THE UMIST CONTROL SYSTEM DESIGN AND SYNTHESIS SUITES

N. Munro

Control Systems Centre, UMIST, Manchester, England

Abstract. Two suites of Fortran programs, which allow the design and synthesis of multivariable control systems are discussed. Both suites operate in an interactive conversational mode, and make appropriate use of graphic output. Powerful data input-output facilities and data transformation routines are provided. These are augmented by floxible analysis and automatic system simulation programs.

Keywords. Computer-aided design; computer-aided synthesis; interactive programs; inverse Nyquist array; characteristic locus; pole assignment; decoupling; optimal control.

INTRODUCTION

The digital computer with graphical display terminals offers a very powerful facility for control system design and synthesis studies. The well established design techniques developed by Nyquist, Bode, Nichols, and Evans all rely heavily on information presented in graphical form, and are ideally suited for implementation on interactive computing facilities with graphical displays. Recent developments in frequency-domain design methods for multivariable systems, which essentially consist of generalisations of the 'classical methods', are also well suited to such implementations. These latter techniques depend entirely on the use of a digital computer to carry out the necessary calculations, which are both complex and tedious. The prime results of these calculations are presented to the control system designer in graphical forms, which provide information about the performance of the system being studied, and give useful guidance on how this performance can be changed by the use of suitable compensators and feedback action.

In addition to design tools many useful synthesis procedures, such as pole assignment, decoupling, and optimisation algorithms, play an essential part in the achievement of a desired performance from a system. These approaches arose largely from state-space theory and usually require the designer to specify exactly what he wants to achieve. This, in turn, implies the solution which is then found algorithmically. Although the synthesis procedures were originally intended for use in a batch-mode environment, their implementation in an on-line conversational mode offers considerable advantages to the control system designer.

In order to carry out computer-aided studies on control systems, the control engineer needs various other facilities. These consist of a means of readily entering data defining a system model, or defining a trial compensator, into the computer in a variety of forms. This data may then need to be manipulated into other forms for subsequent analysis, design, and simulation studies. It is also important to have available the output routines necessary for the presentation of frequency-response and time-response data in a meaningful manner.

A large set of modular overlaid FORTRAN IV programs, all operating in an interactive conversational mode, have been implemented for these purposes on a DEC-10 computer situated in the Control Systems Centre at UMIST. These facilities, to be described briefly in the remainder of this paper, consist of two separate suites of software. One of these provides the frequency-domain design techniques, and the other provides the synthesis methods. Certain facilities are common to both suites of programs and will only be described once. However, before examining the software suites, the nature of the control problem giving rise to the need for these facilities is first considered.

THE NATURE OF THE CONTROL PROBLEM

Fig. 1 shows the general nature of the control problem and thus highlights the facilities needed for the computer-aided study of such systems. As a result of a system not meeting a set of 'desired specifications', a 'control problem' exists. Before we can start a control study, some form of description of the system (usually in the form of a mathematical model) is required. This model is usually nonlinear and has to be linearised about one, or more, operating points to obtain a set of linear models for the subsequent control studies.

Simulation of the nonlinear system model and of the associated linear models provides valuable time-domain characterisation of the system. Analysis of the linear models provides the designer with further system characteristics; e.g. order, stability, controllability, observability, structure; which provide initial guidance as to how effective certain design or synthesis tools will be.

Once an appropriate design or synthesis method has been selected, the procedure to be followed to try to meet the desired system behaviour usually becomes iterative with respect to the resulting linear system control studies. The testing of the resulting control scheme on the full non-linear model often involves the examination of such problems as fine tuning, scheduling, and implementation. Again, simulation is required before final consideration can be given to the implementation of the proposed control scheme on the physical system.

The computing facilities developed at UMIST to examine some of these aspects of a control system study are considered in the remainder of this paper.

THE FREQUENCY-DOMAIN DESIGN SUITE

This computer-aided design suite sonsists of several major software packages (see fig.2), which are briefly described in the following paragraphs. All communication with the suite is via the user's (graphics) terminal, which activates the various facilities by means of predefined mnemonic command words. The supervisor program (SUPER) simply acts as an overall communications module which brings the desired major package into use and passes the appropriate information to its local supervisory program (called an 'executive', but not shown explicitly in fig.2). The main facilities provided in this suite consist of a flexible data input-output package (I/O), a data manipulation package (DATA), a single-input single-output system design package (SISO), a multivariable system design package (MIMO), and a simulation package (TIME). In addition, there is a graphics package (GRAF) and a set of data base management routines; the latter being invisible to the user.

