This function does not provide the same value for optimum v as does the Hamiltonian or the equivalent functions, F_1 and F_2. Comparing F_3 with F_1, it will be recognized as the approximation in which λ_w is replaced by its final value c_w and the error introduced therefore depends on the extent to which λ_w differs from c_w during the earlier part of the flight.

Some indication of the extent to which the airspeed computed from F_3 deviates from optimum is provided by two numerical examples using an aircraft with idealized aerodynamic and propulsive characteristics. The lift, assumed equal to weight, is given by

$$L = \tfrac{1}{2}\rho v^2 S C_{L_\alpha} \alpha$$

and the drag, assumed equal to thrust, is given by

$$D = \tfrac{1}{2}\rho v^2 S (C_{D_0} + C_{L_\alpha} \alpha^2)$$

where ρ is the air density, 0.001267 slugs/ft^3 at 20,000 ft; S the reference area, 3500 ft^2; and the aerodynamic coefficients $C_{L_\alpha} = 10$ and $C_{D_0} = 0.015$. The fuel flow is given by $f = kT$, where k has the value 0.8 lb/h/lb$_{\text{thrust}}$. For a given weight and airspeed, the first of these two equations defines the required angle of incidence α and substitution into the second gives the thrust T required to maintain that airspeed. Multiplication by k then gives the fuel flow. These equations, for a given value of weight, define the function $f(v)$. In the examples given here, the differential equations were integrated forward using a Runge-Kutta method, with iteration procedures to satisfy the final boundary conditions.

With an initial weight of 300,000 lb and cost factors of \$1/lb of fuel and \$1/s of flight time, the optimum velocity profile for a range of 5000 mi. is given in the second column of Table 1. The optimum flight time and final weight are 44,456 s and 109,926 lb. The velocity profile computed from F_3, given in the third column, shows a significant deviation from optimum in the early part of the flight. The flight time and weight in this case are 45,087 s and 110,493 lb, for an additional cost of \$64 with respect to the optimum.

In the case where fuel usage is to be minimized with both range and flight time specified, sometimes referred to as four-dimensional navigation, the same differential equations are integrated but with the boundary condition, $H = c_t$, replaced by the terminal condition, $t = t_f$. The same cruise cost function F_3 has been proposed by Sorensen and Waters[6] and Burrows[7] to compute such trajectories, now using c_t as a parameter to adjust the time of flight. As may be expected, this provides suboptimal flight paths. An example is given in Table 2 for the same aircraft flying over the same range, but with the flight time specified as 62,000 s. The computed velocity profile given in the third column is quite different from the optimum profile in the second column and the fuel usage is 621 lb more than optimum, representing a very significant 0.31% loss. Further details of these computations can be made available on request to the authors.

Conclusions

Cruise trajectories computed by means of the cruise cost function proposed by Erzberger and others and, therefore, also the global flight paths of which they are a part, have been found to be suboptimal. The extent to which the solutions deviate from the optimum may be small in many cases, but an example has been given in which the deviation is quite significant, amounting to a fuel loss of 0.3%. This is an extreme case with an extended flight time unlikely to be used in commercial operations, but it serves to indicate that significant errors can arise and that caution is needed in applying the approximate method, particularly in four-dimensional applications. It is suggested that, unless the error can be estimated or at least bounded for each trajectory, an exact cruise cost function should be used, possibly selected from those discussed above.

References

[1]Bryson, J.W. and Ho, Y., *Applied Optimal Control,* Hemisphere, New York, 1975, Chap. 2.

[2]Burrows, J.W., "Fuel-Optimum Trajectory Computations," *Journal of Aircraft,* Vol. 19, April 1982, pp. 324-329.

[3]Gordon, C.N., "Flight Software for Optimal Trajectories in Transport Aircraft," AIAA Paper 83-2241, 1983.

