig. 2 is the diagram of standard H_∞ problem. Controller $\boldsymbol{C}(s)$ in Fig. 2 can be designed by standard H_∞ procedure to satisfy $\|\boldsymbol{T}_{zr}\|_\infty\leqslant\lambda$, where $\boldsymbol{T}_{zr}$ is the transfer function from $\boldsymbol{r}$ to $\boldsymbol{z}$, and $\lambda>0$ is constant. In order to translate Eq. (4) into the formula of standard H_∞ problem, let's introduce $\boldsymbol{S}_\mathrm{I}$, $\boldsymbol{W}_S$ and $\boldsymbol{W}_U$ into Fig. 1 and redraw it in Fig. 3. Then those in the dashed block correspond to $\boldsymbol{P}$ in Fig. 2, and the diagram satisfies that $\boldsymbol{J}=\boldsymbol{T}_{zr}$. As a result, Eq. (4) can now be solved as a standard H_∞ problem according to Fig. 3.

Ref. [1] also gives the guidelines to how to choose the weighting matrices $\boldsymbol{W}_S$ and $\boldsymbol{W}_U$. With these guidelines, an integrator is incorporated in the control system to eliminate steady state error, but the DPM is not augmented. So the controller order has been reduced.

2 Reduced Order Target Loop Design

Analyzing the nominal controlled plant $\boldsymbol{P}$ in Fig. 3, it can be seen that $\boldsymbol{S}_\mathrm{I}$ is an important part of $\boldsymbol{P}$. Therefore, if a low order $\boldsymbol{S}_\mathrm{I}$ can be arrived, the order of the nominal controlled plant $\boldsymbol{P}$ will be reduced accordingly, and so does the controller.

It is well known in the LTR procedure that the requirements for a certain loop to be a target loop are that this loop should have good performance and robustness. Examining the design procedure of H_∞/LTR method in section 1, it can found that the target sensitivity function is designed using LQR method, but it can also be found that none of other properties of LQR loop is necessary for the design. It is used because and only because it is a loop that "has good performance and robustness". Therefore, if there exists a loop that is simple and has the good performance and robustness of LQR loop, it can be doubtlessly used as our target loop.

Therefore, this paper considers reducing the order of LQR target loop first. Based on Edmunds model-matching design method[6, 7], the LQR loop can be replaced with a lower order loop. Examples in section 4 will show that when the target LQR loop is replaced with a simplified one, the resulted controller will have relatively lower order, but it will have almost the same good performance and robustness as the controller whose target loop is not simplified.

3 Controller Order Reduction

Although the orders can be reduced by carefully choosing weighting matrices and simplifying the target loop, the orders of H_∞/LTR controller are still very high. The controller still needs further order-reduction before implementation.

In this paper, a simple yet effective method, called slow-fast mode order reduction (SFMOR) method, is developed. Assume that the controller $\boldsymbol{C}(s)$ has been derived. Then the modes of the controller can be easily calculated. If these modes have widely separated velocities, especially when including integrators, normal order reduction methods may result in great errors in the modes recombination procedure. So, in this paper, SFMOR is used to reduce the controller order. This method divides the order reduction procedure into 3 steps:

1) Divide the controller into two subsystems according to the mode velocities: $\boldsymbol{C}(s) = \boldsymbol{C}_1(s) + \boldsymbol{C}_2(s)$, where $\boldsymbol{C}_1(s)$ and $\boldsymbol{C}_2(s)$ contain the slow and fast modes of $\boldsymbol{C}(s)$, respectively.

2) Reduce the orders of subsystems $\boldsymbol{C}_1(s)$ and $\boldsymbol{C}_2(s)$ with Schur method, denoting the resulted reduced order subsystems as $\boldsymbol{C}_{1r}(s)$ and $\boldsymbol{C}_{2r}(s)$, respectively. Because subsystem $\boldsymbol{C}_1(s)$ ($\boldsymbol{C}_2(s)$, respectively) contains only the slow (fast, respectively) modes of the original controller $\boldsymbol{C}(s)$, Schur method works effectively in the mode recombination procedure.

