1620 USERS GROUP

WESTERN REGION

MINUTES OF THE MEETING

DECEMBER 11-13, 1963

TEMPE, ARIZONA

ROBERT R. WHITE

WESTERN REGION SECRETARY

CONTENTS

1620 USERS GROUP

PRESIDENT
J. L. DAVIDSON
Long Island Lighting Co.
175 Old Country Road
Hicksville, New York

SECRETARY-TREASURER
MARLENE T. METZNER
Pratt & Whitney Aircraft Co.
Fla. Research & Dev. Center
West Palm Beach, Florida

WESTERN REGION PRESIDENT
ROBERT EBERT
Spectrol Electronics Corp.
1704 S. Del Mar Avenue
San Gabriel, California

MID-WESTERN REGION PRESIDENT
W. A. BURROWS
Dravo Corporation
Neville Island
Pittsburgh 25, Pennsylvania

EASTERN REGION PRESIDENT
J. R. OLIVER
Univ. of Southwestern La.
Box 133, USL Station
LaFayette, Louisiana

CANADIAN REGION PRESIDENT
D. A. JARDINE
Dupont of Canada
Research Center
Kingston, Ontario

EUROPEAN REGION PRESIDENT
H. TOMPA
European Research Associates
95 Rue Gatti De Gamond
Bruxelles 18, Belgium

December 26, 1963

To the Members of the 1620 Users Group:

It is with sincere regret that I must announce my resignation, effective during this meeting, as President of the Western Region. The executive council of the group, acting in accord with the by-laws, has appointed Paul Bickford to serve the remainder of the current term.

Paul's appointment was made upon my recommendation, and I feel that his experience with the group will make him eminently qualified to continue its growth and activity.

In turning the administration of the Western Region over to Paul, I can only say that I shall sincerely miss working with all of you, I express my thanks and appreciation to those of you, too numerous to mention individually, who have worked to make the meetings which I have conducted successful, and I extend my best wishes to the officers and members of the 1620 Users Group for the continued growth and success which you have enjoyed over the last four years.

Sincerely,

Robert A. Ebert

1620 USERS GROUP MEETING
TEMPE, ARIZONA
DECEMBER 11,12,13, 1963
ROSTER OF ATTENDEES

USERS GROUP NO.	INSTALLATION REPRESENTATIVE
1118	NANCY PAQUIN U.S. PUBLIC HEALTH SERVICE ROCKVILLE, MARYLAND
1118	G. ROBERT ORNDORFF U.S. PUBLIC HEALTH SERVICE ROCKVILLE, MARYLAND
1216	CARLIS TAYLOR A F R R I BETHESDA, MARYLAND
1302	T. R. HOFFMAN UNION COLLEGE SCHENECTADY, N. Y.
1334	DR. REIMUT WETTE M.D. ANDERSON HOSPITAL AND TUMOR INST. HOUSTON, TEXAS
1346	ELIAS C. TONIAS ERDMAN AND ANTHONY - CONS. ENGRS. ROCHESTER, NEW YORK
3082	PAUL BICKFORD OKLAHOMA UNIV. MED. RESEARCH COMP. CTR. OKLAHOMA CITY, OKLA.
3082	CARA MITCHELL OKLAHOMA UNIV. MED. RESEARCH COMP. CTR. OKLAHOMA CITY, OKLA.
3166	DR. HERMAN B. WEISSMAN UNIVERSITY OF ILLINOIS CHICAGO, ILLINOIS
3175	HELEN LIGON BAYLOR UNIVERSITY WACO, TEXAS
3182	ROBERT G. LANGE AUTOMATIC ELECTRIC LABORATORIES, INC. NORTHLAKE, ILLINOIS
3240	MELFORD E. MONSEES U.S. ARMY ENGR. DISTRICT KANSAS CITY, MISSOURI
3261	GREGORY J. SHANAHAN CECO STEEL PRODUCTS CORP. CICERO, ILLINOIS

USERS GROUP NO.	INSTALLATION REPRESENTATIVE
3273	ROBERT C. BABIONE ACIC ST. LOUIS, MISSOURI
3273	CHARLES WEISS USAF AERONAUTICAL CHART AND INFO CENTER ST. LOUIS, MISSOURI
5001	R. C. WENRICK ACF INDUSTRIES, INC. ALBUQUERQUE, NEW MEXICO
5001	G. J. REED ACF INDUSTRIES, INC. ALBUQUERQUE, NEW MEXICO
5005	R. C. WEAVER BEAR CREEK MINING CO. SALT LAKE CITY, UTAH
5014	WALTER DAVIS GENERAL DYNAMICS/ASTRONAUTICS SAN DIEGO, CALIFORNIA
5016	RICHARD W. PUGSLEY COMPUTERMAT INC LOS ANGELES, CALIFORNIA
5019	EDGAR M. BLIZZARD JET PROPULSION LAB PASADENA, CALIFORNIA
5020	MARILYN DOIG COLORADO STATE UNIVERSITY FORT COLLINS, COLORADO
5021	J. W. LAFON US ARMY ENGINEER DISTRICT ALBUQUERQUE ALBUQUERQUE, NEW MEXICO
5027	MARVIN RUBINSTEIN ELECTRO OPTICAL SYSTEMS, INC. PASADENA, CALIFORNIA
5032	BOB MANNING GOODYEAR AEROSPACE LITCHFIELD PARK, ARIZONA
5032	N. A. KUFFEL GOODYEAR AEROSPACE LITCHFIELD PARK, ARIZONA
5032	J. MOSS GOODYEAR AEROSPACE LITCHFIELD PARK, ARIZONA

USERS GROUP NO.	INSTALLATION REPRESENTATIVE
5032	DAVID H. O'HERREN GOODYEAR AEROSPACE CO LITCHFIELD PARK ARIZ
5041	DAVID KEY MOTOROLA PHOENIX, ARIZONA
5045	W. WILCOXSON U.S. NAVAL CIVIL ENGR. LAB. PORT HUENEME, CALIFORNIA
5053	ROBERT L. SHUTT SACRAMENTO PEAK OBSERVATORY SUNSPOT, NEW MEXICO
5053	FRANK BIRD SACRAMENTO PEAK OBSERVATORY SUNSPOT, NEW MEXICO
5057	DAVID G. KITZINGER SANDIA CORP ALBUQUERQUE, NEW MEXICO
5057	ELIZABETH L. FROST SANDIA CORP. ALBUQUERQUE, NEW MEXICO
5058	BOB BABCOCK SUNDSTRAND AVIATION DENVER DENVER, COLORADO
5064	ALLEN L. GRAVITT SIGNAL OIL AND GAS CO. LOS ANGELES, CALIFORNIA
5065	ROBERT W. WILLSON SALT RIVER PROJECT PHOENIX, ARIZONA
5065	ERNEST NICHOLS SALT RIVER PROJECT PHOENIX, ARIZONA
5065	MAX A. HAYES SALT RIVER PROJECT PHOENIX, ARIZONA
5078	DR. MORRIS J. GARBER UNIV. OF CALIFORNIA AT RIVERSIDE RIVERSIDE, CALIFORNIA
5078	THOMAS M. LITTLE UNIVERSITY OF CALIFORNIA AT RIVERSIDE RIVERSIDE, CALIFORNIA

USERS GROUP NO.	INSTALLATION REPRESENTATIVE
5079	SAMUEL K. PRINGLE MAGNOLIA PIPE LINE CO. DALLAS. TEXAS
5084	E. B. LOOP UNION OIL CO OF CALIF RODEO. CALIFORNIA
5085	ROBERT D. MOFFITT U.S. ARMY ENGR. DIVISION. N.P. PORTLAND. OREGON
5089	DONALD J. MARTIN U.S. PUBLIC HEALTH SERVICE LAS VEGAS. NEVADA
5089	DAVID L. BAER U.S. PUBLIC HEALTH SERVICE LAS VEGAS. NEVADA
5095	HARRY D. RENICK WEYERHAEUSER CO TACOMA. WASHINGTON
5096	BOYD C. NORRIS U.S. BUREAU OF RECLAMATION SACRAMENTO. CALIFORNIA
5096	R. BRUCE BROWNRIGG U.S. BUREAU OF RECLAMATION SACRAMENTO. CALIFORNIA
5099	BEVERLY DOIG W S M R WHITE SANDS. NEW MEXICO
5104	JOHN R. GUINN TEXAS COLLEGE OF ARTS + INDUSTRIES KINGSVILLE. TEXAS
5120	WILLIAM L. REUTER SO. DAKOTA SCHOOL OF MINES AND TECH. RAPID CITY. SOUTH DAKOTA
5125	CHIN MO LEE UTAH POWER AND LIGHT CO. SALT LAKE CITY. UTAH
5133	JOSE RAMIREZ MASON AND HANGER - SILAS MASON CO.. INC AMARILLO. TEXAS
5139	RICHARD A. HARRIS NORTH TEXAS STATE UNIVERSITY DENTON. TEXAS

USERS GROUP NO.	INSTALLATION REPRESENTATIVE
5144	JAMES J. STANLEY U S WEATHER BUREAU RFC SACRAMENTO, CALIFORNIA
5146	BILL WOLLENHAUPT GOLDSTONE TRACKING STATION BARSTOW, CALIFORNIA
5146	JOE SIBLEY GOLDSTONE TRACKING STATION BARSTOW, CALIFORNIA
5150	SAM THOMPSON HALLIBURTON CO DUNCAN, OKLAHOMA
5150	GEORGE A. LARCADE HALLIBURTON CO DUNCAN, OKLAHOMA
5158	A. GARLAND NECHES BUTANE PRODUCTS CO. PORT NECHES, TEXAS
5165	CHARLES R. HEBBLE CORPS OF ENGINEERS WALLA WALLA, WASHINGTON
5165	CECIL L. ASHLEY U S ARMY CORPS OF ENGINEER WALLA WALLA WASH
5179	L. E. HARVEY FOOTHILL COLLEGE LOS ALTOS HILLS, CALIFORNIA
5181	ROBERT R. WHITE LOS ANGELES DEPT. OF WATER AND POWER LOS ANGELES, CALIFORNIA
5183	JOSEPH P. SNOW UNIVERSITY OF WYOMING LARAMIE, WYOMING
5190	WILLIAM G. LANE CHICO STATE COLLEGE CHICO, CALIFORNIA
5195	S. V. BURKS, JR. PITTSBURGH PLATE GLASS CO. - CHEM. DIV. CORPUS CHRISTI, TEXAS
5199	CHARLES S. WALKER SCHOOL OF ENGINEERING TEMPE, ARIZONA

USERS GROUP NO.	INSTALLATION REPRESENTATIVE
5210	ANNA L. NIEHUES PALO ALTO UNIFIED SCHOOL DISTRICT PALO ALTO, CALIFORNIA
5210	ARLINE K. KAPPHAHN PALO ALTO UNIFIED SCHOOL DISTRICT PALO ALTO, CALIFORNIA
5211	KENNETH KRIEGE CALIFORNIA STATE POLYTECHNIC COLLEGE POMONA, CALIFORNIA
5215	ROSEMARY PETERSEN UCLA - WESTERN DATA PROCESSING CENTER LOS ANGELES, CALIFORNIA
7007	A. C. R. NEWBERRY UNIVERSITY OF ALBERTA CALGARY ALBERTA, CANADA
	SETH P. EVANS PHOENIX COLLEGE PHOENIX, ARIZONA
	WARREN BUXTON PHOENIX COLLEGE PHOENIX, ARIZONA
	JOHN P. MCCALLISTER U S WEATHER BUREAU FT. WORTH, TEXAS
IBM	W. H. DUKELOW I B M KANSAS CITY, MISSOURI
IBM	CHARLES E. BERRY IBM - WESTERN REGION OFFICE LOS ANGELES, CALIFORNIA
IBM	ROBERT A. EBERT IBM - WESTERN DATA PROCESSING CENTER LOS ANGELES, CALIFORNIA
IBM	JAMES E. MORGAN IBM - WESTERN REGION OFFICE LOS ANGELES, CALIFORNIA
IBM	BRUCE J. SOCKS IBM CORP. CHICAGO, ILLINOIS
IBM	ANGELO ARENA I B M WHITE PLAINS, N.Y.