[4]Barman, J.F. and Erzberger, H., "Fixed Range Optimum Trajectories for Short-Haul Aircraft," *Journal of Aircraft,* Vol. 13, Oct. 1976, pp. 748-754.

[5]Erzberger, H. and Lee, H.Q., "Algorithm for Fixed Range Optimal Trajectories," NASA TP 1565, July 1980.

[6]Sorensen, J.A. and Waters, M.H., "Airborne Method to Minimize Fuel with Fixed Time-of-Arrival Constraints," *Journal of Guidance and Control,* Vol. 4, June-July 1981, pp. 348-349.

[7]Burrows, J.W., "Fuel-Optimal Aircraft Trajectories with Fixed Arrival Times," *Journal of Guidance, Control and Dynamics,* Vol. 6, Jan.-Feb. 1983, pp. 14-19.

Multivariable Control Robustness Examples: A Classical Approach

Bong Wie*
The University of Texas at Austin, Texas
and
Arthur E. Bryson Jr.†
Stanford University, Stanford, California

Introduction

THIS Note discusses two examples that have been used to demonstrate the superiority of multivariable control analysis/synthesis techniques based on singular values over classical techniques. Classical stability analysis, by breaking loops one at a time, is known to be an unreliable way of testing robustness to simultaneous perturbations of all the loops. However, classical stability analysis, by successively closing loops, is more reliable. The difference between these two classical approaches has not been well recognized. In this Note, we show that the successive-loop-closure way of analyzing these two examples does predict a lack of robustness in the nominal designs and, furthermore, provides insights into how the design can be changed so as to be more robust.

Examples of Multivariable Control Robustness

Two examples that have been used for the discussion of multivariable control loop robustness are:

Example 1 (Refs. 1, 2):

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \frac{1}{s^2+100} \begin{bmatrix} s-100 & -10(s+1) \\ -10(s+1) & s-100 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (1)$$

Example 2 (Ref. 3):

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \frac{1}{(s+1)(s+2)} \begin{bmatrix} -47s+2 & 56s \\ -42s & 50s+2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (2)$$

Received July 3, 1986; revision received Aug. 7, 1986.

*Assistant Professor, Department of Aerospace Engineering and Engineering Mechanics. Member AIAA.

†Paul Pigott Professor of Engineering, Department of Aeronautics and Astronautics. Fellow AIAA.

where (u_1,u_2) are the two control inputs and (y_1,y_2) the two measured outputs to be controlled. A constant-gain diagonal feedback control logic is given as $u_1=-K_1y_1$ and $u_2=-K_2y_2$. Note that K_1 and K_2 are the loop gains that are different from the gain perturbations from the nominal design used in Refs. 1-4. The nominal design point is given as $K_1=K_2=1$.

Example 1 was used in Refs. 1 and 2 to illustrate deficiencies in the classical single-loop-at-a-time method. Example 2 was used in Ref. 3 to illustrate deficiencies in the inverse Nyquist array and characteristic loci approaches. It was claimed that the classical single-loop-at-a-time method fails to predict a serious lack of robustness in the nominal design for example 1. It was also claimed that the inverse Nyquist array and characteristic loci approaches fail to predict a lack of robustness in the nominal design for example 2. However, these approaches with perfect diagonalization do not consider independent perturbations of the loop gains; rather, they consider only simultaneous perturbations, keeping $K_1=K_2$. This Note shows that the successive-loop-closure approach does predict a lack of robustness in the nominal designs for these examples.

First, consider example 1. Figure 1 shows the root locus vs K_1 of the first loop closure; for $K_1=1$, the closed-loop poles are at $s=0$ and -1. For $K_1>1$, the first loop closure becomes unstable.