3) Recombine $\boldsymbol{C}_{1r}(s)$ and $\boldsymbol{C}_{2r}(s)$ to get the final reduced order controller $\boldsymbol{C}_r(s)$

$$\boldsymbol{C}_r(s) = \boldsymbol{C}_{1r}(s) + \boldsymbol{C}_{2r}(s)$$

From the above procedure, it is clear that SFMOR is particularly effective if the modes of the controller can be divided into two parts according to their velocities. On the other hand, H_∞/LTR controller contains integrators to eliminate steady state errors. The velocities of the modes corresponding to the integrators would definitely be much slower than other modes. According to this correspondence, the modes of H_∞/LTR controller are obviously divided into two parts. As a result, SFMOR should be effective for H_∞/LTR controller, and the examples in section 4 will show this.

4 Examples

In this section, two examples will show the procedure and the effectiveness of the methods discussed in section 2 and section 3.

Example 1 First let's consider the benchmark example as in Ref. [3].

Consider the DPM in Fig. 1. $\boldsymbol{G}(s)$ has the following state space representation

$$\begin{cases} \dot{\boldsymbol{x}} = \begin{bmatrix} 0 & 1 \\ -3 & -4 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \boldsymbol{e} \\ \boldsymbol{y} = \begin{bmatrix} 2 & 1 \end{bmatrix} \boldsymbol{x} \end{cases}$$

First, two controllers are designed. With LQR loop as the target loop, H_∞/LTR method yields controller ① $\boldsymbol{C}_{①}(s)$; when using the simplified target loop, this method turns out controller

② $\boldsymbol{C}_{②}(s)$

$$\boldsymbol{C}_{①}(s) = (1195s^6 + 1.349\times 10^5 s^5 + 1.615\times 10^6 s^4 + 7.476\times 10^6 s^3 + 1.592\times 10^7 s^2 + 1.521\times 10^7 s + 5.283\times 10^6)/(s^7 + 6887s^6 + 1.218\times 10^5 s^5 + 8.161\times 10^5 s^4 + 2.47\times 10^6 s^3 + 3.312\times 10^6 s^2 + 1.543\times 10^6 s)$$

$$\boldsymbol{C}_{②}(s) = (3353s^4 + 3.655\times 10^5 s^3 + 3.101\times 10^6 s^2 + 7.787\times 10^6 s + 5.048\times 10^6)/(s^5 + 1.855 + 10^4 s^4 + 2.58\times 10^5 s^3 + 1.155\times 10^6 s^2 + 1.426\times 10^6 s)$$

Next, controller ① and ② are connected to $\boldsymbol{G}(s)$ according to Fig. 1 to do some simulations, respectively. The results are plotted in Figs. 4 and 5. In one of the simulations for each controller, the unit step input is contaminated by a zero mean white noise of variance $\sigma_n^2 = 0.1$.

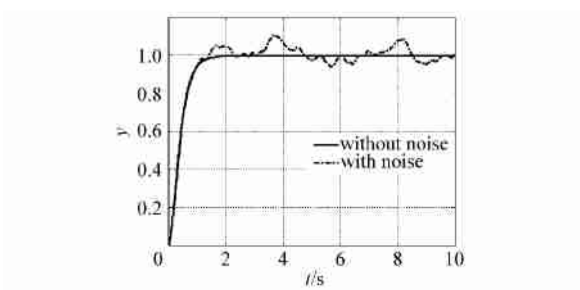

Fig. 4 Unit step responses of controller ①

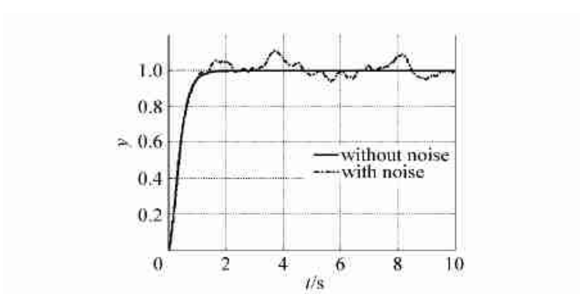

Fig. 5 Unit step responses of controller ②

As shown in Figs. 4 and 5, the controller with reduced order target loop works almost as well as controller ① does. It also rejects noise efficiently. The variance of the noise after controller ② becomes 0.0013443 (0.0012977 for controller ①). It's much smaller than 0.1, the variance of the original white noise. Therefore, a conclusion can be drawn: H_∞/LTR with reduced order target loop would not degrade controller performance. But by comparing the orders of controller ① and ②, it is clear that controller order is reduced.

