

IAC CONTROL SYSTEM ANALYSIS DEVELOPMENT

HAROLD P. FRISCH

GODDARD SPACE FLIGHT CENTER  
GREENBELT, MARYLAND

LARGE SPACE SYSTEMS TECHNOLOGY--1981  
THIRD ANNUAL TECHNICAL REVIEW  
LANGLEY RESEARCH CENTER  
HAMPTON, VA 23665  
NOVEMBER 16-19, 1981

## MODEL (MULTIOPTIMAL DIFFERENTIAL EQUATION LANGUAGE) PROGRAM

The MultiOptimal Differential Equation Language provides a means for generating numerical solutions to systems of differential equations using a digital computer. The notation of this language is similar to that usually used in describing physical systems by differential equations. Thus, the learning process is simplified, programming becomes easier, and debugging is more readily accomplished. Programs written in the MultiOptimal Differential Equation Language are machine translated into FORTRAN 4 code which is optimal in several respects.

The interactive version of MODEL makes use of interactive system routines so that the user may observe the solution as it is being generated and interact with the program in a manner similar to that associated with analog simulation.

The DISCOS-Control version of the MODEL simulation language is used to describe the control system for a plant which is modelled by the DISCOS program. DISCOS plant variables may be referenced in the control system description and all the sensor signals, coupling torques (e.g., motors), momentum devices (e.g., flywheels) and external loads (jets, magnetics, etc.) are automatically linked to the DISCOS plant model.

All MODEL development work is being done at NASA GSFC by Benjamin G. Zimmerman. This in-house work is proceeding on schedule; however, associated documentation is not yet ready for general release.

## A FAMILY OF PROGRAMS WHICH OUTPUT OPTIMALLY WRITTEN FORTRAN PROGRAMS AND SUBROUTINES FOR DYNAMICS ANALYSIS

INPUT: NON-LINEAR ORDINARY DIFFERENTIAL EQUATIONS (ODE'S) IN STANDARD MATHEMATICAL NOTATION

--- FOR EXAMPLE ---

DERIVED EQUATIONS

$$\ddot{y} = A \dot{y} + D x^3$$

$$\dot{x} = E \tan(x\dot{y})$$

INPUT TO MODEL

$$Y''' = A*Y' + D*X**3$$

$$X' = E*TAN(X*Y')$$

END

## THE MODEL FAMILY OF PROGRAMS

MODEL is not a new program at NASA/GSFC; various versions of it have been in use since 1971. These versions however interfaced only with hardware used by the Guidance and Control Branch at GSFC. The IAC program has provided the impetus to not only rewrite the program in a more machine portable form but to also create a family of MODEL programs. This family of programs will significantly expand analysis capabilities via use of a simulation language program.

MODEL-I is in effect the MODEL program that has been used for project support work by the G&C Branch since 1971. It is now considered to be fully checked out and bug free.

MODEL-C is a response to users that desire a more user friendly interface between the equations which define control system dynamics and the DISCOS program. It is now possible to use MODEL-I for standalone controls analysis and then to use the MODEL-I input file as input to MODEL-C to obtain subroutine CONTRL for a coupled controls structure interaction analysis via DISCOS.