Data I/O

The data input-output package accepts a system description in four different forms; namely, measured frequency-response data, transfer-function descriptions. Rosenbrock's system matrix (Rosenbrock, 1970), or state-space equations. Constant data can be entered into matrices or vectors either element-by-element, or by row, or by column, or in diagonal form. Polynomial data can be entered as a series of polynomial factors or as a single polynomial; e.g., $(s^3+4s^2+0.001s+10.0)(2s^2+3.6)$ would be entered on-line as (s3+4.0s2+1.0E-3s+10.0)(2.0s2+3.6), where the round brackets are used to delimit the factors.

All data describing a system is stored in compact files on the system disk-store in the originally entered form. However, polynomial data in factored form is multiplied out into a single polynomial for in-core usage. Transfer-function descriptions including delay-terms and/or common denominators are readily handled. Most input errors are trapped by the programs, and it is virtually impossible for random teletype character errors to cause termination of execution. Data input/output is also provided with respect to the system disk-store and other peripherals. In addition, a data nodification facility is provided which allows individual elements of data to be changed without having to re-enter the entire data-file for the item concerned.

Data Manipulation (DATA)

The data manipulation package provides automatic transformation of data from any one of the above forms to any of the others. This facility makes use of techniques such as curve fitting (Zaman and Griffin, 1970), system matrix manipulations (Rosenbrock, 1970), the Faddeev algorithm (1963), and a minimal realisation algorithm (Munro and McLeod, 1971). Inversion routines for both complex matrices and rational polynomial matrices (Munro and Zakian, 1970) are also provided. These manipulations are primarily needed to put the data into the correct form for any subsequent analysis, design, or simulation required. The local executive program determines the sequence of operations required for a particular transformation, from a look-up table, and brings the necessary routines into use automatically. Certain transformations are undefined when time-delays are present in the system description. (See Fig. 3).

Single-Input Single-Output System Design (SISO)

This package provides the user with the well established linear system design techniques of Nyquist, Bode, Nichols and Evans (see D'Azzo and Houpis, 1960). In each case, the user can specify and change the description of a cascade or feedback compensator or the system, as required. The implementation of the root-locus facility used; due to Ash and Ash (1968); allows systems with time-delays to be handled, thus making this approach fully compatible with the other frequency-domain methods.

Multivariable System Design (MIMO)

The multivariable control system design

facilities provided consist of Rosenbrock's inverse Nyquist array design technique (Rosenbrock, 1974; Munro, 1972), and MacFarlane's characteristic locus design technique (Belletrutti and MacFarlane, 1971; MacFarlane and Kouvaritakis, 1977).

The inverse Nyquist array design technique essentially presents an approach to the multivariable control problem whereby an initial compensator is determined which makes the system 'diagonal dominant' and reduces the design of the original interacting system with m-inputs and m-outputs to the design of m single-loop controllers. The design is carried out in the frequency domain and tends to produce simple control schemes. Also, engineering constraints are more easily satisfied than with some of the synthesis techniques. In the initial design stages, the polar plots of all of the elements of the inverse matrix being considered are displayed. As the design process continues it becomes necessary to display only the polar plots associated with the diagonal elements of the inverse matrix, and finally the polar plots of selected diagonal elements. The characteristic-locus design technique depends on an extension of the concept of eigenvalues and eigenvectors of constant matrices to matrices of rational polynomial forms. Here, the frequency behaviour of the characteristic values and characteristic vectors of the appropriate transfer-function matrix are displayed and modified by the design of a suitable multivariable controller. Both Bode-type plots and multiple direct Nyquist plots are used with this approach. In this latter approach, the concept of dominance is not required.

Rosenbrock's INA method for continuous systems has also been extended to deal with single-rate multivariable sampled-data systems (Munro and Ibrahim, 1974). Here, the controller design is carried out in the fictitious-frequency domain W; using the bilinear transformation $W = (z-1)/(z+1)$; in a manner analogous to that of continuous systems.