The order of controller ② is reduced comparing to controller ①, but it is still too high for the controller to be practically implemented. Therefore, the order of controller ② is further reduced by SFMOR.

Analyzing controller ②, its modes can be easily got: 0, -1.9989, $-5.9583 + 1.728i$, $-5.9583 - 1.728i$, -18536. Clearly, the first mode is much slower than the others. Controller ② is therefore easily divided into two subsystems as following

$$\boldsymbol{C}_{②}(s) = \boldsymbol{C}_{②,1}(s) + \boldsymbol{C}_{②,2}(s)$$

where $\boldsymbol{C}_{②,1}(s) = \dfrac{3.54}{s}$

$$\boldsymbol{C}_{②,2}(s) = (3349s^3 + 2.998\times 10^5 s^2 + 2.188\times 10^6 s + 3.698\times 10^6)/(s^4 + 1.855\times 10^4 s^3 + 2.58\times 10^5 s^2 + 1.155\times 10^6 s + 1.426\times 10^6)$$

Since the first subsystem contains only one state, its order doesn't need reducing anymore. But the second subsystem has 4 states, and with Schur method the reduced system can be got as following

$$\boldsymbol{C}_{②,2r}(s) = \frac{3291s + 2.638\times 10^5}{s^2 + 1.817\times 10^5 s + 1.076\times 10^5}$$

which contains the following two modes: -5.922, -18167. It's very clear now how Schur method recombines the modes of $\boldsymbol{C}_{②,2}(s)$. Finally, the reduced order controller ③ $\boldsymbol{C}_{③}(s)$ results by recombining the two reduced order subsystems

$$\boldsymbol{C}_{③}(s) = \frac{1152s^2 + 1.144\times 10^5 s + 1.323\times 10^5}{s^3 + 6585s^2 + 3.865\times 10^4 s}$$

which contains modes: 0, -5.922, -18167.

For comparison, $\boldsymbol{C}_{②}(s)$ is directly reduced with Schur method. If $\boldsymbol{C}_{②}(s)$ is reduced to a 3-order system, controller ④ $\boldsymbol{C}_{④}(s)$ will be got

$$\boldsymbol{C}_{④}(s) = (3349s^2 + 3.067\times 10^5 s + 2.823\times 10^6)/(s^3 + 1.855\times 10^4 s^2 + 3.287\times 10^5 s + 1.909\times 10^6)$$

with modes: $-8.624 + 4.943i$, $-8.624 - 4.943i$, -18536. Clearly, in the modes recombination procedure, the integrator disappears.

Fig. 6 shows singular values of the errors between $\boldsymbol{C}_{②}(s)$ and the two reduced order controller. The solid line stands for the error between $\boldsymbol{C}_{②}(s)$ and $\boldsymbol{C}_{③}(s)$: $\boldsymbol{E}_{③}(s)=\boldsymbol{C}_{②}(s)-\boldsymbol{C}_{③}(s)$, and the dash-dotted line stands for the error between $\boldsymbol{C}_{②}(s)$ and $\boldsymbol{C}_{④}(s)$: $\boldsymbol{E}_{④}(s)=\boldsymbol{C}_{②}(s)-\boldsymbol{C}_{④}(s)$. From Fig. 6, it is clear that error $\boldsymbol{E}_{③}(s)$ is much smaller than error $\boldsymbol{E}_{④}(s)$. This also testifies to the declaration: SFMOR is particularly fitted for H_∞/LTR controller, while directly applying Schur method would result in bigger error.