After the first loop closure $(u_1=-K_1y_1)$, the second loop transfer function becomes

$$\frac{y_2}{u_2}=\frac{s+(101K_1-100)}{s^2+K_1s+100(1-K_1)} \tag{3}$$

The *zero* as well as the poles are changed by the first loop closure. The coupling numerator polynomial, defined in the Appendix, $N_c(s)=101$ for this example. The transfer function zero of the second loop closure is changed by the coupling numerator polynomial. For $K_1=1$, the root locus vs K_2 shown in Fig. 2b indicates a $\pm\infty$dB gain margin. However, it is not reasonable to conclude that the system has a $\pm\infty$dB gain margin, since this result depends on an exact pole-zero cancellation at $s=-1$. A root locus vs K_2 with $K_1=0.9$ shows that one of the complex poles at $s=-0.45\pm j\,3.13$ approaches the zero in the right half-plane (see Fig. 2a). This will result in a finite positive gain margin. For $K_1=1.1$, the second loop transfer function will have a positive real zero but an unstable pole (see Fig. 2c). This will result in a finite negative gain margin.

If only the $1/s$ transfer function were given to a controls analyst who was aware that the original system was second order, then it would be natural for the analyst to investigate the effects of pole-zero cancellation. While the singular value analysis predicts a lack of robustness in the nominal design for this example, it does not show *why* the nominal design is so sensitive to the gain perturbations. As shown in Figs. 1 and 2, a classical loop-closing approach using simple root loci clearly indicates a lack of robustness in the nominal design and, furthermore, provides a direct way of modifying the nominal design to have a larger stability margin.

Although this example does not represent any particular physical system, a nominal design from a classical control point of view should be $u_1=+K_1y_2$ and $u_2=-K_2y_1$, where K_1 and K_2 are positive gains. It can be easily shown that this nominal design truly has $\pm\infty$dB gain margins for each loop.

Similar conclusions can be made for example 2. Figure 3 shows the root locus vs K_1 of the first loop closure; for $K_1>0.064$, the first loop closure becomes unstable. For $K_1=1$, the closed-loop poles are at $s=+0.09$ and $+43.91$. After the first loop closure $(u_1=-K_1y_1)$, the second loop transfer function becomes

$$\frac{y_2}{u_2}=\frac{50s+2(1+K_1)}{s^2+(3-47K_1)s+2(1+K_1)} \tag{4}$$

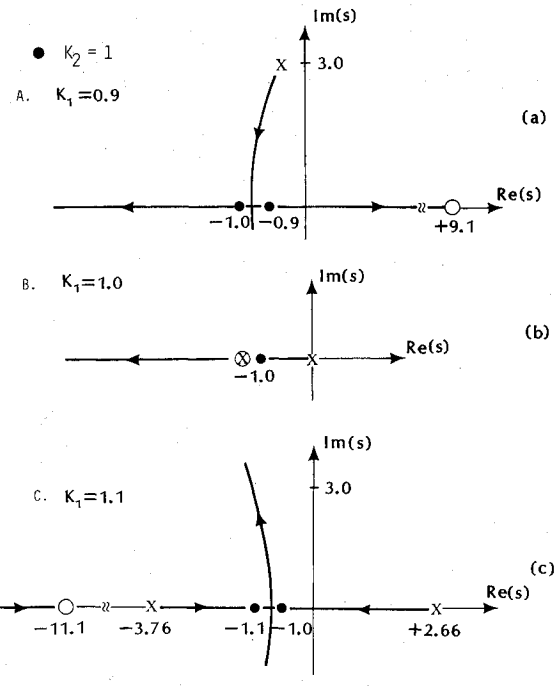

Fig. 2 Root loci vs K_2 with perturbed values of K_1 (example 1).