USERS GROUP NO.	INSTALLATION REPRESENTATIVE
IBM	GERALD R. HOGSETT IBM - WESTERN REGION OFFICE LOS ANGELES. CALIFORNIA

WESTERN REGION 1620 USERS GROUP
MINUTES OF THE SEVENTH MEETING

The group assembled on the campus of Arizona State University in Tempe, Arizona, December 11, 12, and 13, 1963.

All papers listed in the agenda were presented as scheduled except paper A-1, Generalized SPS Routines for Handling Simple Problems, by Thomas L. Yates, which was not given, paper A-2, Expotential and Sinusoidal Curve Fitting, by E. P. Hilar, which was given at Technical Session "G", and paper G-1, Ray Trace Program for a General Lens System, by D. H. O'Herren, which was given at Technical Session "A". Paper E-1, Network Analysis, was presented by Mr. Gerald Hogsett, IBM, Los Angeles. Paper C-2 was read by Bob Ebert.

Copies of the abstracts and papers presented are enclosed with the exception of Paper G-1 which will be included with minutes of a later meeting.

The general meeting was opened on December 11 by President Bob Ebert and after opening remarks was turned over to Jim Morgan of IBM who introduced the other IBM representatives to the group. The sound-off session was moved up to this time to fill in the time for paper A-1. A summary of questions raised and answers to them follows the minutes of the business meeting.

Business Meeting

The business meeting was opened by Bob Ebert on December 12 and under old business, the request was made from the floor that the Roster of Members be arranged alphabetically in order of the name of the installation. The request was referred to Angelo Arena of IBM.

The first item of new business was the resignation of Bob Ebert as president of the Western Region 1620 Users Group. This was necessary under the by-laws as he is no longer connected with a 1620 installation. A letter by Bob to the Users Group is included with the minutes of the meeting.

The executive council, acting under the provisions of the by-laws, appointed Paul Bickford, Western Region Secretary, to complete the unexpired term of office of President, and he appointed Robert R. White to fill the vacant position of Secretary.

With Paul now presiding as President the new business continued with the selection of Denver, Colorado, as the site, and the third week in June, 1964, as the probable date of the next meeting. It was announced that the next fall meeting will be a joint meeting with the Midwestern Region, and will probably be held on the campus of Oklahoma University at Norman, Oklahoma, on November 9 or 22. The program committee will distribute the information regarding dates and agenda as early as possible, and they request abstracts of papers be sent in as soon as possible to aid them. A list of pre-registrants will be available early on the first day of the meeting.

Paul then announced results of the Executive Council meeting at Pittsburgh and they are as follows:

1. Jim Davidson is soliciting suggestions for methods for removing programs from the library. A criterion for value of the program and a means for the removal of inadequate programs is needed.

2. The cost of running the users group has risen. Dues paid by each region to the National 1620 Users Group were $.25 per registrant at each regional meeting. This has now been raised to $.50. Most of this increased cost has been in publication of the Newsletter. This must now be distributed to over 1,000 members while the average number of registrants at each meeting has not risen. It was stated that the registration fees for the meetings may have to be increased.

This concluded the new business and the business meeting was adjourned.

After an excellent luncheon, Mr. Melford E. Monsees, ADP Co-ordinator, U.S. Army Engineer District, Kansas City, Missouri, spoke on "The Impact of Automation on Professional Engineering." A copy of the talk is enclosed.

Two special interest groups met for discussion of mutual problems. The Civil Engineering group was presided over by Elias C. Tonias of Erdman and Anthony, Consulting Engineers, Rochester, New York, and minutes of the discussion are enclosed. The Power Utilities Engineer group was very informal but of benefit to all participants.

Sound-Off Session and Comments

It was announced that Version 2 of FORTRAN/FORMAT is available and has these advantages over Version 1 -

1. Multiple Format Specifications (also in the Pre-Compiler)

2. Source decks are available (on magnetic tape)

3. There are more subroutine options.

Ninety hours or more are required for field installation of 1311 disk drives.

Some operating difficulties with the disk were discussed and these points were brought out.

1. In order to use a FORTRAN IID or SPS IID object deck the Monitor I must be on a working drive. This is because no loader or subroutines are punched.

2. There is no current provision in Monitor I to load object decks to disk unless they are Monitor I compiled. It is therefore necessary to recompile all programs.

There is a great deal of interest in FORTRAN/FORMAT. Requests were made for a disk version and a version to batch compile. It can now be changed to allow free form input similar to the first 1620 FORTRAN compiler. Chuck Berry of IBM Wilshire office can supply instructions for this change.

Requests were made for the following from IBM:

1. Relocatable I/O and arithmetic subroutines for FORTRAN so only those used would be called.

2. A reduction in noise level on the Model 2 1622.

3. A FORTRAN pre-compiler which would give indication of the amount of storage the compiled program takes.

4. A Tabulate command in FORTRAN.

5. A loader for Systems Output Format decks punched under Monitor I control.

6. FORTRAN processors to take advantage of extended machine capabilities.

7. A Report Program Generator for the 1620-1443-1311 combination.

8. COBOL for 1620

9. ALGOL for 1620

10. FORTRAN IV.

Some of these requests were answered in the comments session by Jim Morgan and the other IBM representatives. The questions answered are:

1. Processors specifically designed for larger machine configurations are being investigated.

2. There are no present plans for IBM written COBOL, ALGOL, or Report Program Generators, however, there is an ALGOL processor available from Southern Illinois University.

3. The request for FORTRAN IV was noted and Jim asked for more opinions on the need for it.

The new hardware announced for the 1620 since the last Western Region meeting is the CalComp Plotter in two models, binary capabilities for the Model 2 - 1620, and index registers for the Model 2 - 1620.

It was noted that the FORTRAN II sine subroutine does not handle small angles correctly. IBM programming systems is working on the problem now.

Angelo Arena discussed the program library and said that the KWIC index is still in the process of being developed and may be changed even further, so that it will be easier to find programs by number. It will be published every six months with supplements issued every month. It will be 3-hole punched. Angelo requested that when programs are ordered, unless the user is sure the program fits his need, he order only the documentations. IBM can supply this quickly while program decks and tapes take longer. More descriptive titles for programs would also help this problem.

There is now available from IBM a manual which lists the RPQ's available (A26-5799). Some of these mentioned were:

1. The ability to punch one character on paper tape.

2. A real time clock. The clock IBM is now installing on all equipment is an elapsed time clock only and is not addressable.

3. An addressable IR-2.

Version 2 of FORTRAN II will be available in the near future.

The 1443 print commands are handled in the same manner as 7000 series machines with Column 1 being used for carriage control.

In order to get IBM publications without fail, have the IBM Systems Engineer put the installation name on the SRL list. Mailing is then automatic. Be sure that the installation receives the 1620 Bibliography, A26-5692.

The meeting was concluded with a demonstration of a Model 2 - 1620 with a 1311 disk file, on the evening of December 12, at the IBM Phoenix office and tutorial sessions on December 13 for MONITOR 1, FORTRAN II, and SPS 1620/1710.

1620 USERS GROUP WESTERN MEETING
PHOENIX, ARIZONA, WINTER 1963
CIVIL ENGINEERING SESSION
WEDNESDAY, DECEMBER 11, 1963, 3:30 PM

The highlights of this session were discussions on the disc file, the use of Fortran vs. SPS in Civil Engineering programs, the introduction of COGO and the use of the CALCOMP plotter.

It was the general consensus of the participants that SPS provided a better and faster object program than Fortran; Fortran however may be considered as the language for installations where speed in programming is preferred over speed in program runs.

The use of COGO in Civil Engineering installations should be reserved to engineers who are neophytes in the field of electronic programming and should not by any means replace existing programs or prevent the development of special programs.

Also of great interest was the use of the CALCOMP plotter now adopted by IBM. Successful programs have been written for contour plotting either with or without photogrammetric equipment and in plotting cross sections and profiles. It has also successfully been used in structural design in such cases as in the plotting of influence lines. Another field of successful use is that of hydraulic engineering and hydrology. The use of other plotting devices such as the Digital Scale and the Wilde T7 has tremendously helped in reducing conventional type of work and in tightening the interrelationship of photogrammetry, plotting and computing.

ELIAS C. TONIAS

1620 USERS GROUP
WESTERN REGION

AGENDA FOR 1963 WINTER MEETING
ARIZONA STATE UNIVERSITY
TEMPE, ARIZONA

DECEMBER 11, 12, 13, 1963

WEDNESDAY -- DECEMBER 11, 1963

8:00 - 10:00 Late Registration

9:00 Welcome and Opening remarks - R. Ebert, Regional President, and IBM Representatives

10:15 Coffee Break and informal "get acquainted" session

11:00 Technical Session "A"

A-1 Generalized SPS Routines for Handling Simple Problems.
Thomas L. Yates, Director, Statistics Computing Lab., Oregon State University, Corvallis, Ore.

A-2 Exponential and Sinusoidal Curve Fitting.
E. P. Hilar, Goodyear Aerospace, Litchfield Park, Ariz.

12:00 Break for Lunch

1:30 Technical Session "B"

B-1 Automated Design Engineering.
W. W. Rogers, IBM, Los Angeles, Calif.

B-2 A Payroll and Labor Distribution Program Package.
Elias C. Tonias, Erdman and Anthony, Consulting Engrs., Rochester, N.Y. Richard C. Devereaux, IBM, Rochester, N.Y.

B-3 The SPIRE System - Salaried Personnel Information REtrieval.
Gary J. Reed, Proj. Engr., ACF Industries, Albuquerque, New Mexico

3:00 Coffee Break

3:30 Technical Session "C"

C-1 A 1620 Program for Minimization of Boolean Functions, Expressed as Sums of Minterms.
Thomas R. Hoffman, Prof. of Elect. Engr., Union College, Schenectady, N.Y.

WEDNESDAY -- DECEMBER 11, 1963 (Cont'd.)

