

INTERACTIVE GRAPHICS FOR THE DESIGN OF FEEDBACK CONTROL SYSTEMS

J.M. Wells*, W.H. Lehn†, and W. Podaima‡

*Computer Science Department, University of Manitoba

†Electrical Engineering Department, University of Manitoba

‡National Research Council of Canada

Abstract

This paper describes a system which has been implemented on a DECLAB/40 in the Electrical Engineering Department of the University of Manitoba. The system, written in BASIC and adapted from programs written in FORTRAN for a batch environment by Melsa and Jones, produces a root locus plot for the transfer function of a feedback control system. This plot and the function are displayed (on a VR-14 Graphical Display) and the user can interact with the display, using a light pen and keyboard to modify the transfer function to meet his design criteria.

The program was completed in the spring of 1975. It has since been used by two undergraduate engineering classes. Experience with these classes is described. Proposed extensions to the program are also described. The paper concludes with some comments on the utility of a system like the DECLAB/40 in an Engineering education environment, and a short catalog of other projects.

GRAPHIQUES INTERACTIFS POUR LA CONCEPTION DE SYSTÈMES DE CONTRÔLE PAR RÉTROACTION

Résumé

La présente communication décrit un système qui a été mis en application sur un DECLAB/40, au Département de génie électrique de l'Université du Manitoba. Le système, qui est rédigé en BASIC, est une adaptation de programmes écrits en FORTRAN pour des travaux par lots par Melsa et Jones; il produit un tracé du lieu des racines de la fonction de transfert d'un système de contrôle par rétroaction. Ce tracé et la fonction sont affichés (sur une unité d'affichage VR-14) et l'utilisateur peut communiquer avec l'affichage à l'aide d'un photostyle et d'un clavier, afin de modifier la fonction de transfert selon ses critères de conception.

Depuis l'achèvement du programme au printemps de 1975, deux classes d'étudiants-ingénieurs de premier cycle en ont fait usage. La communication décrit les expériences réalisées avec ces classes, ainsi que des additions proposées pour le programme. En conclusion, on apporte certains commentaires sur l'utilité d'un système comme le DECLAB/40 dans un milieu de formation des ingénieurs, et l'on présente un court répertoire d'autres projets.

INTERACTIVE GRAPHICS FOR THE DESIGN OF FEEDBACK CONTROL SYSTEMS

J.M. Wells, W.H. Lehn, University of Manitoba.
W. Podaima, National Research Council of Canada.

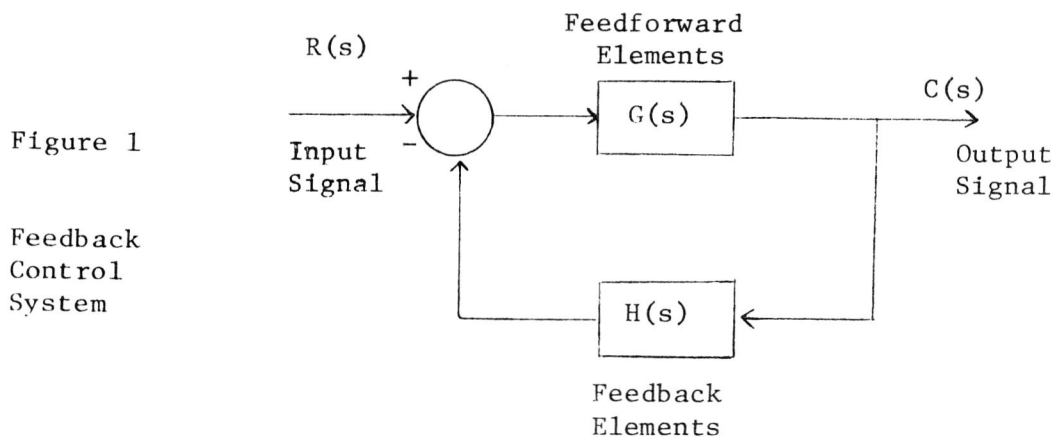
Introduction

The root locus method is a graphical technique for the analysis of feedback control system designs [1,2,3]. An experienced designer can use the method as a guide to synthesizing a control system which satisfies requirements for response and stability characteristics. The program described here produces root locus plots; the designer need do no computation and can adjust his system design in a few minutes' time to meet his design criteria.