In all cases, the user can design pre or post-compensators and feedback compensators (see Fig. 4), and can readily determine their effect on the resulting system by generating and examining the appropriate graphic representations.

System Simulation (TIME)

The simulation package (Munro and Bowland, 1972) provides a means of evaluating the time-responses of both open-loop and closed-loop configurations of the type considered in Fig. 4. Both continuous and single-rate sampled-data system responses to step changes on the system inputs can be obtained for system components described by transfer-function descriptions, state-space equations, or a mixture of both. In the former case, the equations are automatically transformed to a minimal state-space realization (non-minimal, if transport delays are present). The resulting equations are then automatically assembled for integration using a variable step-length Runge-Kutta-Merson algorithm. The time responses so obtained can be displayed individually or superimposed, as required. Identical and individual scaling of responses is provided interactively. The system configuration and definition of the desired simulation are also set up interactively.

Graphics Package

The graphics facilities implemented allow the user to display and graphically edit direct and inverse Nyquist plots, Bode diagrams, Nichols charts, root-locus plots, inverse Nyquist arrays, characteristic loci and misalignment plots, and also system time-responses. Facilities such as origin shifting, element magnification and automatic scaling are provided. The graphic facilities are classed as part of the I/O package.

THE MULTIVARIABLE SYSTEM SYNTHESIS SUITE

In addition to the multivariable system frequency-domain design suite, a set of synthesis techniques is provided in the form of a separate facility. The structure of this suite of programs is shown in Fig. 5, where it can be seen that certain facilities such as the data input-output routines, data manipulation facilities, and simulation programs are common to those developed for the frequency-domain design methods. Again, this suite of programs can be seen to consist of several major packages; each with its own local supervisory program (or 'executive'), all linked by an overall supervisor program (SUPER). All directives entered by a user in the form of mnemonic command words are interpreted by the supervisory modules, which simply activate the required facilities. The main new features to be considered in this set of programs are the facilities provided in the ANALYSIS and SYNTHESIS packages. Data input-output is performed by a flexible (I/O) package (as previously described), and a powerful data manipulation facility (DATA) enables this data to be transformed into any other desired form (see Fig. 3) as required by the synthesis procedure, or analysis procedure, to be applied. These new facilities are now briefly described, in the following sub-sections.

Analysis Package (ANALYSIS)

This package contains such routines as are necessary for the calculation of eigenvalues and eigenvectors, system 'zeros', and singular values. In addition, systems described by state-space equations can be transformed to controllable or observable standard forms. The eigenvalue and eigenvector calculations provided are carried out using the QR-algorithm and eigenvector routines available in EISPAC (1978). System zeros are calculated using the generalised eigenvalue approach, or QZ algorithm, as suggested by Laub and Moore (1976). An alternative method for the determination of system zeros due to Davison and Wang (1974), based on eigenvalue calculations using the QR-algorithm, is also provided. Although the QZ method is numerically robust, there are often difficulties in interpreting the results, particularly in the case of zeros at infinity, and the availability of two separate procedures is often useful in resolving such difficulties. The routine used for the determination of the singular values of a system is also taken from EISPAC (1978) and is particularly useful in resolving certain rank questions (see Strang, 1976) which arise in the determination of the controllability and observability of a system, and in the determination of the consistency of sets of equations (Strang, 1976) which arise in connection with certain pole-assignment algorithms (Munro and Novin-Hirbod, 1979). The transformation techniques required to manipulate a state-space system into its corresponding controllable or observable standard form are described in Munro (1974; 1977) and Munro and Vardulakis (1974).

Although most of the algorithms provided in the ANALYSIS package are also used implicitly by many of the synthesis techniques provided, it is often useful to have them available explicitly.

Synthesis Package (SYNTH)

The synthesis procedures implemented to date consist of procedures for the decoupling of systems for both the state and output-feedback cases, procedures for pole assignment using both state and output-feedback, and optimal control for the quadratic cost-function case. Several different pole assignment algorithms are implemented, see Fig. 6, and provide both the spectral approach (Munro, 1969) and the mapping approaches (Young and Levsen, 1970). Both dyadic and full-rank minimal degree compensator synthesis procedures are provided (Novin-Hirbod, 1978). These allow a variety of pole assignment techniques to be rapidly applied to a given problem and afford the designer a choice of the resulting control schemes. The decoupling algorithms implemented consist of both the state-feedback approach, as described by Falb and Wolovich (1967), and the frequency-domain approach using output-feedback as described by Wolovich (1975). For systems having 'weak inherent coupling' (see Gilbert, 1969), the approach suggested by Wang (1970) can readily be applied using the procedures implemented.