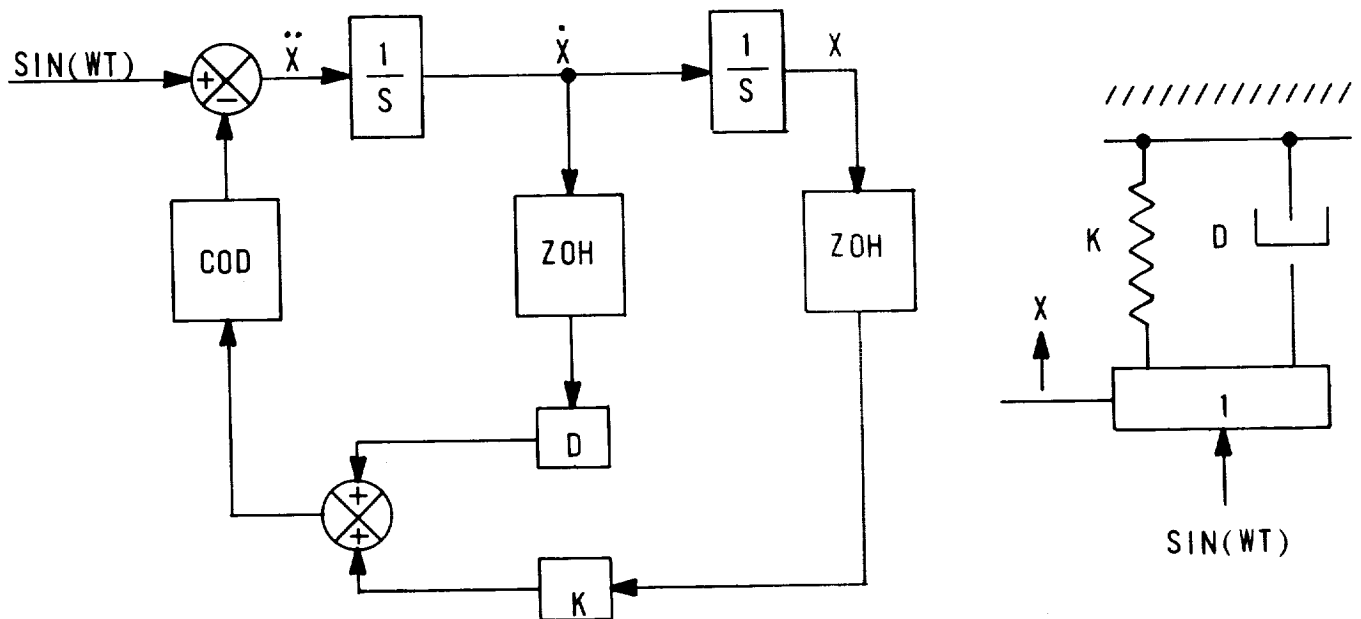
MODEL-S is nearing the final completion stage. It is designed to allow the user to readily input sampled data control equations. While MODEL-I can and has been used for sampled data controls analysis, users had to set up zero order holds and digital computational delays as logic functions. With MODEL-S, users will be capable of defining sampled data controllers in a much more user friendly manner.

MODEL-I	INPUT: NON-LINEAR TIME DOMAIN ODE'S
	OUTPUT: FORTRAN PROGRAM FOR "TRUE" INTERACTIVE TIME DOMAIN ANALYSIS
MODEL-C	INPUT: NON-LINEAR ODE'S WHICH DEFINE CONTROL SYSTEM DYNAMICS FOR THE DISCOS PROGRAM
	OUTPUT: FORTRAN SUBROUTINE COMPATIBLE WITH DISCOS CONTROL SYSTEM DEFINITION NEEDS
MODEL-S	INPUT: NON-LINEAR SAMPLED DATA CONTROL SYSTEM EQUATIONS; NON-LINEAR ODE'S, ZERO ORDER HOLDS AND DIGITAL COMPUTATIONAL DELAYS
	OUTPUT: FORTRAN PROGRAM FOR "TRUE" INTERACTIVE TIME DOMAIN ANALYSIS

## TRIVIAL DEMONSTRATION PROBLEM

The admittedly trivial demonstration problem is presented to dramatize the simplicity of setting up the input data required by any of the MODEL family of programs. Line 1 defines which of the MODEL family is to be used, on line 2 the equation which defines the system is written and on line 3 an END statement is provided.

Users need not specify input parameter values for K, D, W or any initial condition. MODEL will provide default values for all. The user is free to override them when input data is read into the MODEL written program. Furthermore if no output selection commands are given, MODEL outputs all time varying state variables to the lineprinter. "True" interactive plot and analysis capabilities are available by insertion of one more line of input data.