The paper is in four parts. The first part is a brief discussion of control system design to place the root locus method in proper perspective. Second is a description of the hardware and software facilities on which the interactive graphic program is implemented. The third section is an example, with figures from the graphical display, of the steps in an interactive design session. Lastly, experience with undergraduate classes is described, and some general comments follow.

Theory

A feedback control system is a collection of physical components so connected that the output signal is fed back and subtracted from the input signal (see fig. 1).



$$\frac{C(s)}{R(s)} = \frac{G(s)}{1+G(s)H(s)} \quad \text{where } S \text{ is the Laplace variable}$$

The system is designed to accomplish a specified task, that is, for every input signal, a specified response is required. An example of such a system is the power steering system in an automobile: the signal $R(s)$ is the angular position of the steering wheel, and $C(s)$ is the angular position of the front wheels.

Feedback systems have important advantages over non-feedback systems: insensitivity to parameter variations and disturbance signals; high accuracy; and relatively easy physical implementation. However they present some design problems: the properties of the feedback system are significantly different from those of G and H taken separately. In particular, even though G and H are stable, the feedback system may be unstable. The root locus method was developed specifically to predict the behaviour of the feedback system given the elements G and H .

The elements are represented mathematically by linear differential equations with constant coefficients. The Laplace transformation is applied to the equations, and the ratio of output transform over input transform is formed. This ratio, whose numerator and denominator are polynomials in the Laplace variable s , is defined as the transfer function of the device. The transfer function of the feedback system in terms of element transfer functions $G(s)$ and $H(s)$ is $C(s)/R(s) = G(s)/(1+G(s)H(s))$.

The polynomials $G(s)$ and $H(s)$ are normally known in factored form; that is, the poles and zeros of these functions are known. In order to predict the performance and stability of the feedback system, the designer must know the poles and zeros of C/R . It is the poles of C/R (which are the roots of the equation $1+GH = 0$) that determine which complex exponential functions will appear in the transient response, $c(t)$.

The root locus of a system is the locus, or path, described in the complex plane by the roots of $1+GH = 0$, as some parameter (usually the gain constant is $G(s)$) is varied. As originally conceived, the root locus method was a graphical (pencil and paper) procedure designed to simplify finding polynomial roots, and to give the designer a picture of how the roots moved as the parameter was varied.

Automatic root locus plotting methods based on the use of analog computers have been devised [1], but these methods involve the usual problems of wiring, checking and calibrating an analog device. Melsa and Jones have written a collection of programs to deal with the computational aspects of control systems design [4]. One of these does a root locus plot. This collection of programs is written in FORTRAN and designed to be used in a batch processing environment. The work described here makes this computational tool available in an interactive environment and thus greatly increases its effectiveness.

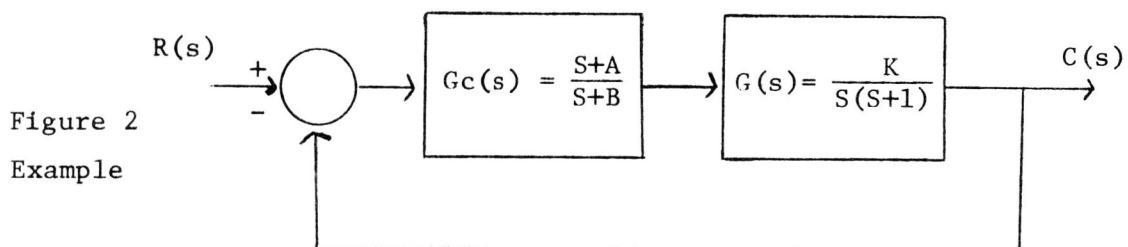
Hardware and Software

The DECLAB/40 is a package of hardware marketed by Digital Equipment Corporation. It includes a PDP-11/40 computer, VT-11 Graphic Display Subsystem and LPS-11 Lab Peripheral System. The configuration at the University of Manitoba also includes two two megabyte RK05 disks and a DEC writer terminal. This configuration of equipment suits the requirements of the Electrical Engineering Department very well. It can be easily incorporated into a wide variety of experimental engineering apparatus and used for measurement and control.