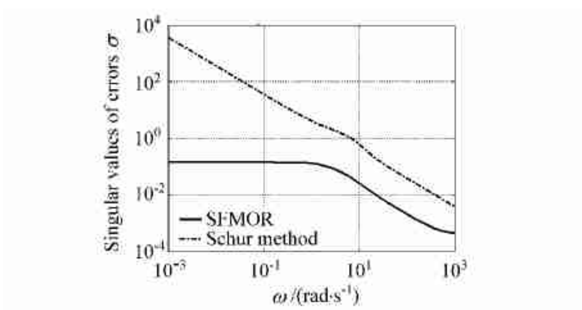

Fig. 6 Singular value plot of error function

Next the reduced order controllers are connected to $\boldsymbol{G}(s)$ to do similar simulations as Figs. 4 and 5. The simulation results of controller ③ and controller ④ are plotted in Figs. 7 and 8, respectively.

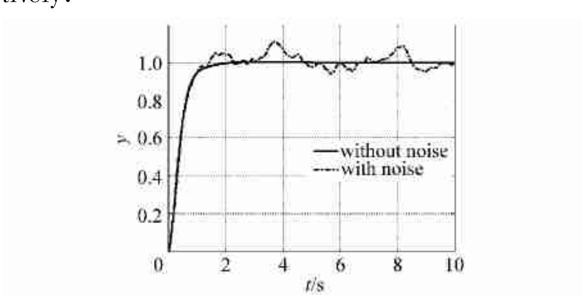

Fig. 7 Unit step responses of controller ③

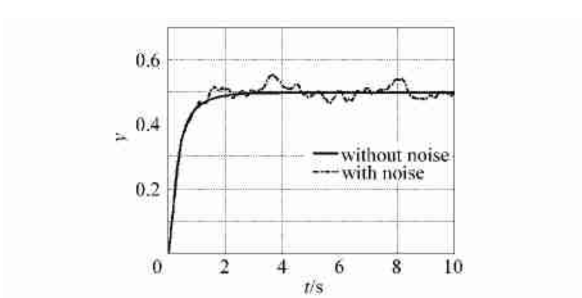

Fig. 8 Unit step responses of controller ④

Simulation results show that only controller ③ achieves almost the same performance as the un-reduced controller does. Controller ④ fails to eliminate steady state error. This is because Schur method has incorrectly eliminated the integrator from controller ②. The variance of the noise after controller ③ is 0.0013312. It is also close to that of the un-reduced controller. The following conclusions can now be drawn: SFMOR is particularly fitted for H_∞/LTR controller; the effect on the performance is almost negligible; SFMOR is more effective than Schur method for H_∞/LTR controller.

Example 2 Next, a lower order controller will be designed for a certain turbofan engine.

First derive a linear model from a non-linear model of the turbofan engine model, at $H=0$, $M=0$ and intermediate power level (DPM). Incorporating actuators in the linear model, its $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$ matrices can be got as

$$\boldsymbol{A}=\begin{bmatrix}-4.6855 & 1.7323 & 1.1791\\ -0.5565 & -2.7219 & 1.9258\\ 0 & 0 & -10\end{bmatrix},\ \boldsymbol{B}=\begin{bmatrix}0\\0\\10\end{bmatrix}$$

$$\boldsymbol{C}=[1\quad 0\quad 0],\ \boldsymbol{D}=0$$

Once the DPM is got, it is possible to design H_∞/LTR controllers with full order and its lower order compartments. Because the design procedure is almost the same as Example 1, only the results of two controllers will be given: (1) the full order controller $\boldsymbol{C}_{lqr}(s)$; (2) the controller $\boldsymbol{C}_{r}(s)$ whose order is reduced by SFMOR.