ZOH ZERO ORDER HOLD  
 COD (DIGITAL) COMPUTATIONAL DELAY

### MODEL PROGRAM INPUT

MODEL-I // [CONTINUOUS SYSTEM]

$$X'' = \text{SIN}(W*T) - K*X - D*X'$$

END

MODEL-S // [SAMPLED DATA SYSTEM]

$$X'' = \text{SIN}(W*T) - \text{COD}(K*\text{ZOH}(X) + D*\text{ZOH}(X'))$$

END

## DISCOS

### A DIGITAL PROGRAM FOR THE DYNAMIC INTERACTION SIMULATION OF CONTROL AND STRUCTURE

The prime emphasis with respect to DISCOS (ref 1) over the past year has been in responding to the desires of several users for a more user friendly input data interface.

It is currently possible to enter all standard DISCOS input data via keyboard response to tutorial prompts, generated by the DISCOS tutorial input program written by Joan Sanborn at GSFC.

With respect to the "user supplied subroutines" for DISCOS, it has been demonstrated at GSFC that this need can to a major extent be eliminated. The breadboard demonstration program which proved this ability is currently being incorporated into DISCOS.

In the process of converting DISCOS from the IBM computer to the VAX, a documentation program was written. ALL DISCOS program documentation is now contained within the source code. The documentation program is capable of recognizing delimiters within the source code and generating the full DISCOS REFERENCE GUIDE. Up to date documentation is now always available to DISCOS users via a rerun of the documentation program.

#### \*\*\*\* NEW CAPABILITIES \*\*\*\*

TUTORIAL PROGRAM TO SET UP INPUT DATA

DISCOS TO MODEL-C INTERFACE

DISCOS/VAX CREATED FROM DISCOS/IBM (COSMIC STANDARD)

PROGRAM TO WRITE DISCOS REFERENCE GUIDE

## SAMSAN

Controls analysts have access to a vast array of published algorithms to solve an equally large spectrum of controls related computational problems. Unfortunately the subroutines which implement the algorithms do not always readily interface with each other, and furthermore they are frequently less applicable than their creators imply in accompanying documentation (ref 2).

SAMSAN is an attempt to collect together in one self-consistent library a complete set of computational algorithms which are applicable for large order classical controls analysis. The SAMSAN library is biased toward numerical accuracy; computational speed is important but accuracy is given higher priority. An attempt is made to include only those algorithms which are backed up by rigorous numerical error analysis and are applicable for large order control system analysis, i.e., order less than about 200.

The limit of about 200 is arrived at from practical computational speed considerations and more importantly from the fact that most iterative algorithms associated with non-symmetric matrices yield results which are exact for some matrix  $A+E$  where  $E$  is viewed as error in knowledge of  $A$ . Furthermore if the associated computing problem is well-conditioned the norm of  $E$  is usually given as:

$$\text{NORM}(E) < K * N^{P-B} * \text{NORM}(A)$$

WHERE

K is about equal to 10.0  
N is the order of the non-symmetric matrix "A"  
P is about equal to 2.0  
B is about equal to 52 (DOUBLE PRECISION ON IBM)

In application when  $N$  exceeds about 200 and "A" contains a fair spread in numeric magnitudes, the norm of  $E$  gets disturbingly large.

## A COLLECTION OF NUMERICALLY SUPERIOR ALGORITHMS FOR SAMPLED DATA CONTROL SYSTEM ANALYSIS

WHY?

A SELF-CONSISTENT SET OF NUMERICALLY RELIABLE ALGORITHMS  
FOR LARGE ORDER CONTROL SYSTEM ANALYSIS DOES NOT EXIST

MANY STANDARD CONTROLS ANALYSIS METHODS ARE BASED UPON THE  
NEED TO SET UP AND SOLVE AN ILL-CONDITIONED COMPUTING PROBLEM

## ILL-CONDITIONED COMPUTING PROBLEMS

The most common function appearing in classical controls analysis literature is the polynomial in power series form. Unfortunately if one desires to analyze large order control system problems via use of polynomials in power series form, one will almost assuredly encounter numerical computation problems. This fact is one of the most important results of numerical error analysis. It is also one of the most consistently ignored facts in frequency domain analysis.