Software support for the DECLAB/40 is based on the single-user RT-11 Operating System. At the University of Manitoba assembler language (MACRO) and the BASIC language are used. Both include graphics support routines provided by DEC. The root locus plotting system described here was programmed entirely in BASIC. The program involves only 324 lines of BASIC code in two overlays.

Sample Design Session

Suppose that a feedback control system is to have response characteristics that require the root locations to be in the complex s-plane at $s = -3 \pm j5.2$. The problem is one of choosing appropriate values for A and B in the system shown in figure 2.



The user begins by entering initial co-efficients for the transfer function. These are typed on the console. Initial scale values for the root locus plot are also entered (see figure 3).

Figure 3
Keyboard Entry
of Coefficients

```

TYPE IN TERMS OF N(S) -
ORDER OF POLYNOMIAL TERM?1
COEFFICIENT( 1 )?0
COEFFICIENT( 2 )?1
ORDER OF POLYNOMIAL TERM?0

TYPE IN TERMS OF D(S) -
ORDER OF POLYNOMIAL TERM?1
COEFFICIENT( 1 )?0
COEFFICIENT( 2 )?1
ORDER OF POLYNOMIAL TERM?1
COEFFICIENT( 1 )?1
COEFFICIENT( 2 )?0
ORDER OF POLYNOMIAL TERM?1
COEFFICIENT( 1 )?1
COEFFICIENT( 2 )?1
ORDER OF POLYNOMIAL TERM?0

INITIAL VALUE FOR K?0

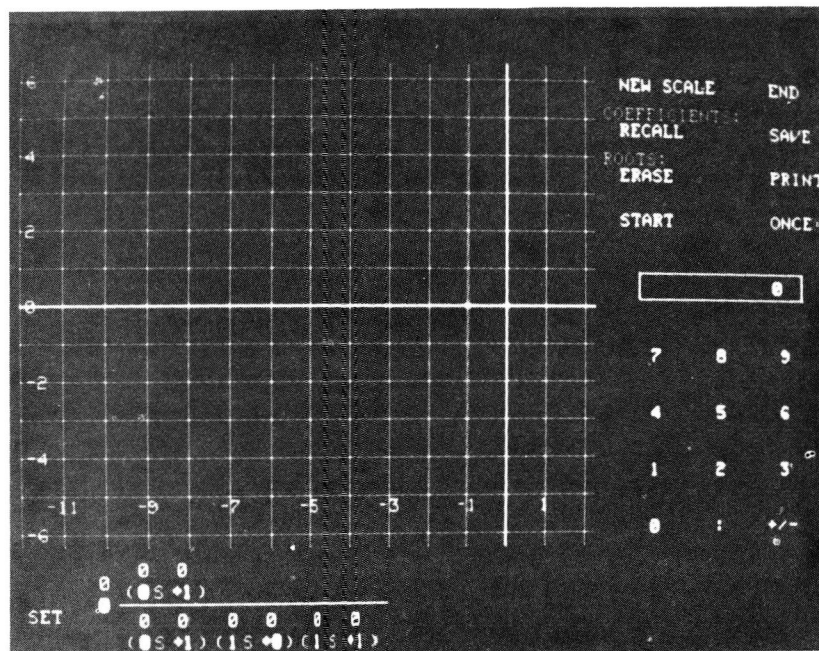
SCALE RANGE (REAL AXIS)?-12.2
SCALE RANGE (IMAGINARY AXIS)?-6.6

```

$N(s)$ here is the numerator polynomial in the product $G(s)H(s)$. $H(s)$ is 1 so $N(s)$ is the underlined portion of $K(S+A)/((S+B)(S)(S+1))$. $D(s)$ is the denominator in this product. The initial plots will be for $G_c(s) = 1$. After a pause of about thirty seconds, the initial display appears (see figure 4).