The first controller, *i.e.*, $\boldsymbol{C}_{lqr}(s)$, has the following representation

$$\boldsymbol{C}_{lqr}=(107.7s^8+4947s^7+9.447\times10^4s^6+9.75\times10^5s^5+5.93\times10^6s^4+2.173\times10^7s^3+4.668\times10^7s^2+5.331\times10^7s+2.43\times10^7)/(s^9+1186s^8+4.228\times10^4s^7+6.141\times10^5s^6+4.663\times10^6s^5+2\times10^7s^4+4.831\times10^7s^3+6.02\times10^7s^2+2.898\times10^7s)$$

When it is used to control the non-linear turbofan engine model, it gives results as shown in Fig. 9 and 10. The aeroengine model is contaminated by a zero mean white noise with variance $\sigma_n^2=641.3493$ $(\text{rev/min})^2$, and the data are obtained from the statistics of a real semi-physical simulation test stand of aeroengine control system. Fig. 9 is the result of a step response and Fig. 10 is the result of an accelerating process. The control schedule for the turbofan engine is to maintain the fan speed to be constant.

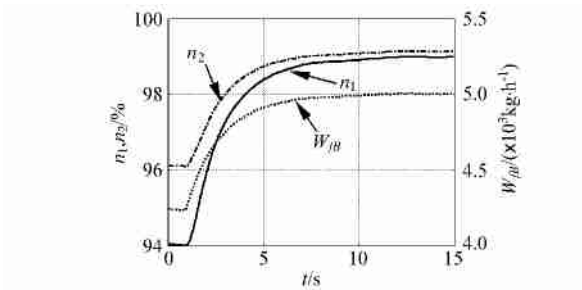

Fig. 9 Step response of aeroengine controlled by $C_{\mathrm{lqr}}(s)$

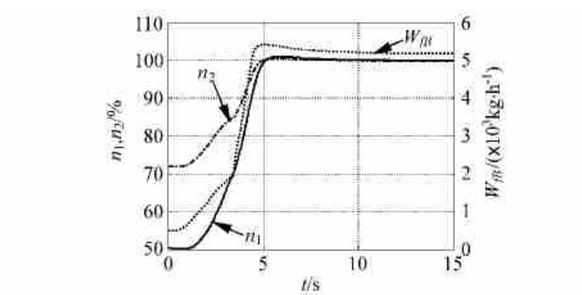

Fig. 10 Accelerating line of aeroengine controlled by $\boldsymbol{C}_{\mathrm{lqr}}(s)$

The second controller, *i.e.*, $\boldsymbol{C}_{\mathrm{r}}(s)$, has the following form

$$\boldsymbol{C}_{\mathrm{r}}(s)=\frac{326.8s^2+4359s+1.064\times10^4}{s^3+3102s^2+1.171\times10^4 s}$$

$\boldsymbol{C}_{\mathrm{r}}(s)$ is also connected to the aeroengine model to do similar simulations as $\boldsymbol{C}_{\mathrm{lqr}}(s)$. The results are plotted in Fig. 11 and 12.

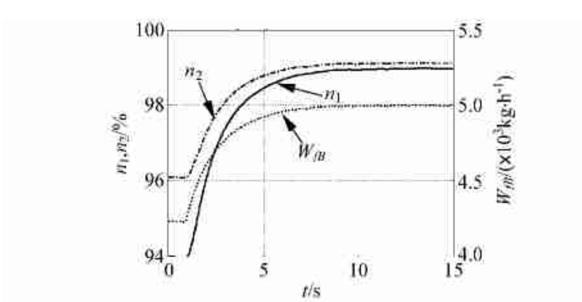

Fig. 11 Step response of aeroengine controlled by $\boldsymbol{C}_{\mathrm{r}}(s)$

After the controllers are designed and simulations are done, the two controllers can now be compared.

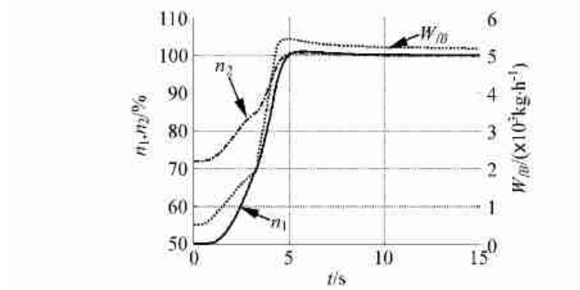

Fig. 12 Accelerating line of aeroengine controlled by $\boldsymbol{C}_{\mathrm{r}}(s)$

First of all, the order of $\boldsymbol{C}_{\mathrm{r}}(s)$ is 3 while that of $\boldsymbol{C}_{\mathrm{lqr}}(s)$ is 9. This means that the controller order has been reduced to an acceptable extend. Therefore, the order reduction procedure proposed in this paper is effective.

