Proof: Let S be any $n \times n$ real symmetric matrix. Write the identity

$$P - \frac{P + P^T}{2} = \frac{P - S}{2} - \frac{P^T - S}{2} = \frac{P - S}{2} - \frac{(P - S)^T}{2} \quad (2)$$

When applying the triangle inequality of the norm to this identity, and noting that $\|(P - S)\| = \|(P - S)^T\|$ (all through this Note, $\|\cdot\|$ symbolizes the Frobenius norm), one obtains

$$\left\| P - \frac{P + P^T}{2} \right\| \leq \left\| \frac{P - S}{2} \right\| + \left\| \frac{(P - S)^T}{2} \right\| = \|P - S\| \quad (3)$$

Then using Eq. (1) the latter result can be written as

$$\|P - P_s\| \leq \|P - S\| \quad (4)$$

Because S is any symmetric matrix, it follows that there is no symmetric matrix that is closer to P than P_s, and hence $P_s = P_c$. □

The preceding proof is an algebraic one. Next we introduce a new calculus-based proof. As before, let S be any real symmetric matrix. Define $\boldsymbol{J}$ as $\|P - S\|^2$; that is,

$$\boldsymbol{J} = \text{tr}\{(P - S)(P - S)^T\} = \text{tr}\{(P - S)(P^T - S)\} \quad (5)$$

We wish to find that S that is the closest to P. Let us express S as

$$S = P_c + \varepsilon H \quad (6)$$

where ε is a scalar. When this expression for S is substituted into Eq. (5), we obtain

$$\boldsymbol{J} = \text{tr}\big\{(P - P_c - \varepsilon H)(P^T - P_c - \varepsilon H)\big\} \quad (7)$$

A necessary condition for $\boldsymbol{J}$ to be minimum is

$$\left.\frac{d\boldsymbol{J}}{d\varepsilon}\right|_{\varepsilon=0} = 0 \quad (8)$$

where H is an $n \times n$ symmetric matrix [imposed by Eq. (6)] chosen such that this derivative exists but otherwise is arbitrary. Application of this condition to $\boldsymbol{J}$ of Eq. (7) yields

$$\left.\frac{d\boldsymbol{J}}{de}\right|_{\varepsilon=0} = \text{tr}\big\{-H\big(P^T - P_c\big) - \big(P - P_c\big)H\big\} = 0 \quad (9)$$

Using the properties $\text{tr}\{A + B\} = \text{tr}\{A\} + \text{tr}\{B\}$ and $\text{tr}\{AB\} = \text{tr}\{BA\}$, we obtain, from Eq. (9),

$$\text{tr}\big\{\big(P^T - P_c + P - P_c\big)H\big\} = 0 \quad (10)$$

Because this holds for any symmetric matrix H for which Eq. (8) holds, the expression in parentheses must be zero. This yields

$$P_c = (P + P^T)/2 = P_s \quad (11)$$

It is easy to see that this is a point of minimum. □

The result $P_c = (P + P^T)/2$ is not at all surprising. To see this, let $p_{c,i,j}$ be the i,j element of P_c, and similarly, let $p_{i,j}$ be the i,j element of P. Obviously, the closest to the i,i element of P in the symmetric matrix P_c is $p_{i,i}$ itself; that is, $p_{c,i,i} = p_{i,i}$, which is identical to writing

$$p_{c,i,i} = \frac{p_{i,i} + p_{i,i}}{2} \quad (12)$$

Next we want to determine $p_{c,i,j}$ when $i \neq j$. Note that, for $S = P_c$, $\boldsymbol{J}$ defined in Eq. (5) is equal to

$$\sum_{i=1}^{n}\sum_{j=1}^{n}(p_{i,j} - p_{c,i,j})^2$$

For each i and j, this sum includes the quantity $\boldsymbol{d}^2$ defined as $\boldsymbol{d}^2 = (p_{i,j} - p_{c,j,i})^2 + (p_{i,j} - p_{c,j,i})^2$, which is contributed by the elements of the difference matrix $\Delta = (P - P_c)$ that are symmetric about the diagonal of Δ. Because P_c is symmetric, we can write $p_{c,i,j} = p_{c,j,i} = x$. Because P_c is the symmetric matrix closest to P, x minimizes $\boldsymbol{d}^2$. It is easy to show that $\boldsymbol{d}^2$ is minimal when

$$x = \frac{p_{i,j} + p_{j,i}}{2} \quad (13)$$

Equations (12) and (13) express the fact that P_c is identical to P_s.