The optimal control facilities provided at present cater only for the steady-state, or infinite settling time, case. The required optimal controller is determined using the eigenvalue/eigenvector approach suggested by MacFarlane (1969), once the appropriate state and input-weighting matrices have been specified interactively by the user. The various weighting factors can be readily altered and the corresponding controller obtained. The resulting feedback systems can each be readily simulated to observe the effects of these variations.

In addition to these synthesis procedures, algorithms have also been implemented for the design of both Luenberger observers (Munro and Vardulakis, 1974) and Kalman filters, which provide good estimates of the state-vector for those cases where the output-feedback approach is not suitable. In the case of the Kalman filter only the steady-state case is currently available and is implemented using the approach suggested by MacFarlane (1969). Finally, a simple but flexible automatic system simulation package (as previously described) provides facilities for the open or resulting closed-loop system responses to be examined.

OTHER FEATURES OF THE CAD SOFTWARE

All data, whether input by the user or generated by the design packages, is stored in data files on the system disk-store. Such a scheme may be likened to using the disk as a virtually unlimited dynamic store whose contents are available to all programs. With such an arrangement it is necessary to define suitable mnemonics by which the user may refer to a given set of data, and also to define the filename(s) under which that data is to be found in the user's own disk-area. All individual programs adhere to this mnemonics list.

In addition to the mnemonic names used to identify system data, a set of mnemonic command words is used to obtain the desired action from the CAD suites in a conversational mode. These mnemonics can be entered arbitrarily in reply to a program generated request for either a directive or data, at almost any point in a conversation. They are used when the user wishes to either interrupt the normal conversation to branch to some other part of the current conversation (or to some other part of another conversation or in reply to the question "WHAT NOW?", which signifies the end of the current conversation.

The reply "?", which is a request for help, entered by the user, causes explanatory texts to be printed out. This feature is extremely valuable since the possible answers to the various questions generated by the C.A.D. suites form a rather comprehensive list, which is difficult to remember. In some cases, two levels of explanation are provided: one to prompt a familiar user, and one to give a more detailed account of available facilities.

CONCLUDING REMARKS

Many of the algorithms used throughout the facilities described above are numerically weak. This is particularly so of the techniques used to manipulate data in polynomial form. It would be of significant value to the control community therefore, if some professional effort could be brought to bear on these problems.

In the development of large facilities of the type described in this paper, the efforts of many people are involved. These contributions are hereby gratefully acknowledged, and in particular the recent efforts of Dr. S. Novin-Hirbod (1978) in the development of the synthesis facilities. Acknowledgements are also due to the Science Research Council for providing the facilities on which this work has been carried out.

REFERENCES

Ash, R.H. and G.R. Ash (1968). Numerical computation of root-loci using the Newton-Raphson technique. IEEE Trans. on Auto. Control, Vol.AC13, pp.576-582.

Belletrutti, J.J., and A.G.J. MacFarlane (1971). Characteristic loci techniques in multivariable control system design. Proc. IEE, Vol.118, No.9, pp.1291-1297.

Davison, E.J., and S.H. Wang (1974). Properties and calculation of transmission zeros of linear multivariable systems. Automatica, Vol.10, pp.643-658.

D'Azzo, J.J., and C.H. Houpis (1960). Feedback Control System Analysis and Synthesis. McGraw-Hill, New York.

EISPAC-Release 2 (1978). Eigensystem Subroutine Package, Argonne National Laboratory, Argonne, Illinois, USA.

Faddeev, D.K., and V.N. Faddeeva (1963). Computational Methods of Linear Algebra. Freeman, San Francisco, California.

Falb, F.L., and W.A. Wolovich (1967). Decoupling in the design and synthesis of multivariable control systems. IEEE Trans. on Auto. Control, Vol.12 (6), pp.651-659.

Gilbert, E.G. (1969). The decoupling of multivariable systems by state-feedback. Siam J. Control, Vol.7(1), pp.50-63.