Next, from the simulation results, it is clear that the reduced order controller $\boldsymbol{C}_{\mathrm{r}}(s)$ performs almost as well as the un-reduced controller $\boldsymbol{C}_{\mathrm{lqr}}(s)$ does. $\boldsymbol{C}_{\mathrm{r}}(s)$ also rejects noise effectively. And the simulation results show that $\boldsymbol{C}_{\mathrm{r}}(s)$ is a robust controller as $\boldsymbol{C}_{\mathrm{lqr}}(s)$. This can be easily demonstrated by the following three reasons: ① the controller is designed using the linear DPM, but offers good performance on the nonlinear model; ② The accelerating simulations show that the controller offers good performance in a wide range of the power level. It should be pointed out that in the accelerating process, there are two kinds of controllers that work: one works when n_2 rotates slower than 85% of its design speed, the other works when n_2 rotates faster than 85% of its design speed. The controllers developed in this paper take in force when n_2 rotates faster. ③ Further simulations at other points in the full envelope (Simulations in this paper are at sea-level, static point in the envelope.) of the turbofan engine are also done. The simulation results show that controller $\boldsymbol{C}_{\mathrm{r}}(s)$ does offer good performance at those points. But because of the length limitation, these results are omitted.

Now the conclusions can be drawn: the order reduction procedure proposed in this paper can effectively simplify the H_∞/LTR controller, but it would not degrade controller performance and robustness.

5 Conclusions

(1) This paper develops a new H_∞/LTR method. It can result in a lower order controller, but does not degrade performance and robustness.

(2) This paper develops a new order reduction method: SFMOR. SFMOR is very effective for controllers whose modes can be divided into a slow part and a fast part according to their velocities. Particularly, in the examples mentioned above, SFMOR shows its superiority to Schur method.

(3) The second example in section 4 shows that the reduced-order controller designed in this paper has good performance and robustness for the turbofan engine.

References

[1] 杨刚, 孙健国. 一种新的控制系统 H_∞/ LTR 设计方法 [J]. 航空学报, 2004, 25(2): 104—107.
Yang G, Sun J G. A new H_∞/LTR method for control system design [J]. Chinese Journal of Aeronautics, 2004, 25(2): 104—107. (in Chinese).

[2] Chen B M, Saberi A, Sannuti P. A new stable compensator design for exact and approximate loop transfer recovery [J]. Automatica, 1991, 27(2): 257—280.

[3] Han K C, Hsia T C. On reducing compensator bandwidth of LQG/LTR control: an H_∞ optimization approach [A]. ACC [C]. 1990. 924—929.

[4] Safonov M G, Chiang R Y. A Schur method for balanced model reduction [J]. IEEE Trans on Automat Contr, 1989, AC-34(7): 729—733.

[5] Moore B C. Principal component analysis in linear systems: controllability, observability, and model reduction [J]. IEEE Trans on Automat Contr, 1981, AC-26(1): 17—31.

[6] Shrider A, Geroger J C, Xin Z, *et al*. H_∞ control design for a jet engine [R]. AIAA-98-3753, 1998.

[7] Edmunds J M. Control system design and analysis using closed-loop Nyquist and Bode arrays [J]. IJC, 1979, 30(5): 773—802.

Biographies:

YANG Gang Born in 1976, received his Ph. D. degree from Nanjing University of Aeronautics and Astronautics in 2004. He is specialized in aeroengine modeling and robust control. Presently he is doing post-doctorate researches in China Aviation Motor Control System Institute. E-mail: yg_203@yahoo.com

SUN Jian-guo (1939-) is a professor and director for doctors in College of Energy and Power Engineering, Nanjing University of Aeronautics and Astronautics during 1982-1984, he was doing research works in Columbia Univ. in America. He is specialized in modeling, control and diagnosis of aeroengine, robust control and integrated flight/propulsion Control. E-mail: jgspe@nuaa.edu.cn