Figure 4

Initial Display



The graphical display is divided into four areas:

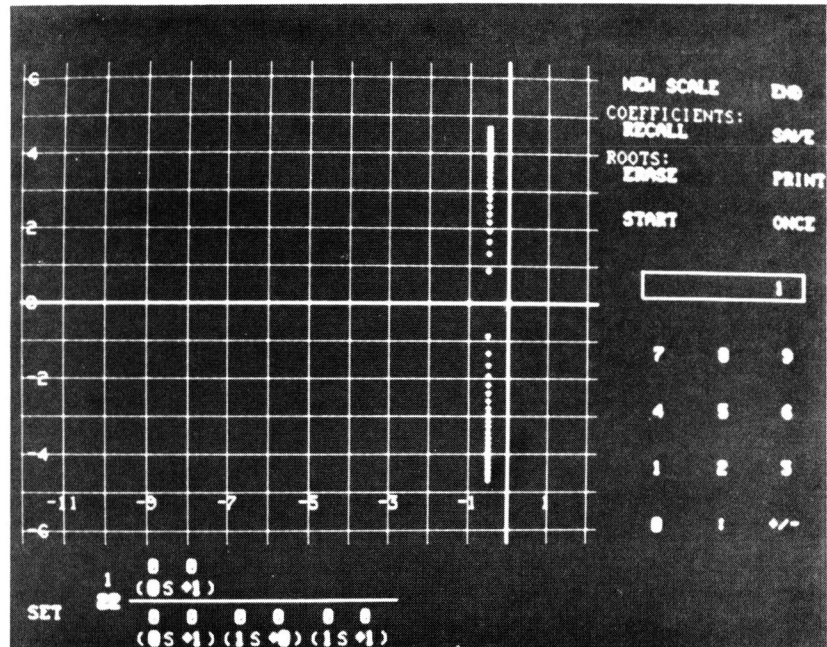
- (1) the root locus plot (left and centre of screen)
- (2) light-button commands (upper right)
- (3) keyboard (lower right)
- (4) transfer function (lower left).

The co-efficients of the transfer function are all light pen modifiable, as are the increments displayed adjacent to them. The program may be allowed to "run" (START), plotting roots as the co-efficients are all incremented, or it may be single stepped, (ONCE) calculating one new set of roots. Other commands will clear the plot (ERASE), print out the roots of the polynomial (PRINT), store (SAVE) and recall (RECALL) coefficients, permit rescaling of the plot (NEW SCALE) or terminate execution (END).

In figure 5 the display shows roots plotted for gain values (first co-efficient in the transfer function) running from 0 to 22 in steps of 1. New roots appear on the screen at about one second intervals. This dynamic presentation gives the user an excellent grasp of how the system stability varies as a function of changing co-efficient values.

Figure 5

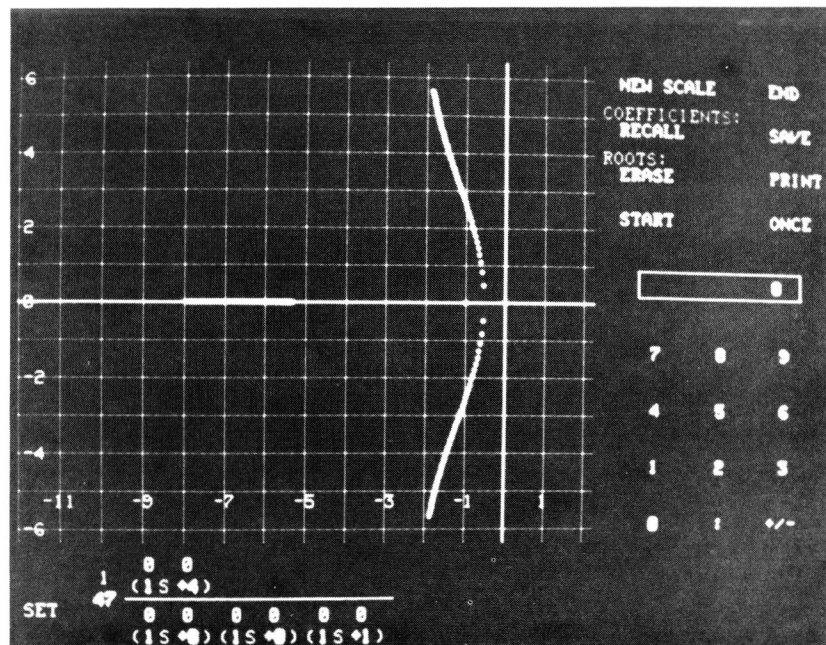
Roots plotted
for gain from
0 to 22



Touching the STOP command gives the user the opportunity to clear the display and vary any of the co-efficients in order to place the roots in any desired location. In this example, the roots do not pass through the desired location, so the user changes the co-efficients to try $G_c(s) = \frac{S+4}{S+8}$. Plotting the root locus for gain values from 0 to 47 he obtains the plot of figure 6.