Conclusions

In this Note we pointed out that the symmetrized real matrix is also the symmetric matrix that is the closest, in the Euclidean norm, to the matrix being symmetrized. This implies that, when symmetrizing the solutions to Riccati and Lyapunov equations, one actually replaces the solution by its closest symmetric matrix. A proof of this fact that followed a proof given in the literature in 1955 was presented. A new, calculus-based, proof was also introduced. It was shown that this result can be obtained using simple rationale.

References

[1]Bryson, A. E., and Ho, Y. C., *Applied Optimal Control*, Blaisdell, Waltham, MA, 1969.

[2]Gelb, A. (ed.), *Applied Optimal Estimation*, MIT Press, Cambridge, MA, 1974.

[3]Friedland, B., *Control System Design*, McGraw–Hill, New York, 1986.

[4]Bellantoni, J. F., and Dodge, K. W., "A Square Root Formulation of the Kalman–Schmidt Filter," *AIAA Journal*, Vol. 5, No. 7, 1967, p. 1312.

[5]Kwakernaak, H., and Sivan, R., *Linear Optimal Control Systems*, Wiley, New York, 1972, p. 248.

[6]Brown, R. G., and Hwang, P. Y. C., *Introduction to Random Signals and Applied Kalman Filtering*, 2nd ed., Wiley, New York, 1992, p. 272.

[7]Fan, K., and Hoffman, A., "Some Metric Inequalities in the Space of Matrices," *Proceedings of the American Mathematical Society*, Vol. 6, Feb. 1955, pp. 111–116.

Translational Motion Control of Vertical Takeoff Aircraft Using Nonlinear Dynamic Inversion

Yoge Patel* and Phill R. Smith†

Defence Evaluation and Research Agency, Bedford MK41 6AE, England, United Kingdom

Nomenclature

$a_{x\,\text{dem}}, a_{y\,\text{dem}}, a_{z\,\text{dem}}$	= translational acceleration commands in Earth/wind x, y, and z directions, respectively
F_x, F_y, F_z	= body axis forces in x, y, and z directions, respectively
g	= gravity
m	= aircraft mass
p, q, r	= body axis roll, pitch, and yaw rate, respectively
u, v, w	= axial, lateral, and normal components of aircraft velocity relative to the air, respectively
V_{bw}	= acceleration to velocity transformation bandwidth

Received July 25, 1995; revision received July 10, 1997; accepted for publication Sept. 3, 1997.

*Senior Scientist, Flight Management and Control Department. Member AIAA.

†Principal Scientist, Flight Management and Control Department. Member AIAA.

$v_{x\,\mathrm{dem}}, v_{y\,\mathrm{dem}}, v_{z\,\mathrm{dem}}$ = translational velocity commands in Earth/wind axes x, y, and z directions, respectively
α, β = angles of attack and sideslip, respectively
θ, ϕ, ψ = pitch, bank, and heading angles, respectively (Euler angles)
θ_{xz}, θ_{xy} = thrust vector nozzle angles in $(x\text{-}z)$ and $(x\text{-}y)$ planes, respectively

Subscripts

bw = bandwidth
E/W = Earth/wind axis

Introduction

THE use of nonlinear dynamic inversion (NDI) for robust flight control law design has received considerable attention recently.[1–4] This work exploits the potential of NDI to provide ideal control, without recourse to conventional design methods, to determine what a given vehicle might be capable of in maneuver terms and to investigate response requirements. The terms ideal control and exact NDI are used to convey decoupled, high-bandwidth response without use of implicit input/output linearization. The approach thus concentrates on the use of NDI to provide a precise functional description of requirements and hence to minimize nugatory time at the control law design stage. This approach was previously explored by application to a fixed-wing, poststall aircraft.[5]

The exact NDI approach provides a fast prototyping tool for developing a detailed control law design specification that can be taken rapidly into real-time piloted simulation. The approach 1) enables functional full-flight envelope control laws to be worked up, including switching, blending, and logic strategies, with no investment in conventional control law design; 2) determines control power requirements for novel maneuvers or new configurations; 3) aids assessment of aircraft maneuver characteristics in off-line and piloted simulation; and 4) develops perfect control laws for use as an absolute benchmark to measure how good a given control law designed using conventional methods is.

Early work developed algorithms that provide ideal control of rotational aircraft motion by application of NDI pitch, roll, and yaw moments to decouple angular body accelerations.[2,5] These algorithms have been matured considerably since conception by a number of researchers to encompass the control of angular body rates, body attitudes, angle of attack, sideslip, and velocity vector roll responses.