C-2 Critical Speed, Stress, and Bearing Reaction Calculations for a General Shaft, Using Numerical Integration.
Ralph B. Bates, Mgr. of Engr. Computing, Industrial Div. of American Standard, Detroit, Mich.

C-3 Three Dimensional Surface Fit.
David G. Kitzinger, ACF Industries, Albuquerque, New Mexico

5:00 Adjournment of Day's Sessions.

8:00 p.m. New Users Meeting, to be followed by Sound-Off Session at Approximately 8:30.

THURSDAY -- DECEMBER 12, 1963

9:00 Technical Session "D"

D-1 Maximum Likelihood Resolution of Two Mixed Normal Distributions.
Reimut Wette, D.Sc., Asst. Biometrician, The University of Texas, M.D. Anderson Hospital and Tumor Inst., Houston, Texas

D-2 Comparison of Two Methods of Finding Significant Contributors in Multiple Regression.
M. J. Garber, Director, Computing Ctr., University of California, Riverside, Calif.

10:15 Coffee Break

10:45 Technical Session "E"

E-1 Network Analysis.
H. N. Tyson, Jr., IBM, Los Angeles, Calif.

11:15 BUSINESS MEETING

12:00 LUNCHEON "The Impact of Automation on the Professional Engineer"
Melford E. Monsees, ADP Co-ordinator, U.S. Army, Corps of Engineers, Kansas City, Mo.

1:30 Technical Session "F"

F-1 An Integrated Earth Work System.
Cecil L. Ashley, ADP Co-ordinator, U.S. Army Engineering District, Walla Walla, Wash.

F-2 Hydro-System, A Daily Operation Analysis.
Charles R. Hebble, Civil Engr., U.S. Army Corps of Engineers, Walla Walla, Wash.

THURSDAY -- DECEMBER 12, 1963 (Cont'd.)

3:00 Coffee Break

3:30 Technical Session "G"
G-1 Ray Trace Program for a General Lens System.
D. H. O'Herren, Goodyear Aerospace, Litchfield Park, Arizona

4:00 IBM Reports and Discussion of Sound-Off Session.

5:00 Adjournment of Day's Sessions.

8:00 p.m. Demonstration of the 1620, Model II, with 1311 Disc Pack at the Phoenix IBM Branch Office. (Transportation will be arranged).

FRIDAY -- DECEMBER 13, 1963

9:00 Workshop Session "A"
A-1 MONITOR I

1:00 Workshop Session "B"
B-1 FORTRAN II
B-2 SPS 1620/1710

4:00 Conclusion of Workshop Sessions, and Final Adjournment of Meeting.

1620 USERS GROUP
WESTERN REGION

ABSTRACTS FOR 1963 WINTER MEETING
ARIZONA STATE UNIVERSITY
TEMPE, ARIZONA

DECEMBER 11, 12, 13, 1963

NO.	TITLE, AUTHOR, ABSTRACT

A-1 GENERALIZED SPS ROUTINES FOR HANDLING SOME SIMPLE PROBLEMS.
Thomas L. Yates, Director, Statistics Computing Lab., Oregon State University, Corvallis, Ore.
Two-way frequency distributions; output editing and formatting; card reproducing. (No further abstract available.)

A-2 EXPONENTIAL AND SINUSOIDAL CURVE FITTING.
E. P. Hilar, Goodyear Aerospace Corp., Litchfield Park, Arizona.
A specified number of exponentials are fitted to given equally spaced data points. The frequencies of the exponentials can be a mixture of real frequencies and complex conjugate pairs. The least mean square error criterion is used to find both the frequencies and the coefficients of the exponentials. The program is written in FORTRAN language. The limitations of the program will be discussed.

B-1 AUTOMATED DESIGN ENGINEERING.
W. W. Rogers, IBM Corp., Los Angeles, California
A presentation of Automated Design Engineering and how to achieve it. The role of a computer, ability to capture design logic, use of decision tables, comparison to manual design methods, and considerations involved with establishing Automated Design Engineering will be discussed.

B-2 A PAYROLL AND LABOR DISTRIBUTION PROGRAM PACKAGE.
Elias C. Tonias, Head of Data Proc. Dept., Erdman & Anthony, Consulting Engineers, Rochester, N.Y., and Richard C. Devereaux, IBM Corp., Rochester, N. Y.
The objective of this paper is to demonstrate how some free 1620 time may be utilized in a relatively small scientific or engineering installation through the use of a package of commercial programs. The successful operation of such a program package since the first of this year (1963) has helped to justify the installation of a 1620 in another firm. This package, designed for use with the basic 20K 1620

(Cont'd.)

Computer, with paper tape I/O and without any peripheral equipment, produces the payroll reports, and complete labor cost distribution and breakdown reports. Written in FORTRAN for easy maintenance, the ideas from this program package might prove a worthwhile tool in justifying the installation, or in increasing the production ratio of a 1620 in a small scientific or engineering account. With few or even no alterations, parts of the package may be used to handle other various time and cost distributions.

B-3 THE "SPIRE" SYSTEM - SALARIED PERSONNEL INFORMATION RETRIEVAL.

Gary J. Reed, Project Engineer, ACF Industries, Albuquerque, New Mexico

The Albuquerque Division of ACF Industries has developed a powerful management tool for use in personnel administration. This tool is the SPIRE system, utilizing an IBM 1620.

The system involves over 30 programs designed to utilize information from a master file of information maintained on magnetic tape. This master file contains over 300,000 different items of information. By judicious choice of information blocking, search time has been kept to a minimum for all applications.

The SPIRE system current applications include: in-plant recruitment, providing resumes of personnel qualified to fill vacant positions; preparation of quarterly reports containing information on employees eligible for merit considerations; creation of summary reports on salary increases for any time period; man power inventories; projections for salary budgeting; preparation of data for salary surveys; current salary status reports; and many other similar reports.

The information contained in the SPIRE system has proved to be complete for all applications thus far. The approach and system layout have provided an economic way of producing information that no hand techniques could supply at any cost. In the applications where the results could be obtained by clerical methods, enormous cost savings have resulted.

The SPIRE system has been operational since February 1963. The use of existing programs and the creation of new applications consistently increase reflecting overwhelming management acceptance.

C-1 A 1620 PROGRAM FOR MINIMIZATION OF BOOLEAN FUNCTIONS, EXPRESSED AS SUMS OF MINTERMS.

Thomas R. Hoffman, Prof. of Elect. Engr., Union College, Schenectady, N.Y.

A FORTRAN program implements a major portion of the Boolean minimization problem by reducing a sum of minterms to a logically equivalent set of prime implicants.

Minterms are entered as 3-digit octal-coded fixed point numbers. With the help of a special table, digit comparisons at the octal level reveal terms combinable according to the Boolean identity $XA + X\bar{A} = X$. Systematic

(Cont'd.)

determination of all possible combinations of this type, coupled with a bookkeeping system that keeps track of the different subsets produced, leads to the desired result. Prime implicants are printed out in octal code, to complete the processing.

The program can handle minterms having as many as nine variables, although the 1620 memory capacity (20K) may be exceeded in problems involving both large numbers of variables and long lists of minterms.

C-2 CRITICAL SPEED, STRESS, AND BEARING REACTION CALCULATIONS FOR A GENERAL SHAFT, USING NUMERICAL INTEGRATION.

Ralph B. Bates, Mgr. of Engr. Computing, Industrial Div. of American Standard, Detroit, Michigan

Critical speed, stresses, and bearing reactions can be calculated on a digital computer for a general two-bearing shaft, any number of cross sections, variable loading, and any length.

Besides eliminating the tedious labor of the calculations, the computer provides flexibility. A number of calculations may be rapidly made to optimize design, or to check out application variations on a standard design.

The method is illustrated in detail by an example calculation. Briefly it consists of dividing the shaft into increments of length, determining the load and shaft moments of inertia in each increment and the computer calculates critical speed, stresses, and bearing reactions.

C-3 M-151, THREE-DIMENSIONAL SURFACE FIT.

David G. Kitzinger, ACF Industries, Albuquerque, New Mexico

This code uses multiple interpolation techniques in combination with extensive transformation of variables to effect accurate fits of most smooth functions of three variables. With limited experience in selecting the form of curve fits in two dimensions, the programmer can fit complex three-dimensional surfaces. Output is designed to facilitate successive better approximations to the function in terms of additional transformation of variables. Second and third order fitting of functions is aided by a statistical error analysis. Two magnetic tapes and a 60K memory are required, although modifications can be easily made for smaller machines.

D-1 MAXIMUM LIKELIHOOD RESOLUTION OF TWO MIXED NORMAL DISTRIBUTIONS.

Reimut Wette, D.Sc., Asst. Biometrician, Univ. of Texas, M.D. Anderson Hospital and Tumor Institute, Houston, Texas

Iterative estimation of the five parameters from a sample taken from two normal distributions by the method of maximum liklihood seems preferable over the method of moments, because generalization to more than two parent distributions appears easier. Two approaches to estimate the maximum likelihood information matrix were abandoned in favor of the third, which cut down on computing time

(Cont'd.)

(1620 FORTRAN II) for the complete estimation procedure considerably. Computing time is, aside from variations due to behavior of the procedure depending on goodness of initial estimates and structure of the sample, directly proportional to the size of the sample. Therefore, a grouping-of-data program was developed specifically for this problem, and large samples, preceding the estimation procedure and reducing computing time to reasonable limits.

D-2 COMPARISON OF TWO METHODS OF FINDING SIGNIFICANT CONTRIBUTORS IN MULTIPLE REGRESSION.

M. J. Garber, Director, Computing Ctr., University of California, Riverside, Calif.

(No abstract available.)

E-1 NETWORK ANALYSIS.

H. N. Tyson, Jr., IBM Corp., Los Angeles, Calif.

Discussion will center about the general technique of network analysis, and the capabilities of a system of the 1620 programs utilizing this technique for the analysis of electronics circuits. The programs operate under MONITOR I on a 40K 1620. The capabilities allow for AC, DC, and transient analysis.

F-1 AN INTEGRATED EARTH-WORK SYSTEM.

Cecil L. Ashley, ADP Co-ordinator, U.S. Army Engineering District, Walla Walla, Washington

(No Abstract Available.)

F-2 HYDRO SYSTEM DAILY OPERATION ANALYSIS

C. E. Hildebrand, L. A. Dunstan, C. R. Hebble, R. D. Moffitt, U.S. Army Engr. District, Walla Walla, Washington

The program is a mathematical model of a system of hydro-electric projects. It is a general program applicable to any river system and scheme of development. It is capable of accurately simulating the hour-by-hour operation of a hydro system for as long a real time period as desired. It will determine the effects produced by existing hydro stations in regards to reservoir levels, river stages, and alternative distributions of system load among a group of hydraulically and electrically integrated hydro stations. The program makes it possible to determine the operating characteristics of planned future projects in regards to backwater encroachment on upstream reservoirs, pondage requirements, the effect of peaking discharges on downstream river stages and reservoirs, and effects of added power installations. Two versions of the program exist: One for an IBM 1620 with 40K memory, and one for an IBM 1620 System with 60K in the IBM 1620 and 4K in the IBM 1401.