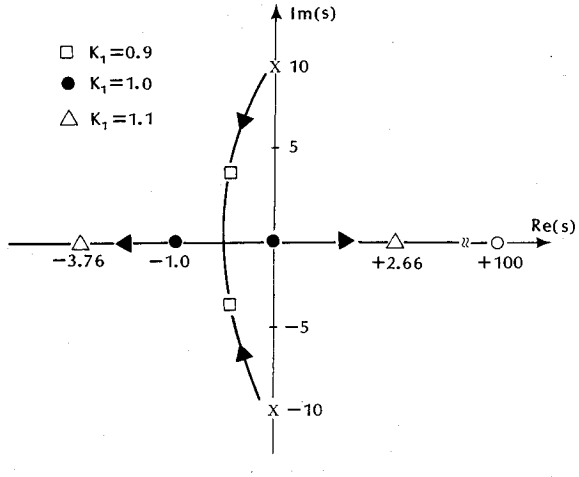

Fig. 1 Root locus vs K_1 (first loop closure of example 1).

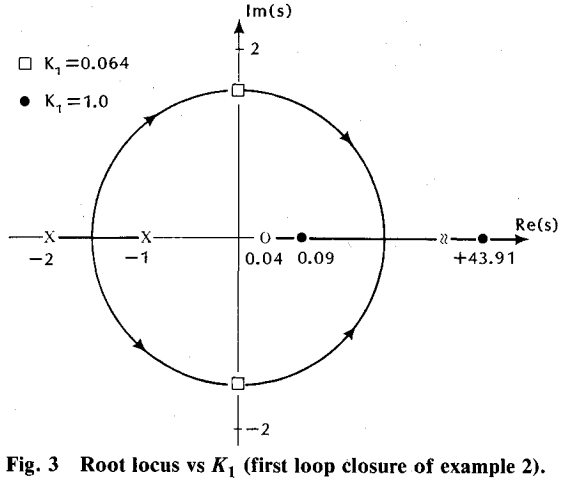

Fig. 3 Root locus vs K_1 (first loop closure of example 2).

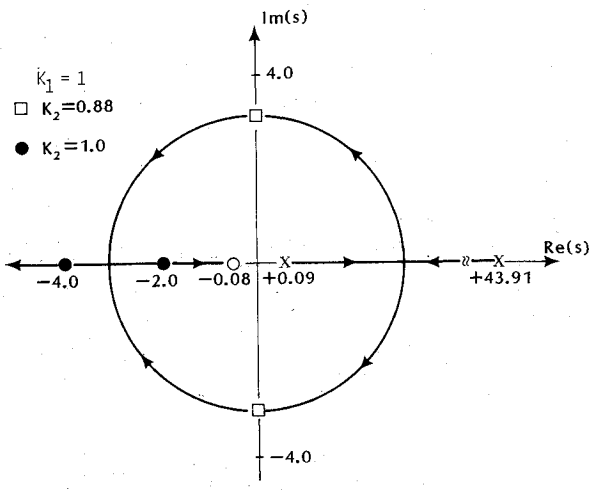

Fig. 4 Root locus vs K_2 with $K_1 = 1$ (example 2).

The coupling numerator polynomial $N_c(s) = 2$ for this example. As can be seen from Fig. 4, the second loop closure with $K_2 = 1$ brings the unstable poles back to the stable region at $s = -2$ and -4. While Ref. 3 states that this example has a $\pm\infty$dB margin from the inverse Nyquist array and characteristic loci approaches, it can be seen from Fig. 4 that it has $+\infty$dB and only -1.1 dB gain margins (imaginary axis crossing at $K_2 = 0.88$). Similarly, by closing the K_2 loop first, it can be shown that the K_1 loop has $-\infty$dB and only $+1.0$ dB gain margins. These approaches with perfect diagonalization do not consider independent gain perturbations; they consider simultaneous gain perturbations, keeping $K_1 = K_2$. The $\pm\infty$dB gain margin mentioned in Ref. 3 should not be interpreted as an independent gain variation.

Conclusions

This Note has discussed two examples that have been used for the demonstration of the superiority of multivariable control analysis/design techniques based on singular values over other design approaches. We have shown that the classical successive-loop-closure approach using simple root loci provides clear indications of lack of robustness for these examples and also gives insights into how the design can be changed so as to be more robust.