Laub, A.J., and B.C. Moore (1976). Calculation of transmission zeros using QZ techniques. Systems Control Report No.7618, University of Toronto, Control Science and Engineering Department, Toronto, Canada.

MacFarlane, A.G.J. (1969). Dual-system methods in dynamical analysis. Proc. IEE, Vol.116, No.8, pp.1458-1462.

MacFarlane, A.G.J., and B. Kouvaritakis (1977). A design technique for linear multivariable feedback systems. Int. J. Control, 25, pp.837-874.

Munro, N. (1969). Computer Aided Design of Multivariable Control Systems, Ph.D. Thesis, UMIST, Manchester, England.

Munro, N., and V. Zakian (1970). Inversion of rational polynomial matrices. Electron. Lett., Vol.6, No.19, pp.629-630.

Munro, N., and R.S. McLeod (1971). Minimal realisation of transfer-function matrices using the system matrix. Proc.IEE, Vol.118, No.9, pp.1298-1301.

Munro, N. (1972). Design of controllers for an open-loop unstable multivariable system using the inverse Nyquist array. Proc. IEE, Vol.119, No.9, pp.1377-1382.

Munro, N., and J.B. Bowland (1972). Transform techniques and automatic system simulation. U.K. Simulation Council Symposium, UMIST, Manchester, England.

Munro, N. (1974). Further results on pole-shifting using output feedback. Int. J. Control, Vol.20, No.5, pp.775-786.

Munro, N., and T.A.S. Ibrahim (1974). Computer-aided design of multivariable sampled-data systems. IEE Int. Conf. on Computer Aided Design, Southampton, England.

Munro, N., and A.I. Vardulakis (1974). Correspondence on computer-aided design procedure for reduced-order observers. Proc. IEE, Vol.121, No.4, pp.310-312.

Munro, N. (1977). Further results on pole-shifting using output feedback, Control System Design by Pole-Zero Assignment (ed. F. Fallside), Academic Press, London, pp.123-135.

Munro, N., and S. Novin-Hirbod (1979). Pole assignment using full-rank output-feedback compensators. To appear in Int.J. System Science.

Novin-Hirbod, S. (1978). Computer Aided Control System Synthesis, Ph.D. Thesis, UMIST, Manchester, England.

Rosenbrock, H.H. (1970). State-Space and Multivariable Theory. Nelson, London.

Rosenbrock, H.H. (1974). Computer-Aided Control System Design. Academic Press, London.

Strang, G. (1976). Linear Algebra and Its Applications, Academic Press, New York.

Wang, S.H. (1970). Design of precompensator for decoupling problem. Electron. Lett., Vol.6 (23), pp.739-741.

Wolovich, W.A. (1975). Output-feedback decoupling. IEEE Trans. on Auto. Control, Vol.AC20(1), pp.148-149.

Young, P.C.,and L.D. Levsen (1970). Linear systems design, Naval Weapons Centre Report NWC TP 4927, China Lake, California.

Zaman, M., and A.W.J. Griffin (1970). Transfer function from sampled impulse response. Measurement and Control, Vol.3, pp.T101-T108.