G-1 RAY TRACE PROGRAM FOR A GENERAL LENS SYSTEM.

D. H. O'Herren, Goodyear Aerospace Corp., Litchfield Park, Arizona

This program, written in FORTRAN with Format, and requiring 40K core storage, is designed to trace rays through complex lens systems, outputting intersection points at selected surfaces in the system. Cylindrical, conic, and toric surfaces can be handled and can be arbitrarily oriented with respect to the optical axis. Lens surfaces can be virtually any contour describably by second degree (or less) equations or at least divisable into sections which can be so described. A sample ray tracing problem is given to illustrate the use of the program.

The compiler workshop sessions to be held on Friday will assume that those attending will have a working knowledge of the externals of the programming systems (i.e. how to write programs in FORTRAN or SPS, etc.). The purpose of these sessions will be to present some of the "Hows" and "Whys" of the internal workings of the compilers. The session on MONITOR I will include more information on using the monitor, as well as delving into some of the internals.

THE IMPACT OF AUTOMATION ON PROFESSIONAL ENGINEERING

A PAPER PRESENTED

AT

ARIZONA STATE UNIVERSITY

TEMPE, ARIZONA

12 DECEMBER 1963

BY

MELFORD E. MONSEES
ADP COORDINATOR
U. S. ARMY ENGINEER DISTRICT
KANSAS CITY, MISSOURI

SOCIAL AND ECONOMIC ASPECTS OF AUTOMATION

There are many differing views regarding the impact that the electronic computer will have on the development of our social and economic lives, but most of us can agree that the potentialities of automatic data processing (ADP) are limited only by the boundaries of our individual imaginations. The outlook, therefore, is open to wide-ranging speculations.

The truly great impact of digital computation will be a dramatic speedup in the rate of technological progress -- and concurrently in the evolution of our social and economic lives.

Electronic data processing is one of the most powerful catalysts of technological development yet discovered. This is so because ADP has the ability of extending the capabilities of man's intellect.

The human mind is the most powerful, most versatile, most useful natural gift bestowed upon man by his Creator. Any instrument that can substantially increase its capabilities is certain to have a profound effect on our future development.

Man succeeded in building our technology to an extraordinarily sophisticated level during World War II without an aid to his intellectual capability even remotely approximating the power of ADP. Now, with the aid of computers that increase the productivity of his intellect in many areas by factors running into the tens of thousands, we are certain to see significant advances in the tempo of technological, economic and social progress.

Another broad area of impact will be in fields of business, industry and communications. We are all familiar with the progress ADP has already made in automating the business office. And computers are now being used increasingly in manufacturing--not only in process control applications, but also as an aid to the efficient management of the overall manufacturing operation.

We are entering a battle for our very survival in the market places of the world. Much emphasis is being placed on economic development. Technological progress, rising productivity and ascending standard of living are the true sources of economic strength. They are vital to national survival in today's competitive world.

As we move forward, we will encounter the problems always inherent with social and economic change. These problems should in no way warrant artificial restrictions on technological development for this is vital to the success of any business in a free economy.

It may well turn out that the efforts of the new technology will be far more lastingly felt in its impact on many of the traditional principles and practices of management. Many traditional personnel practices are obviously going to be automated or abolished and various leader groups will change in power and prestige.

While there are no precise means as yet of measuring the speed of technological change, it is reasonable to assume that by the mid-1960's, as those born during the Second World War establish families and the challenge of foreign competition becomes more intense, the present rate of change will increase. At the present time, productivity increases in the nonfarm sector amount to roughly 2.5 percent a year. Even a moderate speedup in this rate would mean that by 1990, a relatively short span of less than 30 years, industry could double its production with the same labor as it employs today.[1]

THE ENGINEERING EMPLOYMENT SITUATION[2]

Engineering, the second largest professional occupation, is exceeded in size only by teaching; for men, it is the largest profession. The approximately 875,000 engineers in the United States in mid-1960 have made major contributions to the design, construction, and efficient utilization of the machines, equipment, roads, and buildings used by the Nation's 180 million people. Engineers provide technical, and frequently, managerial leadership in industry and Government. They develop new products and processes, design many types of machines and structures, and contribute in countless other ways to the technological progress of the country and to the national defense.

The outlook is for continued rapid expansion of the engineering profession. Engineering has been one of the fastest growing professional

---

[1]Joseph A. Raffaele, "Automation and the Coming Difussion of Power in Industry," Personnel, (May-June 1962), 30.

[2]U. S. Bureau of Labor Statistics, Employment Outlook for Engineers, (Washington: U. S. Government Printing Office, 1961), pp. 101, 103-104.

occupations in the United States in the past 50 years, and there is every indication that the demand for engineers will continue to grow. As in recent years, there will probably be a particular need for engineers with advanced degrees to teach and to do research.

Some of the major factors expected to raise the demand for engineering personnel are: Continued high levels of Government spending for defense, accentuated by the increasingly large amount of engineering time necessary for the development of modern weapons; growth of population and expansion of industry; increasing complexity of industrial technology, as such the trend toward automation of industrial manufacturing processes; and further growth in expenditures for research and development. In particular, the large sums spent for research and development in recent years by both industry and Government -- total research and development expenditures in the United States amounted to more than $13 billion in 1960-61 -- have broadened existing areas of employment for engineers and opened up new ones, such as those concerned with computers, missiles, and nuclear energy. As scientific frontiers are extended, more areas of work for engineers will be provided.

THE EMERGENCE OF AUTOMATION IN ENGINEERING

That the systems concept is having a reaction in the engineering field is generally recognized. The question is how much. The change in a lot of features surrounding the engineering profession is beginning to assume large proportions and that rate of change is increasing rather than leveling off.

For instance, we are told that the amount of engineering information and scientific information which is directly relevant to engineering problems has doubled within the last fifteen years. Even now the volume of knowledge related to engineering is so great that no one man can possibly know it all, even though he does nothing but study from boyhood to senility. For us engineers that means two things. We must specialize more than was formerly necessary or desirable, and second, we must be diligent students throughout our active lives. If we fail to study and keep abreast of the developments which directly apply to our chosen fields of engineering, we shall quickly become back numbers and soon thereafter become useless to an advancing civilization.

As a civil engineer, I am more familiar with the developments in that field than in some of the others. I recall the observations I have made and the discussions I have had with various men over the past 25 years, relative to the extreme reluctance of engineers to adapt labor-saving devices <u>in their own work</u>. Our fellow engineers were accomplishing much in the industrial world in devising and perfecting labor-saving equipment, but we civil engineers were extremely slow in demanding any sort of machinery or equipment that would make our work easier.

For example, up to a few years ago, we were using the same methods of surveying that had been devised in ancient times. The transit was a little better than the old surveyor's compass, but the long and tedious process of making ground surveys was basically unchanged for 3,000 years. Now electronic measuring devices for survey parties are becoming standard equipment.

The electronic computer is an item which cuts across all the fields of engineering and many of the areas of science as well. They have been in use now for only about 15 years, but in that time they have increased the computational ability of mankind one million times. The industry has grown in 15 years from zero to the production of a billion dollars worth of equipment during the year 1960. Since that date there has been an upsurge in the number of manufacturers of electronic computers. Competition has become keener than it was a few years ago and we can confidently look forward to a greater variety of computers, composed of more dependable pieces of equipment, at a cheaper price than they command today.[3]

The electronic computer systems have eliminated much of the routine drudgery that has long been the bane of an engineering office. Also, they have eliminated the need for those men who were fitted for nothing more than routine work. There will be less and less need for the man who can only run a calculator or a slide rule, and who has been noted in an office

---

[3]Murray A. Wilson, "Change or Progress," American Engineer, (June, 1962), 29.

primarily because he could remember all the formulae that applied to his line of work, able to recall and use them accurately as the occasion demanded. This work, the machine can do better and much more rapidly.

The other feature is that the computer can solve the basic, theoretical equation and eliminate the necessity for the approximations and shortcuts that have been in common use in so many phases of design, simply because the basic formulae were so long and complicated that the process of solving them longhand took so long that no one could afford to use them.

On the other hand, the new facilities will put a premium on imagination and ingenuity. These have always been desirable qualities in an engineer, but in the past a great many engineers have been kept gainfully employed on jobs that require little of either. This situation is changing and the prizes in the future are going to those men who are well endowed with these two important attributes.

EFFECT OF AUTOMATION ON ENGINEERING EDUCATION

Actual and tangible changes in the practice of the art and science of engineering are being reflected in our schools and are under constant discussion in publications, so it seems probable that we have created a new concept of the profession of engineering, or if you please, a new image of the engineer. The curricula in our colleges are in a state of fluidity with considerable differences in their means of meeting the challenge of the changing conditions. On one thing all seem to agree--that is, that the engineer of tomorrow and the day after, will need to be more thoroughly grounded in the basic physical sciences than was thought necessary for his predecessor. Others have felt that an equally important need of the coming engineers is a knowledge of the past such as is gained through a study of the "humanities." In an attempt to make room for these additional courses in a curriculum already overcrowded, something had to give, so the subprofessional subjects such as shop practice, material testing, laboratory work of various kinds, surveying and similar courses are being eliminated. The logical justification of this elimination is that these functions are actually to be performed by subprofessional men anyway, or as we propose to call them, the engineering technicians.

Modern problems of engineering are no respecters of traditional boundaries between the specialities. Accordingly, changes are being made by schools of engineering. It is understood that Dr. Keith Glennan has made broad moves which will go far to establish interdisciplinary approaches in engineering education at Case Institute of Technology. At the undergraduate level he has consolidated the departments of chemical, civil, electrical and mechanical engineering into a single administrative unit -- the Engineering Division. The engineering faculty are re-grouping in a natural way according to their common professional interests such as systems, design, energy conversion, materials, information processing and other emerging fields.

Degrees in electrical, mechanical, civil and chemical engineering will continue to be granted, but a new degree at Case -- probably named Bachelor of Science in Engineering -- will be offered. It will give the student the opportunity to plan his elective program -- with faculty advice -- to suit his career interests. Such a program can be designed to lead more effectively into advanced work.

There is little doubt that the four-year graduate will continue to play an important role in industry, but there is an ever increasing need for the engineer with the depth of knowledge and experience produced by work at the advanced graduate level.

Also, Dean B. R. Teare, Jr. of the College of Engineering and Science, Carnegie Institute of Technology, said recently in a letter to me -- "The electronic computer has certainly made an impact on the individual courses in our engineering programs but it has not yet been the reason for extensive curriculum changes. Some engineering departments, pressed with the lack of time in a four-year program, have had to decide between continuing courses in engineering graphics and courses in computer logic."[4]

The electronic computer is also having its impact on the curriculum at M.I.T. In a letter I received last January from Dean Gordon Brown's office some of the directions in which the computer was leading were outlined.[5] For example the inclusion in the curriculum at M.I.T. of the following subjects:

Digital Computer Programming Systems

Mathematical Methods in Civil Engineering

Digital System Application.

---

[4]B. R. Teare, Jr., Carnegie Institute of Technology, in letter to author dated January 3, 1963.

[5]Gordon Baty, Adm. Asst. to Dean of school of Engineering, M.I.T., in letter to author dated January 18, 1963.