This Note develops new NDI control laws for the translational modes required to maneuver vertical/short takeoff and landing (V/STOL) aircraft in the hover and in forward flight. The dynamical responses that are ultimately decoupled and prescribed by these control laws are the aircraft translational accelerations in body axes. The control laws determine the forces required to achieve commanded translational accelerations, velocities, and position. The required (x, y, z) forces are generated by three-dimensional thrust vectoring or by two-dimensional thrust vectoring in the x–z plane with lateral translation controlled by bank angle. Algorithms that provide forward and lateral translation motion control via pitch and bank angles are also presented. Application of the new translational motion control laws is demonstrated using off-line simulation of a V/STOL aircraft.

Translation Motion Control Laws

This section develops NDI algorithms for perfectly decoupled control of translation accelerations, velocities, and position via three-dimensional/two-dimensional thrust vectoring and aircraft attitudes. The algorithms require manipulation of the following body axis equations of motion for translational accelerations of a rigid aircraft[6]:

$$\frac{1}{m}\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} qw - rv \\ ru - pw \\ pv - qu \end{bmatrix} - \begin{bmatrix} -g\sin\theta \\ g\sin\phi\cos\theta \\ g\cos\phi\cos\theta \end{bmatrix} \tag{1}$$

To derive the control laws, the translational motion commands are first transformed from the axis system appropriate for forward flight and hover to the body axis reference frame. Then, simple reordering of Eq. (1) is performed to define the force that must be applied to each axis to achieve commanded translational accelerations.

Acceleration/Velocity Control in Hover via Thrust Vectoring

Translational acceleration and velocity commands in hover are expressed in the Earth axis reference frame. If $\psi = 0$, an Earth to body axis transformation in Eq. (1) gives

$$\frac{1}{m}\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ \cos\phi\cos\theta & \cos\phi & \sin\phi\cos\theta \\ \cos\phi\sin\theta & -\sin\phi & \cos\phi\cos\theta \end{bmatrix} \times \begin{bmatrix} a_{Ex\,\mathrm{dem}} \\ a_{Ey\,\mathrm{dem}} \\ a_{Ez\,\mathrm{dem}} \end{bmatrix} - \begin{bmatrix} -g\sin\theta \\ g\sin\phi\cos\theta \\ g\cos\phi\cos\theta \end{bmatrix} \tag{2}$$

Equation (2) determines the total forces required to satisfy commanded accelerations. Velocity demands in hover are translated into acceleration demands using $a_{Ex\,\mathrm{dem}} = (v_{Ex\,\mathrm{dem}} - v_x)V_{\mathrm{bw}}$. Control of commanded velocities is obtained by feeding approximated accelerations $a_{Ex\,\mathrm{dem}}$ into the acceleration control law of Eq. (2). If required, each matrix row of Eq. (2) can be used independently and mixed with different response types in other axes.

Acceleration/Velocity Control in Forward Flight via Thrust Vectoring

For the aircraft in forward flight, translational accelerations and speed relative to the flight path are considered. Commands expressed in wind axes are transformed to body axes:

$$\frac{1}{m}\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \begin{bmatrix} \cos\alpha\cos\beta & -\cos\alpha\sin\beta & -\sin\alpha \\ \sin\beta & \cos\beta & 0 \\ \sin\alpha\cos\beta & -\sin\alpha\sin\beta & \cos\alpha \end{bmatrix} \times \begin{bmatrix} a_{Wx\,\mathrm{dem}} \\ a_{Wy\,\mathrm{dem}} \\ a_{Wz\,\mathrm{dem}} \end{bmatrix} - \begin{bmatrix} -g\sin\theta \\ g\sin\phi\cos\theta \\ g\cos\phi\cos\theta \end{bmatrix} \tag{3}$$

Flight-path speed control is achieved by translation to acceleration using $a_{W\,\mathrm{dem}} = (v_{W\,\mathrm{dem}} - v_W)V_{\mathrm{bw}}$. For modest angles of attack and sideslip, F_x would be met by thrust (or airbrake), whereas the generation of F_y and F_z would require direct side force and lift control, respectively.