Figure 6

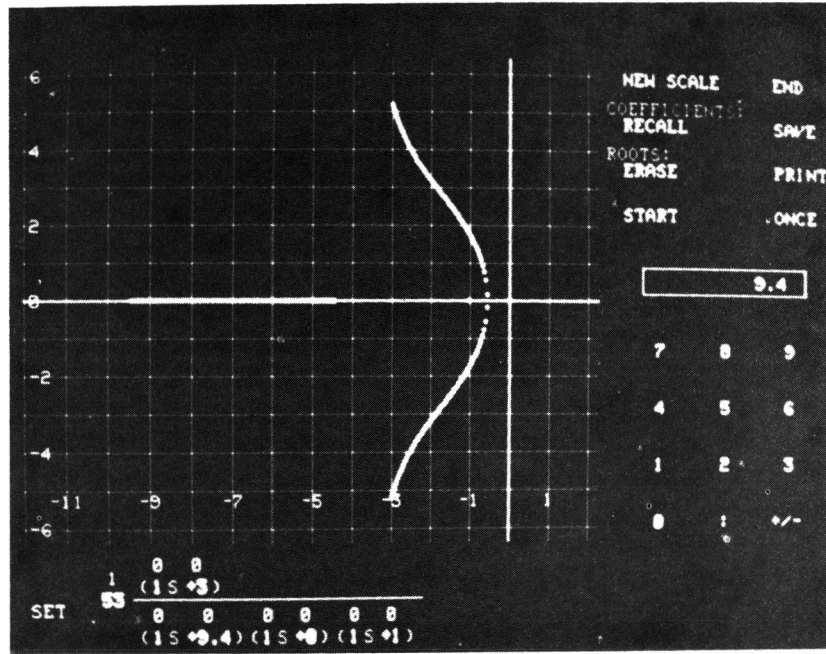
$$G_c(s) = \frac{S+4}{S+8}$$



A bit of experimentation with the co-efficients of G_c reveals how the locus varies as these co-efficients vary, and finally the user tries $G_c(s) = \frac{s+3}{s+9.4}$. The display is controlled so that it stops precisely when the roots are in the desired location (see figure 7).

Figure 7

$$G_c(s) = \frac{s+3}{s+9.4}$$



In response to the PRINT command, the root locations are printed on the console (figure 8).

Figure 8

POLYNOMIAL COEFFICIENTS

```

159
 62.4
 10.4
 1
    
```

Output via
PRINT command

ROOTS

```

-2.98966      -5.19897
-2.98966      5.19897
-4.42068      0
    
```


The value of K is read off the display. The user can proceed to check the system response in the time domain using another (recently completed) program (see figure 9).