Other related subjects have been included in the curriculum, but it pointed out that the automatic computations is not presented as an end in itself, but as another tool of analysis in the engineer's kit. The subjects merely focuses upon the techniques available to the engineer for exploiting the power of electronic computations.

Dean Brown's office also advised me that the impact of the computer upon research activities has been enormous. A copy of the semiannual Report, available from the M.I.T. Computation Center, can give you some idea of its magnitude. Yet, however important these computer-related activities have become to the School, there is a danger involved in attributing any of them simply to the availability of computer technology. For this is only one of the influences which have converged to create many of our most exciting research projects and subject offerings. Others include new methods in statistics and operation analysis, systems analysis and synthesis, theory of learning, and the information technologies.

TANGIBLE BENEFITS FROM AN ELECTRONIC COMPUTER SYSTEM

First hand knowledge of the impact of automation on professional engineering has been obtained as a member of the staff of the U. S. Army Engineer District, Kansas City, Missouri. This Corps of Engineers office has civil works engineering and construction in States of Kansas, Missouri, Nebraska, Iowa, and Colorado, and military engineering and construction in the States of Kansas and Missouri. The total work of this district averages about 70 million dollars per year and includes the design and construction of large multipurpose dams, levees, floodwalls, channel improvement works, pumping plants, and necessary utilities, highways, railroads, bridges, etc., required to be relocated in connection with flood control projects. Also, included in the assigned work of the district is the construction of military facilities and structures for the Army and Air Force.

Computer facilities for engineering applications have been available since January 1958. The initial computer was the Burroughs E-102, which, although of limited capability, was used until replaced in December 1960 with an IBM 1620 paper-tape system. Utilization of the paper-tape system for engineering applications increased rapidly and in January 1962, the system was augmented to provide for high speed punched card input-output. During this period, the utilization of electronic computer systems has provided tangible benefits through reduced cost of construction, savings in engineering and clerical manpower, and by providing a superior end-product or flood control structure. To date the major effort in implementing ADPS procedures has been directed toward high-benefit engineering applications. Only preliminary phases of planned implementation of ADPS in areas of personnel administration, property accounting, real estate activities and fiscal responsibilities has been possible. As of December 1963, over 75 computer programs were being used in the fields of mechanical engineering, structural engineering, hydraulic engineering, hydrology, reservoir regulation and earthwork and soil mechanics.

The currently installed system in the Kansas City District Office has provided the following:

a. Improvement of the engineering and design product.

b. More timely and accurate data which is fully responsive to the needs of management and engineering, including data previously not economically obtainable.

c. Savings in costs by maintaining continued evaluation and balance of equipment and personnel.

d. Savings in costs and engineering manpower by application of ADP principles.

e. Simplified and reduced manual data handling and eliminated duplication of files, reports, and entrance of source data.

Further benefits are being obtained through realization of the following objectives:

a. Expansion of hydropower and pumped storage-power study programs to be used in connection with reservoir regulation, hydrology and hydraulic engineering programs used for design, construction, and operation of multipurpose projects.

b. Expansion and refinement of structural design analysis programs used for preliminary and final design of various civil engineering projects.

c. Expansion and refinement of earthwork and soils mechanics programs used for quantity computations and stability analysis of large and small embankments.

d. Implementation and expansion of Critical Path Scheduling techniques to be used for coordination of construction activities as well as coordination and scheduling of engineering and design programs.

e. Refinement of personnel administration and reporting procedures.

f. Implementation of one additional phase of the engineering budget management data to eliminate manual posting and to provide more accurate readily accessible data for management and estimating purposes.

g. Exploitation of the principle of "management by exception" through the potentialities of data processing equipment and techniques by continued education and training of personnel in the use of machine oriented reporting procedures and elimination of duplication of detail.

h. Full utilization of presently installed data processing equipment by insuring that all data processing activities are essentially high-benefit programs.

CONCLUDING OBSERVATIONS

Last year, 1962, was a year which may very well be recognized as the beginning of the first plateau of maturity for the industries that automatic control has helped to create.

Signs of maturity are also evident among the scientists and engineers who created these new industries and who must continue to act as whole partners with management and finance in continuing to create and exploit the new scientific breakthroughs which will firmly establish automatic control in its ultimate position as the greatest servant of mankind. This maturity takes a number of forms:

a. The number of engineers and scientists enrolled in post-graduate or extension courses in management is testimony that the importance of market and management factors in technical decisions is not widely recognized by the technical experts. The idea of technical performance, cost, schedule, physical characteristics, and reliability has now become widely accepted as a factor in technical decisions. Finally, an increasing number of scientists and engineers has come to realize that their technical brilliance is wasted if their ideas cannot be sold, and that it is a hollow satisfaction to be able to prove that something new and wonderful can be done unless means are found to ensure that it will be done.

b. Electronics engineers, dynamicists, and even civil engineers have discovered the benefits of computers; and the electrical, mechanical, and field service engineers are recognizing the importance of automation in the translation of their diagrams and equations into operating realities.

c. The overriding importance of reliability in concept, design, manufacture, operation, and maintenance of automatic control systems has been thoroughly recognized.

d. Our new ability to use high-speed, high-capacity digital computers as controllers for automatic control systems, plus the advent of microminiaturization, has provided the flexibility for universal application to systems of almost any complexity, involving almost any combination of scientific disciplines. This new flexibility provides the base from which automatic control can be adapted to applications ranging from space vehicles to automatic factories -- from complex air-traffic-control systems to the most microscopic of biological measurements and processes -- from the unmanned vehicles of oceanological exploitation to the complex man-machine systems of industry and sociology.

e. In the areas of technology, today, because of the broadening application of this industry into all fields, the engineer and the automatic control technologist must be able to understand and communicate with technologists from almost every field of endeavor. This universality of automatic control science application has brought about a lowering of the barriers of disciplinary specialty to permit an intermingling of the most widely diversified technologies.

Finally, there is no doubt that the electronic computer has had its effect and will continue to have an impact, not only on professional engineering but on the curriculum and educational program in our schools of engineering throughout the United States.

BIBLIOGRAPHY

Joseph A. Raffalle. "Automation and the Coming Diffusion of Power in Industry," Personnel, (May - June 1962).

Frank W. Reilly. "Policy Decisions and EDP Systems in the Federal Government," Public Administration Review, (September 1962).

U. S. Bureau of Labor Statistics. Employment Outlook for Engineers, (Washington: U. S. Government Printing Office, 1961).

Murray A. Wilson. "Change or Progress," American Engineer, (June, 1962).

B. R. Teare, Jr. Carnegie Institute of Technology, ltr. dated January 3, 1963.

Paul O. Roberts. "Computer Models of Future Roads," (The Technology Review, December, 1962).

Martin Greenberger. Management and the Computer of the Future, (New York: John Wiley & Sons, Inc., 1962).

Glenn Murphy. "Whither Engineering Education," Journal of Engineering Education (November, 1962).

Roger W. Bolz. "Automation as a Social Problem," American Engineer, (March, 1962).

GOODYEAR

GOODYEAR AEROSPACE

CORPORATION

ARIZONA DIVISION

LITCHFIELD PARK, ARIZONA

EXPONENTIAL AND SINUSOIDAL CURVE FITTING

by

E. P. Hilar

AAP- 18845 November 20, 1963

LIST OF REFERENCES

(1) Dr. F. A. Willers, Practical Analysis, Graphical and Numerical Methods, Dover Publications, Inc., 1948

(2) Louis Weisner, Introduction to the Theory of Equations, The MacMillian Co.

(3) 1620 Polynomial Rootfinder by Barstows Method, J. W. Wentzien, IBM 1620 Library (1620 - 07.0.040)

INTRODUCTION

This paper presents a program written around the method of fitting exponental curves to data as given by F. A. Willers in his book, "Practical Analysis"[1]. The program fits exponentals with either real or complex conjugate pair frequencies. The frequencies and the coefficients of the exponentals are fitted using the least mean square error criteria. The number of exponentals to be fitted is specified as part of the input data. The program is written in Fortran language.

THEORY OF METHOD

Given N observations equally spaced in the independent variable x by an amount h and originating at $x = 0$, it is desired to fit a sum of n exponentals to them. The observations will be represented by the dependent variable y. The desired fit will be written

$$y(x) = Co + \sum_{1}^{n} C_j e^{\alpha_j x} \tag{1}$$

where the coefficients, C, are real, and the frequencies, α, are real or occur in complex conjugate pairs which give rise to terms of the form

$$C\, e^{\alpha x} \cos(\beta x + \phi) \tag{2}$$

in Equation (1).

The restriction on N, the number of data points, is

$$N > n + 1 \tag{3}$$

The derivation of the method of fitting the exponentals to the data will begin by assuming that the points actually fit the exponental representation exactly. An expression involving only the frequencies, α, and the successive differences of the data points will be derived. The fiction of an exact fit will then be removed by introducing an error term into this relationship. The above assumption may be written

$$y(m) = Co + \sum_{1}^{n} C_j e^{\alpha_j (m-1)h} \tag{4}$$

where y(m) is the m^{th} data point.

Defining the difference between successive data points as

$$D(j) = y(j+1) - y(j) \tag{5}$$

Using Equation (4), Equation (5) may be written as

45

$$D(j) = \sum_{1}^{n}{}_{j} \; C_j(e^{\alpha_j h} - 1)\, e^{\alpha_j (m-1)h} \tag{6}$$

Defining

$$U(j) \equiv e^{\alpha_j h} \tag{7}$$

$$f(j) \equiv C_j e^{\alpha_j (m-1)h} \tag{8}$$

Equation (6) may be written

$$D(j) = \sum_{1}^{n} (U(j)-1)\, f(j) \tag{9}$$

Using Equations (5) through (8)

$$D(m+k) = y(m+k+1) - y(m+k)$$

$$= \sum_{1}^{n}{}_{j} \; f(j)\,(U(j)-1)\,U(j)^k \tag{10}$$

Now n+1 equations of the form (10) will be written out

$$D(m) = \sum f(j)(U(j)-1)$$

$$\vdots$$

$$D(m+n) = \sum f(j)(U1_j)-1)U_{(j)}{}^n \tag{11}$$

where the sum runs from 1 to n on j.

Under the assumption that the curves are an exact fit, all of these equations (11) hold true. Now if we consider them as n+1 equations in the n unknown f(j) then the determinate of the equations must be zero. This restriction on the determinate yields the following equation

$$\sum_{k=0}^{n} D\,(m+k)\,S(n-k) = 0 \tag{12}$$

The functions S(j) are the symmetric functions and are defined in the two equivalent forms shown below[2].

$$S(0) = 1$$

$$S(1) = -\sum_{j=1}^{n} U(j)$$

$$S(2) = \sum_{j=1}^{n} U(j) \left(\sum_{k=2}^{n-1} U(k)\right) \tag{13}$$

$$\vdots$$

or, given an n^{th} order polynomial whose zeros are the U(j)s, then the polynomial can be written as

$$\sum_{k=0}^{n} S(k) z^{n-k} = 0 \tag{14}$$

where S(0) = 1

In the problem of fitting the exponentals to the data we do not know the value of the symmetric functions before hand. If we did the required frequencies could be calculated from them. But by removing the fiction of an exact fit through introducing an error term into Equation (12) and writing it as

$$\sum_{k=0}^{n} D(m+k)\,S(n-k) = \epsilon(m) \tag{15}$$

the error can be minimized by the proper choice of the symmetric functions. The means of choosing them will be the least mean square error criteria.