The symmetric matrices arrived at in structures analysis are guaranteed to have a linearly independent eigenvector for every eigenvalue. This fact leads to the ability to generate reduced order dynamic models via modal analysis.

The non-symmetric matrices of controls and structures-controls analysis do not necessarily and usually do not have a full set of linearly independent eigenvectors. This fact prohibits a complete decoupling of the system equations via eigen analysis. Theoretically the best that can be done is to almost decouple the equations by reducing the associated matrix to Jordan canonical form. This cannot be done using floating point arithmetic. Furthermore if one unsuspectingly requests all eigenvectors for a matrix which does not have a full set, the net result will be either a singular or a near singular eigenvector matrix and hence one useless for modal analysis.

## MODERATE DEGREE POLYNOMIALS IN POWER SERIES FORMAT, (N .GE. 15)

$$A_N S^N + \dots + A_2 S^2 + A_1 S + A_0$$

### COMPUTATION OF POLYNOMIAL ROOTS

### EVALUATION OF POLYNOMIAL NEAR A ROOT

## NON-SYMMETRIC MATRICES

### INVERSION OF NEAR SINGULAR EIGENVECTOR MATRICES

### REDUCTION OF MATRIX TO JORDAN CANONICAL FORM

## SAMSAN

### \*\*\*\* NEW CAPABILITIES \*\*\*\*

The most common origin of high degree polynomials in controls analysis is from the expansion of determinants of polynomials in either the Laplace variable "S" or the Z-transform variable "Z". Brute force expansion and combination of like power terms leads naturally to power series polynomials.

SAMSAN has an algorithm which will determine all eigenvalues of a matrix of low to moderate degree polynomials. By using this algorithm it is possible to obtain polynomials associated with determinantal expressions in factor polynomial form, a more desirable form from the standpoint of numerical analysis (ref 3).

SAMSAN contains the algorithms associated with the program BLOCK IT distributed by COSMIC. These algorithms enable one to reduce a non-symmetric matrix to block diagonal form via a real non-singular transformation matrix. The net result is that SAMSAN users have the ability to almost decouple a full set of system equations and obtain reduced order models via a quasi-modal analysis method.

Frequency domain analysis methods have been a mainstay of controls design methods for the past several decades. Surprisingly, algorithms for generating frequency response data are not widely distributed.

SAMSAN users have a full frequency response computation capability. Frequency response in the "S", "Z" or "W" domain is available via algorithms with advanced stepping logic.

COMPUTE ALL EIGENVALUES OF A MATRIX OF LOW DEGREE POLYNOMIALS

COMPUTE THE REAL NON-SINGULAR TRANSFORMATION MATRIX WHICH WILL  
REDUCE A REAL NON-SYMMETRIC MATRIX TO BLOCK DIAGONAL FORM

FOR TRANSFER FUNCTIONS DEFINED AS POLYNOMIAL RATIOS IN EITHER  
THE "S", OR "Z" DOMAIN, COMPUTE BODE, NICHOLS, NYQUIST & ROOT  
LOCUS FREQUENCY RESPONSE DATA IN EITHER "S", "Z" OR "W" DOMAIN



## REFERENCES

1. Bodley, C.S., Devers, A.D., Park, A.C., and Frisch, H.P., "A Digital Computer Program for the Dynamic Interaction Simulation of Controls and Structure (DISCOS)." NASA Technical Paper 1219, Vols 1 and 2, May 1978.
2. Frisch, H.P., "Pitfalls and Guidelines for the Numerical Evaluation of Moderate-Order System Frequency Response." NASA Technical Paper 1814, June 1981.
3. Frisch, H.P., "Reduced Order Feedback Control Equations for Linear Time and Frequency Domain Analysis." NASA Technical Paper 1818, June 1981.