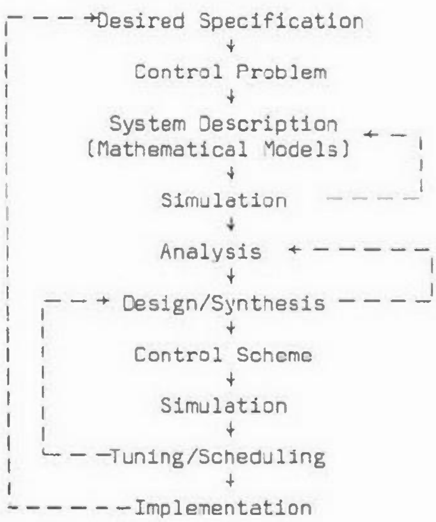

Fig. 1 : Nature of the control problem

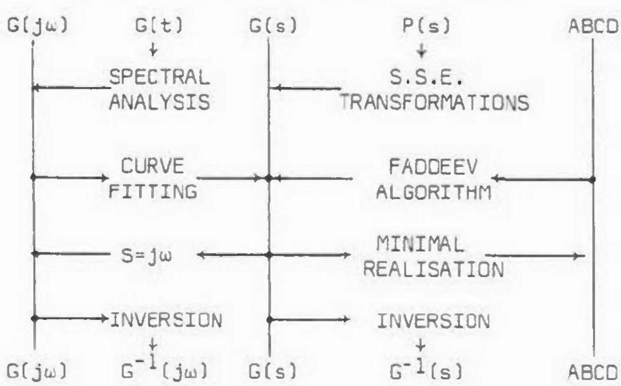

Fig. 3: Data manipulations

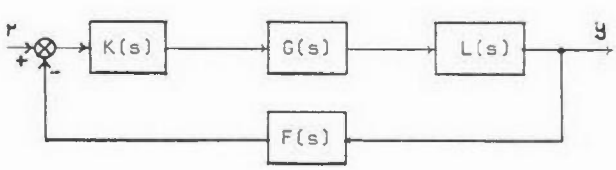

Fig. 4 : Feedback configuration

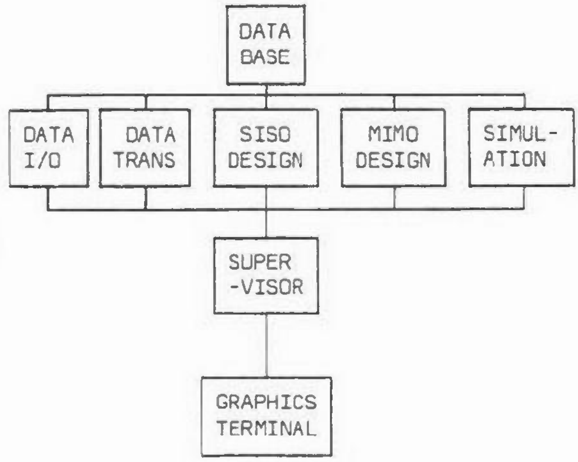

Fig. 2 : Structure of frequency-domain design suite

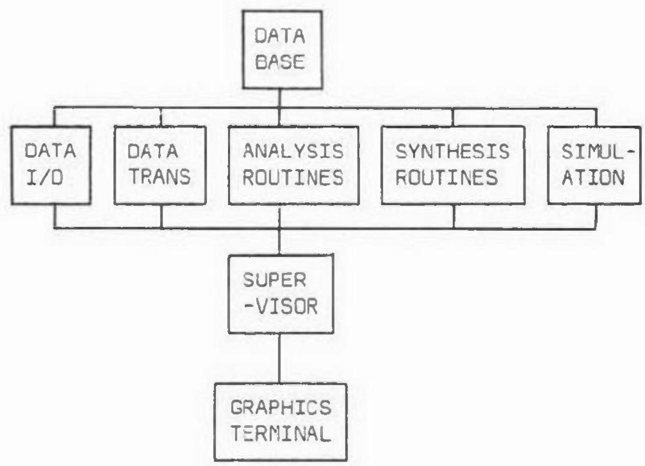

Fig. 5 : Structure of synthesis suite

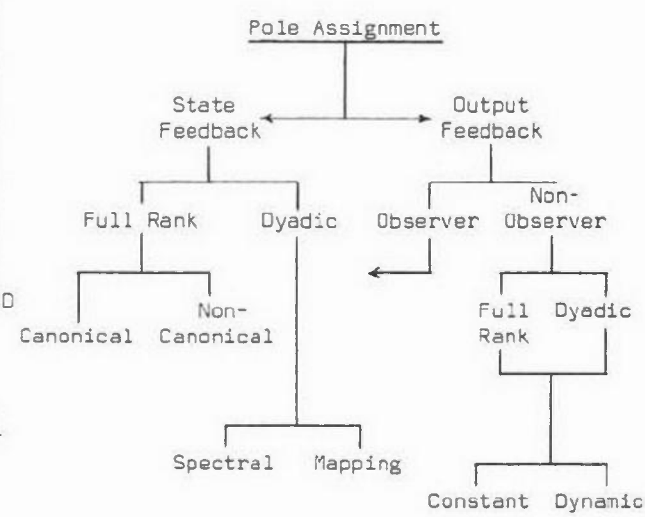

Fig. 6 : Pole assignment algorithms