Squaring Equation (15) and summing over the permissible values of m yields

$$\sum_{m=1}^{N-(n+1)} \left(\sum_{k=0}^{n} D(m+k)\ S(n-k) \right)^2 = \sum_{m=1}^{N-(n+1)} \epsilon(m)^2 \tag{16}$$

Since it is the symmetric functions that are being fitted, the partial derivatives with respect to the symmetric functions of the left hand side of Equation (16) are equated to zero yielding n equations of the form

$$\sum_{k=0}^{n} \left(\sum_{m} D(m+\ell)\ D(m+k) \right) S(n-k) = 0 \tag{17}$$

where ℓ runs from 1 to n and all sums on m run from 1 to N-(n+1) unless otherwise noted.

The D(j)s are calculated from the data and the n simultaneous Equations (17) are used to solve for the best fit symmetric functions. Equation (14) is then solved yielding for its roots the U(j)s. For a real root the frequency is calculated from

$$\alpha_j = \frac{\ell n\ U(j)}{n} \tag{18}$$

For a complex conjugate pair of roots of the form

$$a \pm ib \tag{19}$$

the corresponding frequencies are calculated from

$$\alpha = \frac{1}{2h}\ \ell n(a^2+b^2) \tag{20}$$

$$\beta = \frac{1}{h}\ \arctan\ (b/a) \tag{21}$$

where the corresponding terms in Equation (1) are now of the form (2).

Once the frequencies have been found the fitting of the coefficients of Equation (1) to the data is a straight forward problem using the least mean square error criteria on the following equation:

$$Co + \sum_{1}^{n} C_j e^{\alpha_j (m-1)h} - y(m) = \epsilon(m) \tag{22}$$

where the coefficients, C, are now to be fitted. The development from this point on is Fortran oriented and will be presented in the discussion of the program itself.

METHOD AND PROCEDURE

The description of the program will emphasize the means used to speed up the calculations of the various portions of the problem. Once these are understood the listing becomes self-explanatory.

The first major task is to generate the coefficients of the symmetric functions in Equations (17). To accommodate the DO loops the following definition will be made

$$DE^{m}(j) \equiv D(m+j-1) = y(m+j+1) - y(m+j) \tag{23}$$

In the program the symbol is written

DE(j) and the value of m is kept track of by a DO loop.

Now defining

$$E^{m}_{j,k} \equiv DE^{m}(j) * DE^{m}(k) \tag{24}$$

The n equations (17) may be written

$$\sum_{k=1}^{n+1} \left(\sum_{m} E^{m}_{j,k} \right) * S(n+1-k) = 0 \tag{25}$$

where j runs from 1 to n

The desired coefficients are the sums

$$\sum_{m} E^{m}_{j,k} = \sum_{m} DE^{m}(j) * DE^{m}(k) \tag{26}$$

A rapid method of calculating these coefficients will now be described. This method makes use of the following properties which are easily derived from Equations (23) and (24).

$$DE^{m+1}(j) = DE^{m}(j+1) \tag{27}$$

$$E^{m+1}_{j,k} = E^{m}_{j+1,k+1} \tag{28}$$

$$E^{m}_{j,k} = E^{m}_{k,j} \tag{29}$$

$$\sum_{m} E^{m}_{j,k} = \sum_{m} E^{m}_{k,j} \tag{30}$$

The quantities DE, E, and $\sum_{m}$ E are stored as they are calculated. The procedure is as follows: m is set to one, and all the required values of the DEs and the Es are calculated and stored. The values of the Es are entered into the coefficient sums of the form (26) as the first term. Then m is set to 2 and the DEs are shifted one space to the left using the property shown in Equation (27). The missing term is calculated from Equation (23). The stored Es are shifted up and left one space using the property shown in Equation (28). The missing terms are calculated using Equation (24). The values of the Es are entered into the coefficient sums as the second term. The program proceeds as for m = 2 until all the terms of the coefficient sums have been entered. A separate but concurrent aspect is that of storage. The properties of Equations (29) and (30) show that only terms on and to the right of the diagonal need be calculated. The remaining space is used for storage. The terms $E^{m}_{j,k}$ are stored in E(j,k+1) and the terms $\sum_{m} E^{m}_{j,k}$ are stored in E(k,j). At the end of the procedure described above Equation (30) is used to complete the array of coefficients.

The n simultaneous linear Equations (25) are solved by a Gaussian reduction routine. The solution yields values for the n symmetric functions (13).

The n^{th} order polynomial (14) is now solved using a stripped down version of the Barstow method[3]. The roots of Equation (14) may be either real or occur in complex conjugate pairs. In the program the roots are stored in the following manner: Two fields R(j) and M(j) are defined where j runs from 1 to n. If the j^{th} root of Equation (14) is real then it is stored in R(j) and M(j) is set to zero. If the j^{th} and j+1 roots are a complex conjugate pair then the real part of the roots is stored in R(j) and M(j) set to one and the imaginary part of the roots is stored in R(j+1) and M(j+1) is set to two. The field M(j) serves to identify the contents of the field R(j).

The program does not use the variable S(j) directly but stores the solutions of the n linear equations in the field x(j) where

$$S(j) = x(n+1-j) \quad 0 < j \leq n \tag{31}$$

The entry into the polynomial solver places the symmetric functions in the field A(j) where

$$A(j) = S(j-1) = x(n+2-j) \quad 1 < j \leq n, \quad A(1) = S(0) = 1 \tag{32}$$

The transfer from x to A is done directly.

Once the roots have been found the frequencies may be calculated using the following form of Equations (18), (20) and (21)

$$\alpha_j = \frac{1}{h} \ell n(R(j)) \quad \text{when } M(j) = 0$$

or

$$\left.\begin{aligned} \alpha_j &= \frac{1}{2h} \ell n(R(j)^2 + R(j+1)^2) \\ \beta_{j+1} &= \frac{1}{h} \arctan\left(\frac{R(j+1)}{R(j)}\right) \end{aligned}\right\} \begin{aligned} &\text{when } M(j) = 1 \\ &\text{and } M(j+1) = 2 \end{aligned}$$

If a real root or the real part of a complex root is negative then the remainder of the program cannot give meaningful results and the program halts.

The remainder of the program is a straight-forward determination of the coefficients of Equation (22). The discussion will be concerned with a rapid way of generating the needed numbers. The most general form for n = 3 will be considered here. The generalization of the results is quite straight-forward. Equation (22) is rewritten

$$Co + C_1 e^{\alpha_1 (m-1)h} + C_2 e^{\alpha_2 (m-1)h} \operatorname{Cos}(\beta_3 (m-1)h)$$

$$+ C_3 e^{\alpha_2 (m-1)h} \sin(\beta_3 (m-1)h) - y(m) = \epsilon(m) \tag{34}$$

Defining DO loop notation

$$\overset{m}{V}(1) = 1$$

$$\overset{m}{V}(2) = e^{\alpha_1 (m-1)h}$$

$$\overset{m}{V}(3) = e^{\alpha_2 (m-1)h} \ \mathrm{Cos}(\beta_3 (m-1)h) \tag{35}$$

$$\overset{m}{V}(4) = e^{\alpha_2 (m-1)h} \ \mathrm{Sin}(\beta_3 (m-1)h)$$

$$\overset{m}{V}(5) = -y(m)$$

$$x(j+1) = C_j$$

where m is a superscript on V.

Using Equations (35) Equation (34) is written as

$$x(1)*\overset{m}{V}(1)+x(2)*\overset{m}{V}(2)+x(3)*\overset{m}{V}(3)+x(4)*\overset{m}{V}(4)+\overset{m}{V}(5) = \epsilon(m) \tag{36}$$

Fitting the Cs or the xs by the least mean square error criteria as in the first part of this report yields n+1 linear simultaneous equations of the form

$$\sum_{k=1}^{n+1} \left(\sum_{m} \overset{m}{V}(j)*\overset{m}{V}(k) *x(k) \right) + \sum_{m} \overset{m}{V}(j)*\overset{m}{V}(n+2) = 0 \tag{37}$$

where j runs from 1 to n+1 and all the sums on m now run from 1 to N unless otherwise noted.

Defining

$$\overset{m}{E}(j,k) = \overset{m}{V}(j) * \overset{m}{V}(k) \tag{38}$$

where m is a superscript.

Equations (37) become

$$\sum_{k=1}^{n+1} \left(\sum_m \overset{m}{E}(j,k) \right) * x(k) + \sum_m E^m(j,n+2) = 0 \tag{39}$$

where j runs from 1 to n+1.

Now the Vs may be calculated from equations of the form (35) for each m but the amount of time spent doing so is prohibitive. Recursion formulas may be developed for the Vs by noting that Equations (33) lead directly to

$$R(j) = e^{\alpha_j h} \qquad \text{when } M(j) = 0$$

and

$$\left.\begin{aligned} R(j) &= e^{\alpha_j h} \cos(\beta_{j+1} h) \\ R(j+1) &= e^{\alpha_j h} \sin(\beta_{j+1} h) \end{aligned}\right\} \text{when } M(j) = 1 \text{ and } M(j+1) = 2 \tag{40}$$

Now from Equations (35) and (40) it is obvious that

$$\overset{m+1}{V}(1) = \overset{m}{V}(1) = 1 \tag{41}$$

$$\overset{m+1}{V}(2) = \overset{m}{V}(2)\, e^{\alpha_1 h} = \overset{m}{V}(2) * R(1) \tag{42}$$

$$\overset{1}{V}(2) = 1 \tag{43}$$

The recursion formulas for V(3) and V(4) are obtained with the help of well known trigonometric identities as

$$\begin{aligned} \overset{m+1}{V}(3) &= e^{\alpha_2 mh} \left(\cos(\beta_3(m-1)h) \cos \beta_3 h - \sin(\beta_3(m-1)h) \sin \beta_3 h\right) \\ &= \overset{m}{V}(3) * R(2) - \overset{m}{V}(4) * R(3) \end{aligned} \tag{44}$$

$$\overset{1}{V}(3) = 1$$

and

$$\begin{aligned} \overset{m+1}{V}(4) &= e^{\alpha_2 mh} \left(\sin(\beta_3 (m-1)h) \cos \beta_3 h + \cos(\beta_3 (m-1)h) \sin \beta_3 h\right) \\ &= \overset{m}{V}(4) * R(2) + \overset{m}{V}(3) * R(3) \end{aligned} \tag{45}$$

$$\overset{1}{V}(4) = 0$$

The generalized recursion relationships are

$$V(1)^{m+1} = 1$$

$$V(j)^{m+1} = V(j)^{m}*R(j-1) \text{ when } M(j-1) = 0$$

or

$$V(j)^{m+1} = V(j)^{m}*R(j-1) - V(j+1)^{m}*R(j)$$

$$V^{m+1}(j+1) = V(j+1)^{m}*R(j-1) + V(j)^{m}*r(j) \quad \left\{\begin{array}{ll}\text{when } M(j-1) & = 0 \\ \text{and } M(j) & = 2\end{array}\right. \qquad (46)$$

$$V(n+2)^{m+1} = -y(m+1)$$

and the values for m = 1 are

$$V(1) = 1$$

$$V(j) = 1 \quad \text{when } M(j-1) = 0, 1$$

or $\qquad$ (47)

$$V(j) = 0 \quad \text{when } M(j-1) = 2$$

$$V(n+2) = -y(1)$$

The program calculates the sums on m in Equations (39) somewhat as before. The value of m is set to one and the initial values of the Vs are calculated from Equations (47) using M(j) to control the choice of equations. The Vs are stored and the Es are calculated using equation (38) and entered directly into the sum on m of equations (39) as the first term. The value of m is then set to two and Equations (46) used to calculate the new Vs with M(j) as the control. The Es are calculated and entered into their sum as before. The program proceeds as for m = 2 until the sums are complete. As in the early part of this paper only the terms on and to the right of the diagonal need be calculated step-by-step.