Three-Dimensional and Two-Dimensional Thrust Vectoring

The total force acting on the aircraft in any one direction comprises contributions from the body, wings, control surfaces, etc. (the aerodynamic forces), and the engine, i.e., $F_{\mathrm{Total}} = F_{\mathrm{aero}} + F_{\mathrm{engine}}$. Thus, for V/STOL aircraft with three-dimensional thrust vectoring, the total engine thrust and the demanded nozzle angles in the x–y and x–z planes required to satisfy acceleration commands are

$$\mathrm{Thrust}_{\mathrm{Total}} = \sqrt{F^2_{x\,\mathrm{engine}} + F^2_{y\,\mathrm{engine}} + F^2_{z\,\mathrm{engine}}}$$
$$\theta_{xy} = \tan^{-1}\left(\frac{F_{y\,\mathrm{engine}}}{F_{x\,\mathrm{engine}}}\right), \qquad \theta_{xz} = \tan^{-1}\left(\frac{F_{z\,\mathrm{engine}}}{F_{x\,\mathrm{engine}}}\right) \tag{4}$$

For aircraft with two-dimensional thrust vectoring in the x–z plane, translational motion control in all three axes requires the air-

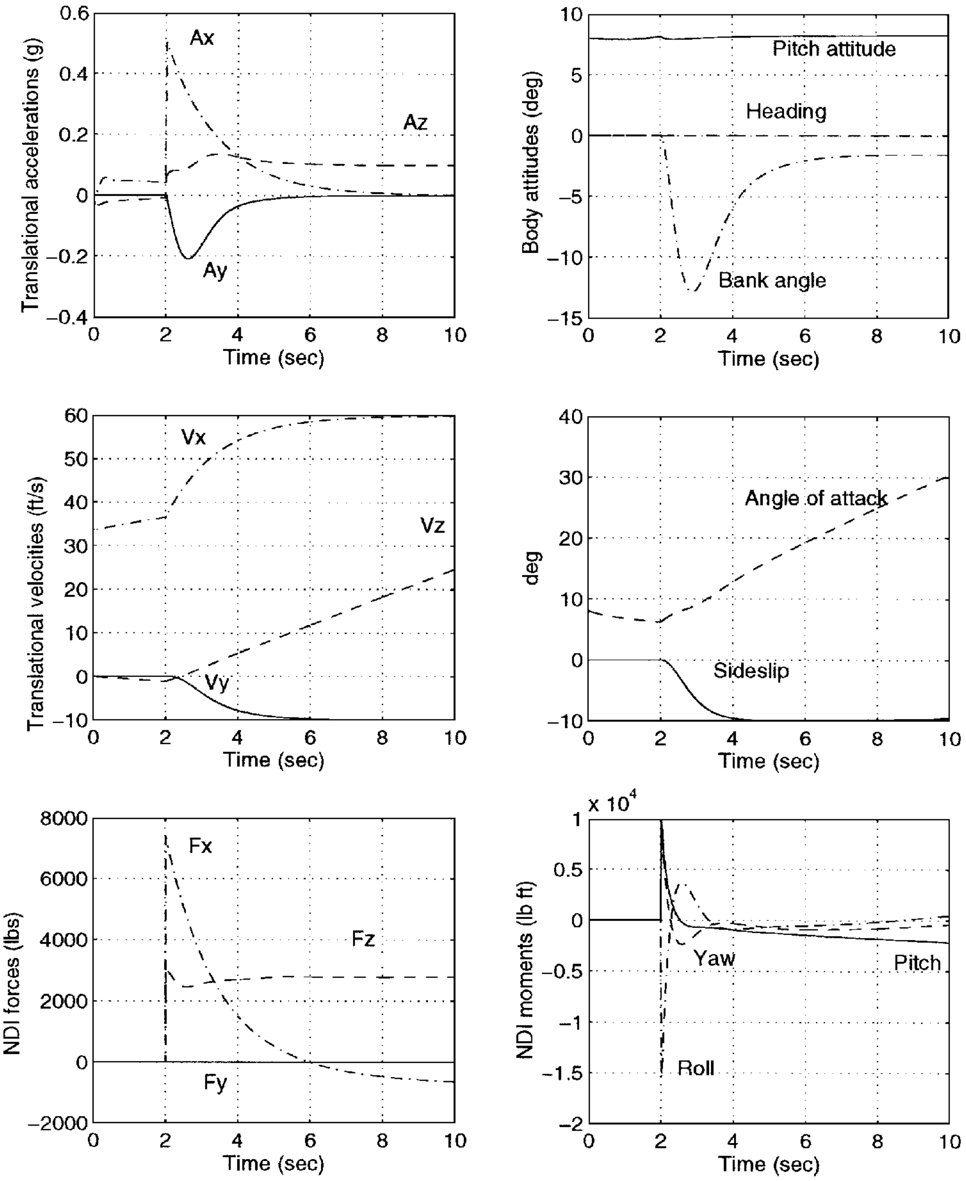

Fig. 1 Control of heave acceleration, forward velocity, and lateral velocity using two-dimensional thrust vectoring and attitudes.