THE GAIN COEFFICIENT?53

THE ORDER OF THE NUMERATOR(# OF ZEROES)?1
ZERO(1)?-3,0

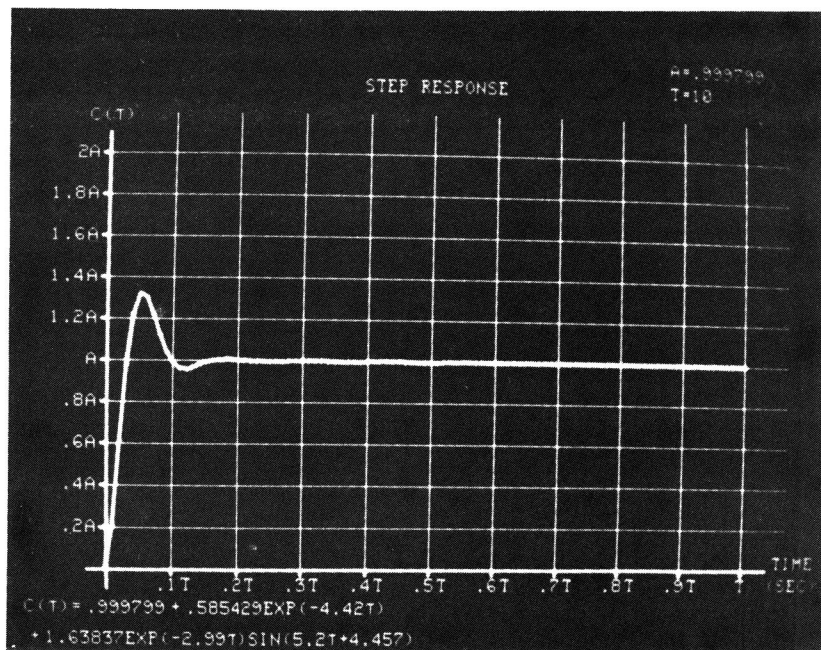
THE ORDER OF THE DENOMINATOR(# OF POLES)?3
POLE(1)?-4.42,0
POLE(2)?-2.99,5.2
POLE(3)?-2.99,-5.2

THE VERTICAL AXIS IS DEFAULTED AT A MAXIMUM
AMPLITUDE OF TWICE THE STEADY STATE VALUE
TO SCALE IT TYPE 1, IF NOT TYPE 0?0

THE TIME AXIS IS DEFAULTED AT T=0 TO 10 SECONDS
TO SCALE IT TYPE 1, IF NOT TYPE 0?0

Figure 9

Time Domain
Response to
Step Function



Experience with Classes

The root locus example described above corresponds almost exactly to a laboratory experiment which has been carried out by the senior-year electrical engineering class for the last two years. Before the lab is assigned, the students have received theoretical instruction in the root locus method, as well as a few problems to be solved with pencil and paper. They have not however received instruction on the design of compensating networks ($G_c(s)$).

In preparation for the lab, the class is given about 30 minutes' discussion, wherein the problem to be solved, as well as the root locus program, is described. Then each student is assigned 30 minutes of computer time, in which he must locate the pole and zero of the compensator by trial and error.

This minimal amount of formal discussion and computation time is sufficient to permit intelligent execution of the experiment. The students learn by direct interaction how root locus plots behave when parameters are changed. The proper use of the various program features is so self-evident and simple that the program is immediately effective for control system design.

Conclusions

The interactive graphic tool for feedback control system design which is described here is very effective and relatively cheap. Careful attention to the man-machine interface means that it can be used after a few minutes instruction. The software environment provided by DEC made system creation and modification a relatively easy task; this is important in graphics because one has a very poor fix on the desirable user interface until something is actually running.

More generally, an integrated system including graphic capability and analogue interfaces, all supported by high-level software, is a very nice one for an electrical engineering education environment. About two years have now elapsed since the system was installed and it is very intensively used. A short list of some projects currently underway follows.

Other Projects

Automatic testing of impedance relays used in power system protection (G.W. Swift).

Extraction of electrocardiograph signals out of background muscle-signal noise by digital signal processing (E. Shwedyk).

Design, simulation and response measurement of wave digital filters (G.O. Martens).

Graphical presentation of element grids as data preparation for finite-element analysis (A. Wexler).

Simulation of image transmission through a refracting atmosphere using graphic display (W. Lehn).

Direct digital control, multiplexed to serve several independent systems simultaneously (W. Lehn).

Acknowledgements

The original idea for this project came from Professor Glen W. Swift whose assistance is gratefully acknowledged. Financial assistance was provided by NRC grant number A7648.

References

- [1] D'Azzo, John J., and Houpis, Constantine H., Feedback Control System Analysis and Synthesis. McGraw-Hill, New York, 1960.
- [2] Dorf, R.C., Modern Control Systems. Second edition, Addison-Wesley, Reading, Mass., 1974.
- [3] Ogata, K., Modern Control Engineering. Prentice-Hall, Englewood Cliffs, N.J., 1970.
- [4] Melsa, James L., and Jones, Stephen K., Computer Programs for Computational Assistance in the Study of Linear Control Theory. Second edition, McGraw-Hill, New York, 1973.