The same subprogram is used to calculate the xs of Equations (39) as with Equation (25). The coefficients of Equation (1) and form (2) are then calculated using

$$C_o = x(1)$$

$$C_j = x(j+1) \qquad \text{when } M(j) = 0$$

$$\left.\begin{aligned} C_j &= \sqrt{x(j)1)^2 + x(j+2)^2} \\ \phi_{j+1} &= \arctan\left(\frac{x(j+2)}{x(j+1)}\right) \end{aligned}\right\} \quad \begin{aligned} &\text{when } M(j) = 1 \\ &\text{and } M(j+1) = 2 \end{aligned}$$

The method of calculating the two variance terms is discussed in F. A. Willer's book[1].

AUTOMATED DESIGN ENGINEERING

W. W. ROGERS

IBM

LOS ANGELES, CALIFORNIA

Gentlemen:

I would like to talk to you today about a new computer application, Automated Design Engineering.

A. D. E. is the use of the computer in the design of non-prototype products, the type of engineering commonly referred to as custom engineering, application engineering, or product engineering. Specifically, A. D. E. can benefit your company through significantly reducing your engineering lead time, increasing your engineering productivity, and decreasing your engineering costs. These and the other advantages of the application can, in turn, result in an improved competitive position for your company in your industry.

First, I would like to tell you what A. D. E. is. Second, I will discuss where A. D. E. applies. Third, I will spend a little time on how A. D. E. works. Fourth, I would like to cover in more detail the advantages of A. D. E.

Computers have been used, in the past, in the design of many products for industry. Computers have been utilized successfully in the design of circuits, missiles, motors, transformers, and telephone equipment, to name just a few. In each case, the use of the computer has resulted in great savings and great increases in efficiency and productivity for the companies involved. Now a newly developed computer application, Automated Design Engineering, vastly increases the power of the computer as an engineering tool. With A. D. E., the computer can now be used to solve more engineering problems on a wider range of products.

Conceptually, A. D. E. involves the storing of design logic in a computer so that the computer can accept customer orders as input and automatically generate complete designs. The input to the system would be the same type of orders you now receive from your customers. The completed design would contain the same type of information normally given by your engineers to your manufacturing department; such things as product characteristics, part numbers, assembly numbers, bills of material, purchases parts list, drawing numbers and so forth.

The principles of A. D. E. can best be described by using an example. An A. D. E. system for a company that manufactures pumps, for instance, would receive customer requirements in the form of orders or requests for bids. In this case, the customer needs a pump to pump 150^{o} Fahrenheit carbonic acid at the rate of 100 gallons per minute with a head of 40 feet and he wants the pump motor wired for 220 and 440 volts. These requirements would be entered into the computer and the completed design would be printed out.

These requirements would be entered into the computer, would be processed by the computer with the aid of the stored design logic, and the completed design would be produced. The completed design information can be in many different forms ... such as part numbers, assembly numbers, drawing numbers, manufacturing instructions, etc. In this example the completed design information consists of the pump frame number, FQ 6; the model number CXR; the suction and discharge pipe diameter, 3 and 4 inches, respectively; the impeller diameter, 8 3/4 inches; the motor speed, 1750 RPM; etc., etc. In short, the completed design would consist of the information needed by manufacturing to build the product.

Now that we have discussed what A. D. E. is, where does it apply? Automated Design Engineering applies primarily to "custom" or "product" design which can be defined as "non-prototype custom design variations of a standard product line to meet your customers' requirements on a continuing basis." Even though some of the tools, techniques, and methods available through A. D. E. are applicable to prototype design, the major impact of A. D. E. will be in the custom or product design area.

There are many companies in industry today which produce custom designed variations of a standard product line in response to customer orders or requests for bids. "Custom" or "product" design is common in such products as pumps, motors, generators, switch gear, transformers, electrical measuring equipment, industrial furnaces, switchboards, engineering and research instruments, heat exchangers, steam turbines, conveyors, and air compressers, to name just a few.

Let us look at the problem of the "custom" or "product" engineer. In industry today, customer orders for products will come to the design engineer. His problem is three-fold:

First, he must determine what to build to satisfy customer requirements.

Second, he must translate these customer requirements into workable parts and assemblies and their drawings.

Third, he must prepare the complete paperwork for manufacturing. It has been determined through actual experience in industry that these problems can be solved on a computer with great savings of time and money and great increases of efficiency and productivity.

Now to help you visualize what a system would look like in your company, let us look at a typical Automated Design Engineering System in operation. A customer order coming in would go to the Engineering Department where two functions would be performed. First, the order would be edited to insure the completeness and validity of the customer order. Second, any necessary pre-engineering would be performed. Once the order editing and pre-engineering has been completed, the customer order would go to the key punching room where the customer requirements would be transcribed onto cards and fed into the computer.

The computer is programmed with design logic and has available to it tables, standards, and reference information. The output of the system would be completed design which would then be reviewed by the engineering department and passed on to manufacturing. Now that we have covered what A. D. E. is, and where it applies, and what an operating system would look like, how does an Automated Design Engineering System actually work? The key to Automated Design Engineering is the ability to capture the design logic by which customer requirements are translated into product specifications.

We have developed new tools, new techniques and new methodologies which will assist you in developing an A. D. E. system for your company. One of the most important tools, and the heart of this new system, is Decision Tables, a technique for capturing the design logic that comprises custom or product design. The ability to capture design logic is the key to A. D. E. Before we discuss Decision Tables, let us clarify what we mean by design logic.

What is design logic. . . ? Design logic is the complex decision-making process through which the engineer proceeds in designing a product to meet a particular customer's requirements. The engineer reads the customer order and then, through the medium of design logic, proceeds to design and select the various parts and assemblies required to satisfy the customer's requirements. The result is the completed design of the product.

Let us look at a sample of design logic and how this design logic can be captured using decision tables. We will use as our example the armature for a voltmeter. If we were to ask an engineer working for a company that manufactures voltmeters how he designs the armature for a voltmeter, he might say: "Well, if the customer requires DC service for a speed application and asks for a single phase instrument with villivolt rating, then I know I have to use a moving coil. If two windings are needed, which frequently is the case, then the part number will be -12526A. Drawing number A26A will be used in this case. If the rating value specified is between 76 and 200 millivolts and we need a moving coil, the main winding will use Aluminum wire 16 mils in diameter. The number of turns and the number of layers will come from these two formulas which relate to each other. Based on the case we're using, we'll have 13 turns on the main winding and one layer. Again, I have to look up the drawing number . . . it's 012526-1A. Now the coil will also need a damper winding which for a 76 to 200 millivolt rating and a scale size specified as four inches, will take 8 mil Copper wire and half the number of turns on the main winding which is also shown on drawing number 012526-1A." This complex decision-making process can be captured by use of a new tool called decision tables.

A decision table consists of four quadrants. In the top two quadrants are the customer requirement names and values. In the lower two quadrants are product specifications, names and values. This decision table is a summary of part of the data given to us by the voltmeter design engineer. In the upper left hand quadrant is a list of customer requirement names. . . service, application, rating units and number of phases. Values which the customer might specify for these requirements are given in the upper right hand quadrant. . . DC, speed, millivolt, 1, etc. In the lower right hand quadrant are values for these product specifications. . . . . moving coil, inductive, 1 plus the number of phases, etc. The decision table is read as follows: If the service is DC, and if the application is temperature, then the type of armature needed is amoving coil, and tables number 2 is the table to which we should go next. Rule 4 would be read as follows.

If the service is AC, and if the rating units are milliamps, and if the number of phases is 1, then the type of armature required is inductive, and the number of windings is calculated as being ONE PLUS the number of phases, and the next table is number 2. Each different set of conditions which can exist is called a rule. In this case there are six feasible sets of conditions or six rules. Once the decision table is established, programmed and entered into the computer, the computer will automatically select which rule applies to each individual customer order.

It takes hundreds of decision tables to capture and store the design logic for an entire product or product line. When all of the required decision talbes have been programmed and placed in the computer memory, customer orders can be entered into the computer design in a logical stepwise manner.

Now that we have covered what A. D. E. is, where it applies, and have gone into a little detail on how it works, let us turn our attention to what an A. D. E. system can do for you, or the advantages of an A. D. E. system. We have already indicated that you can reduce your engineering costs and at the same time, obtain increased productivity and efficiency from your engineering force. Overall lead times can, in addition, be significantly reduced through a reduction of design engineering time, materials procurement time and manufacturing lead time. The design time, for example, can be reduced from weeks or even months to a matter of minutes. This reduction in design time will allow you to give faster response to customer orders and requests for bids. Faster response to bids, on the one hand, can have a significant effect on your profits if there is a high correlation, as exists in many industries, between speed of your response and acceptance of your bids. Faster response to orders, on the other hand, can result in faster deliveries and improved customer relations. And furthermore, Automated Design Engineering can provide a significant increase in your business activity by allowing you to respond to more bids if you now find yourself unable to respond to all of the requests for bids that are made to your company -- due to the lack of available time in your engineering department. In addition, faster response to bids can have a significant effect on your profits if there is a high correlation, as exists in many industries, between speed of your response and acceptance of your bids. Also, bid costs can be reduced because A. D. E. can reduce the out-of-pocket cost incurred when bids are designed but not won, since the engineering cost per bid can be greatly reduced.

Material savings are another potential advantage of Automated Design Engineering. By calculating exactly how much material is required for each job, A. D. E. can reduce overdesign and waste. Another way in which material savings can be realized is through reducing the number of parts in inventory with the same specifications but different part numbers. As you develop your A. D. E. system, these duplications will become readily apparent and can be eliminated. The consistent use of the best design practices eliminates the proliferation of methods and materials and insures a design commensurate with requirements. All too often today, the pressures of competition and of constantly shrinking delivery times tend to result in over-design or in picking a design which is perhaps more expensive than the specifications actually call for and the result is loss of profit. Bid and order costing can be incorporated into your Automated Design Engineering system to allow you to not only design but also to price both labor and material for your design. Through the automatic generation of engineering paperwork much of the clerical burden which wastes so much of the time and talent of engineers today can be alleviated.