craft to bank to generate the necessary side force. The commanded total thrust and nozzle angle, respectively, are calculated using

$$\text{Thrust}_{\text{Total}} = \sqrt{F^2_{x\,\text{engine}} + F^2_{z\,\text{engine}}}, \qquad \theta_{xz} = \arctan\left(\frac{F^2_{x\,\text{engine}}}{\text{Thrust}_{\text{Total}}}\right) \tag{5}$$

Acceleration/Velocity Control via Attitudes

These response types assume that thrust magnitude is variable, whereas thrust direction is held constant. Translational control is achieved by aligning the force and gravity components, using aircraft attitudes, to produce the desired body forces in the (x, y, z) directions. A rearrangement of Eq. (2) determines the pitch attitude and bank angles corresponding to commanded body accelerations to be

$$\theta = \arcsin\left[\frac{m(\dot{u}_{\text{dem}} + qw - rv) - F_x}{mg}\right]$$
$$\phi = \arctan\left[\frac{m(\dot{v}_{\text{dem}} + ru - pw) - F_y}{m(\dot{w}_{\text{dem}} + pv - qu) - F_z}\right] \tag{6}$$

The calculated attitudes are then fed into the rotational NDI control laws described in Ref. 5.

Position Control

This is achieved by applying forces that produce velocities in the direction of the commanded (x, y) position. For an aircraft at position (x_1, y_1) and heading ψ, in an arbitrary Earth axis reference frame with origin (x_0, y_0), its bearing relative to the origin is

$$\chi = \arcsin\left[\frac{y_1 - y_0}{\sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}}\right] \tag{7}$$

The desired longitudinal and lateral displacements from the origin are given by

$$s_{x\,\text{dem}} = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}\cos(\psi - \chi)$$
$$s_{y\,\text{dem}} = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}\cos(\psi - \chi) \tag{8}$$

The x and y displacements are transformed successively into velocity and then acceleration demands and fed into Earth axis acceleration response control laws to achieve the (x, y) position.

V/STOL Aircraft Responses

The translational motion control laws are applied to a V/STOL aircraft using off-line simulation. The NDI forces and moments are injected directly into the aircraft equations to achieve the prescribed response. The aircraft is required to attain heave acceleration $a_{Ez\,\mathrm{dem}} = 0.1\ g$ and forward velocity $v_{Ex\,\mathrm{dem}} = 60$ ft/s via two-dimensional thrust vectoring in the x–z plane and lateral translation rate $v_{Ey\,\mathrm{dem}} = -10$ ft/s via bank angle from a 32-ft/s trim condition. Figure 1 shows the time responses of the aircraft to these commands after 2 s of open-loop flight from trim. The responses show accurate achievement of the commands within 4–8 s. Some form of attitude control must be applied to prevent the aircraft rolling, pitching, and yawing during translation maneuvers. In this example, pitch attitude, calculated using Eq. (6) and implemented as described in Ref. 5, is set to be compatible with commanded translation to prevent the aircraft pitching.

These commands are specifically chosen to demonstrate the ability to mix different response types in the three body-referenced Earth axes. Response mixing requires NDI forces and moments to work together compatibly. With velocity and attitude/position commands, the authority of forces and moments can be controlled by the choice of bandwidths. The bandwidths limit accelerations and velocities and hence serve as a means of governing the degree of control exercised by each response type.

Conclusions

This Note has presented and demonstrated nonlinear dynamic inversion algorithms for translation motion control of V/STOL aircraft using three-dimensional and two-dimensional thrust vectoring. The use of the exact nonlinear dynamic inversion approach as a fast prototyping tool in control law design has been proposed. Further research is examining the effects of actuation, sensors, discrete digital elements, and structural dynamics to develop a very complete approach to aid engineers in the development of appropriate control laws and vehicle configurations.

To translate conceptual control into real or actual control, the dynamic inversion forces and moments must be mapped into demands on engine and aerodynamic control surfaces. The development of such methods with due regard to motivator redundancy, blending between hover and forward flight, robustness to gusts and aerodynamic coefficient uncertainties, and actuator limits continues to receive much research attention. The new algorithms have a useful role in resolving the first two of these issues because they make use of both aerodynamic and thrust vector control to achieve a single response.