Appendix

This Appendix briefly describes a classical successive-loop-closure method[5] applied to a dynamic system with two inputs and two outputs:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \frac{1}{D(s)} \begin{bmatrix} N_{11}(s) & N_{12}(s) \\ N_{21}(s) & N_{22}(s) \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad \text{(A1)}$$

where s is the Laplace transform variable, y_1 and y_2 are the outputs, u_1 and u_2 are the control inputs, $D(s)$ is the characteristic determinant, and the $N_{ij}(s)$ are the numerator polynomials.

A diagonal feedback control logic is assumed:

$$u_1 = -K_1(s)y_1, \qquad u_2 = -K_2(s)y_2 \quad \text{(A2)}$$

where $K_1(s)$ and $K_2(s)$ are the diagonal feedback compensators. The closed-loop characteristic equation then becomes

$$D^2 + (K_1N_{11} + K_2N_{22})D + K_1K_2(N_{11}N_{22} - N_{12}N_{21}) = 0 \quad \text{(A3)}$$

Using the relation[5]

$$N_{11}N_{22} - N_{12}N_{21} = D(s)N_c(s) \quad \text{(A4)}$$

where $N_c(s)$ is defined as the coupling numerator polynomial, the closed-loop characteristic equation can be written as

$$D + K_1N_{11} + K_2N_{22} + K_1K_2N_c = 0 \quad \text{(A5)}$$

After the first loop closure, the characteristic equation of the second loop can be written as

$$1 + \frac{K_2(N_{22} + K_1N_c)}{D + K_1N_{11}} = 0 \quad \text{(A6)}$$

The second loop compensator, $K_2(s)$ can then be designed in a manner similar to the method for the first loop design. However, the *zeros* as well as the poles of the second loop transfer function are changed by the first loop closure. The new zeros are related to the coupling numerator. This property is useful to find the new zeros for the second loop closure in the root locus analysis. However, a detailed assessment of the multivariable robustness using the coupling numerator needs further research.

References

[1]Doyle, J.C., "Robustness of Multiloop Linear Feedback Systems," Presented at 17th IEEE Conference on Decision and Control, San Diego, CA, Jan. 1979.

[2]Doyle, J.C., "Multivariable Design Techniques Based on Singular Value Generalizations of Classical Control," AGARD GRP Lecture Series No. 117, Oct. 1981.

[3]Doyle, J.C. and Stein, G., "Multivariable Feedback Design: Concepts for a Classical/Modern Synthesis," *IEEE Transaction on Automatic Control,* Vol. 26, Feb. 1981, pp. 4-16.

[4]Hartman, G.L. and Wall, J.E., "Multivariable Control Design for Performance and Robustness," AIAA Professional Study Seminar Notes, Seattle, WA, Aug. 18-19, 1984.

[5]McRuer, D., Ashkenas, I., and Graham, D., *Aircraft Dynamics and Automatic Control,* Princeton University Press, Princeton, NJ, 1973, pp. 163-177.

Relationship Between Kane's Equations and the Gibbs-Appell Equations

Edward A. Desloge*
Florida State University, Tallahassee, Florida

I. Introduction

In an earlier article in this journal Kane and Levinson[1] have shown that a certain general set of equations is particularly well suited to obtaining specific equations of motion for complex spacecraft. In subsequent books and articles Kane refers to these equations as "Kane's equations."[2] In this paper we will show that Kane's equations are simply a particular form of the Gibbs-Appell equations, which were first discovered by Gibbs[3,4] in 1879, independently discovered and studied in detail by Appell[5-8] twenty years later, and discussed in a number of textbooks.[9-14] We will do this by showing that Kane's equations are an intermediate set of equations that occurs in the derivation of the Gibbs-Appell equations from Newton's equations.

Received May 15, 1986; revision received July 28, 1986.

*Professor, Physics Department.