References

[1]Morton, B. G., Elgersma, M. R., Harvey, C. A., and Hines, G., "Nonlinear Flying Quality Parameters Based on Dynamic Inversion," Flight Dynamics Lab., AFWAL-TR-87-3079, Wright–Patterson AFB, OH, Oct. 1987.

[2]Snell, S. A., Enns, D. F., and Garrard, W. L., Jr., "Nonlinear Inversion Flight Control for a Supermaneuverable Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 15, No. 4, 1992, pp. 976–984.

[3]Enns, D., Hendrick, R., Roatcap, R., and Caton, J., "Robustness of Dynamic Inversion vs μ Synthesis Lateral-Directional Flight Control Example," *Proceedings of the AIAA Guidance, Navigation, and Control Conference* (Washington, DC), AIAA, Washington, DC, 1990, pp. 210–222.

[4]Snell, A., "Cancellation Control Law for Lateral-Directional Dynamics of a Supermanoeuvrable Aircraft," *Proceedings of the AIAA Guidance, Navigation, and Control Conference* (Washington, DC), AIAA, Washington, DC, 1990, pp. 701–709.

[5]Smith, P. R., "Functional Control Law Design Using Exact Nonlinear Dynamic Inversion," *Proceedings of the AIAA Atmospheric Flight Mechanics Conference* (Washington, DC), AIAA, Washington, DC, 1990, pp. 481–486.

[6]Etkin, B., *Dynamics of Atmospheric Flight*, Wiley, New York, 1972, p. 103.

Mixed μ-Synthesis via H_2-Based Loop-Shaping Iteration

Ciann-Dong Yang*
National Cheng Kung University,
Tainan, Taiwan 701, Republic of China
Chia-Yuan Chang†
Ford Lio Ho Motor Company, Ltd.,
Chung Li, Taiwan 320, Republic of China
and
Yun-Ping Sun‡
National Cheng Kung University,
Tainan, Taiwan 701, Republic of China

I. Introduction

ENGINEERS encounter the robust control problems with respect to structured mixed (real and complex) uncertainty in many practical applications. Such a concept of mixed uncertainty provides control engineers with a more proper description and less conservative design than the general uncertain systems. Unfortunately, the computation of structure singular value with respect to mixed perturbation sets is very complex and known to be nonpolynomial (NP) hard.[1] As a result, the mixed μ-synthesis becomes a more challenging problem.

There are two representative approaches, D, G-K iteration and μ-K iteration,[2,3] to mixed μ-synthesis. Although a few successful designs have been presented, the practical use of D, G-K iteration is obstructed by the need for fitting purely complex scalings with high-order, all-pass transfer functions, as well as the need for fitting in phase and in magnitude. In comparison with D, G-K iteration, the advantage of μ-K iteration is that one needs to fit in magnitude only; however, the disadvantage is that one needs to compute not only the mixed but also the corresponding complex μ upper bounds at each iteration. Therefore, the μ-K iteration still is quite involved and computationally demanding.

A new approach, called the H_2-based loop-shaping iteration, has been proposed and successfully applied to H_∞-synthesis and the complex μ-synthesis.[4,5] This H_2-based mixed μ-synthesis approach is totally different from the H_∞-based mixed μ-synthesis approaches (D, G-K iteration and μ-K iteration). The mixed μ-controller is synthesized by performing a sequence of weighted H_2 optimizations, not H_∞ optimizations; therefore, this approach is less computationally demanding. In addition, the algorithm is quite easy to implement.

II. Reviews on μ

Consider a generalized augmented plant P partitioned as

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \tag{1}$$

To test for robust performance, we first pull out the uncertain perturbations and rearrange the uncertain system into an $N\Delta$-structure. N is the lower linear fractional transformation of augmented plant P closed by controller K, and M is the upper linear fractional transformation of N closed by uncertainty Δ:

$$\begin{aligned} N &= F_l(P, K) \\ &= P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{21} \end{aligned} \tag{2}$$

Received March 15, 1997; revision received Sept. 10, 1997; accepted for publication Sept. 11, 1997.

*Professor, Institute of Aeronautics and Astronautics. Member AIAA.

†Senior Program Engineer, Program Engineering Management, Local Vehicle Team.

‡Graduate Student, Institute of Aeronautics and Astronautics.