Automated Design Engineering also provides an improved error-checking facility. The sources of errors in engineering today are many. These errors can creep in through the customer order, through the sales engineers, through engineering errors or material list preparation. Errors can result in the wrong material, too much material, or too little material being available at assembly points. They result in wrong design and in over-design. With an Automated Design Engineering System these functions are performed automatically. Error checking capabilities are built into the computer to catch and eliminate such miscalculations, which are the real problem in industry today. Greater management control is possible because management policies can be incorporated in an Automated Design Engineering System with the design logic, to insure that management policies on engineering are carried forward.

Now that we have discussed the advantages of A. D. E., I would like to show you how we can assist you in developing and implementing an Automated Design Engineering system in your company. IBM has available tools, techniques, and methods to help you in developing your system. There are two steps in the development of an operating Automated Design Engineering system. These are the Survey and the Implementation. The survey represents the first step. Its purpose is to determine the applicability of the A. D. E. system to your product lines. It will provide you with an analysis of your present system. It will determine the requirements of the new system and will give you a preliminary design of the new system as it can be applied specifically to your company. Lastly, upon completing the survey, you will be able to measure the advantages that will accrue to your company through the use of Automated Design Engineering. The new tools have been developed to prepare an analysis of your present design engineering operation from four related but different standpoints: Those of time, cost, accuracy, and operations. After the survey is completed you are ready for the next step, implementation. The purpose of the implementation phase is to develop a completely tested and operational Automated Design Engineering system. In order to do this, an analysis of your customer specifications, and an analysis of your product structure must be prepared. The techniques and tools which IBM will provide will also enab le you to capture the design logic of your product line onto decision tables, to perform the detailed systems design, the programming, the testing, the conversion and finally the initial operation of your Automated Design Engineering system. These tools, techniques and methods are tested and proven. They are available to you in the form of printed material. For example, the A. D. E. General Information Manual will introduce you to the survey and implementation phases of this new system. In addition, we have prepared a detailed reference manual to provide your engineers with the "How To Do It" information necessary to develop an A. D. E. system.

A PAYROLL AND LABOR DISTRIBUTION PROGRAM PACKAGE

by

ELIAS C. TONIAS

and

RICHARD C. DEVEREAUX

December 1963

ERDMAN & ANTHONY

82 St. Paul Street

Rochester 4, New York

70

TITLE — A Payroll and Labor Distribution Program Package

AUTHORS — Elias C. Tonias of Erdman & Anthony, Consulting Engineers, 82 St. Paul Street, Rochester 4, New York

and

Richard C. Devereaux of IBM Corporation, 540 Main Street East, Rochester 4, New York

DIRECT INQUIRIES TO — EliasC. Tonias

ABSTRACT

The objective of this paper is to demonstrate how some free 1620 time may be utilized in a relatively small scientific or engineering installation through the use of a package of commercial programs. The successful operation of such a program package since the first of this year (1963) has helped to justify the installation of a 1620 in another firm. This package, designed for use with the basic 20k 1620 Computer, with paper tape I/O and without any peripheral equipment, produces the payroll report, and complete labor cost distribution and breakdown reports. Written in FORTRAN for easy maintenance, the ideas from this program package might prove a worthwhile tool in justifying the installation, or in increasing the production ratio, of a small account. With a few or even no alterations, parts of the package may be used to handle other various time and cost distributions.

TABLE OF CONTENTS

I. INTRODUCTION

It is no secret that the 1620 is not designed for excessive data handling and the IBM Corporation definitely is not marketing it for commercial uses. In fact it is a small scientific computer and as its many users will attest, it performs with excellence in this role. Despite the machine's fine record in the field, its inability to handle excessive data (especially the basic 1620 with paper tape) produces one distinct problem area which can present itself during the selling phase or after the installation of the system. It is known as machine utilization.

Present day engineering or scientific establishments do have considerable, if not abundant, amounts of commerical (accounting) type of work which could be automated and thus increase the utilization of a computer. This problem of the 1620 utilization is not as predominant in a large data processing oriented company as it most probably has a commercial computer system. In such a company the 1620 is being looked upon for the engineering or research and development departments. A large company would presumably not expect high utilization on a machine which would be operated in an open shop atmosphere. Their objectives would be more intangible, such as the release of engineers for more creative type work and the increase in speed of routine computations.

On the other hand, in a small engineering account, where the rental cost of $1,600 per month represents a large investment, the subject of machine utilization is of great concern. Even though a two or three hour run a day can justify the installation of a 1620 in such an account there still is the feeling that more should be gotten for this kind of money. It is for such an establishment that this paper is primarily intended.

The company for which these programs were developed has a bi-weekly payroll for two offices, one of ninety and the other of forty employees. It takes about two days (16 hours) of the computer's time every other week to produce the complete payrolls and labor cost distribution. The operation of these programs since January 1, 1963 has helped to increase the average utilization of the computer to 40 hours per week. As for economic value of these programs, it cannot be measured in dollars and cents. It places valuable up-to-date information about the cost, progress and estimation of projects in the fingertips of management in a matter of a few hours. Neat reports may be produced in a moment's notice. It may be said that these programs in addition to increasing machine utilization do contribute in improving management operation.

II. SYSTEMS ANALYSIS AND DESIGN

In trying to analyze and successfully design a payroll and labor cost distribution program package the following must be taken into account: the memory capacity of the computer, the presence or absence of special features, the type of input-output and the method of operation of the shop including personnel availability. The concern where these Payroll and Labor Cost Distribution Program Packages were developed employed the basic 1620 with a 20k memory, without any special features, and with paper tape input-output hardware. This concern, a consulting engineering outfit, operates on a semi-open shop basis without a specialized machine operator. One cannot start any simpler than that nor can one have any more handicaps as far as the 1620 is concerned.

Of the three major languages of the 1620, machine, SPS, and Fortran, the latter was chosen by the authors because the programs could very easily be maintained and revised according to the needs of the company. In addition, Fortran offers a lot easier means of programming and debugging and to the best of the author's knowledge such a task would be a first. It must be noted here that all of the following programs (see exceptions later in this paper) were originally written in Fortran with Format and later changed to UTO Fortran. For those unfamiliar with UTO Fortran, it was developed by Mr. E. Stewart Lee and Mr. James A. Field of the University of Toronto, Ontario. UTO in general is similar to Fortran with Format except that it brings the program origin down to 06950, is slightly faster, the subroutines are about 10 per cent shorter, the use of EXECUTE PROCEDURE n statements (similar to CALL statements) facilitate programming and save memory when properly used, and has quite a flexible input format (input data does not have to be in strict accordance with the format statement).

One question that had to be answered early in the planning stages was the availability of checks for the 1620 typewriter. Cardboard checks were out because they necessitated a typewriter RPQ. Paper checks were chosen and designed by IBM. A picture of a check appears in the appendix of this paper.

The check is of the high-low design with the stub under the check proper. The perforation of the right can be placed any where. When designing a pay check the following hints should be considered:

(1) When designing the check, it is advisable to make the spacing between the printed lines a multiple of two or three. This means that at program time during the carriage control operations, the typewriter can be run in double or triple space mode and thereby save a considerable number of carriage returns.

(2) Hint (1) may be accomplished with statements of the form 1000 FORMAT (//) or if using UTO Fortran with the CONTROL 102 statement. In UTO the statement CONTROL 108 may be used for tabulation.

(3) The check should be designed so that variable alphameric data appears only on the left edge. Note the example check where the two dates (XX/XX/XX) and the name are all left adjusted. This is very important in a situation such as the following. The employees name is part of the master record and must be the first data in that record. For example, the name and year to date gross, FICA, and withholding taxes would read in as:

```
    ACCEPT TAPE 100, YGR, YFICA, YFWT
100 FORMAT (15H EMPLOYEE NAME F9.2,F7.2,F8.2)
```

The variable alphameric name is stored in the format statement itself. When saying PRINT 100, the name alone will be printed out. If the name were not left-adjusted in the master record and in the Format statement, it would be impossible to get it out alone. Another technic that could be used is to carry the name all by itself in a separate master record.

The payroll system for which these program packages were designed is based on the following criteria:

(a) Employees are to be paid bi-weekly with time being reported on two time sheets (the programs may easily be changed to any other pay system).

(b) Reported time is to be reported by each employee by operation (also referred as activity herein) for each project.

(c) There are three types of paying systems (pay codes): straight salary (code 1), straight time (code 2) and time and a half for overtime (code 3).

(d) There may not be more than 130 employees. This company uses 20 activities only. If need be this number may be increased at the expense of running time.

(e) Three master records are to be kept: (1)the payroll master tape containing the name, number, rate, pay code, dependents, year-to-date gross, FICA, tax withholdings and expenses, quarter-to-date gross, FICA, tax withholdings, social security number, hospitalization, bonds, savings, life insurance, pension, and miscellaneous deductions; (2) the project master tape containing the project number and todate totals on direct labor, expenses, and indirect labor (sick leave and vacation); (3) the activity Master tape containing to-date totals for each activity for each project.

As mentioned in the introduction the biggest handicap in this operation is that of handling data. Because of this, changes to any of the master tapes should be kept to a minimum as the slightest revision requires the reproduction of the entire tape. Master tape edit programs may be written to facilitate tape revisions, especially massive ones such as the zeroing of the registers at the end of a quarter or the year.

The other case where data handling enters the picture is in the data. This is discussed in the following section.

III. THE PROGRAMS

Below is a general description of the programs in the Payroll and Labor Cost Distribution Programs.

A. Data Preparation

Considerable thought was given to the input data since this was the start of a chain of events which would produce all of the payroll and labor distribution reports. As part of this paper's appendix one will find the time sheet used for reporting work time and expenses. Since this data form the basis to both packages it is typed into the Data Preparation program. The sequence of this data is composed of a minimum of three records per employee. The first contains the employee's number and the remaining contain the activity, number of hours or personal expenses and the number of the project. A negative activity denotes the end of a time sheet. For a bi-weekly payroll two negative activities are required. The program immediately punches on tape the accepted data in a similar format. This is the Labor Distribution Data Tape. Then it proceeds to compute the regular and overtime hours and personal expenses on a weekly basis which is stored in memory for each employee. This information will later produce the Payroll Data Tape. To do this the Payroll Master Tape containing the employees' pay codes is read into the program at the start of the run storing each employee's number and pay code. It should be mentioned here that activities 18, 19 and 20 are reserved by the programs for vacation, sick leave and personal expenses.

One bit of logic that might be helpful in the program is a technique for being able to go into memory and make corrections after all data is typed in and while the payroll data tape is still stored. This could be accomplished by assigning a storage cell number to each employee as he is entered into memory for the first time. This storage cell number would then represent the position in all four arrays (man number, regular hours, overtime hours, expenses) where this particular employee's data could be found. The number, along with a memory map of the arrays would then give the exact location of any data in question. A portion of such a table is included in the appendix. If an error were made in the typing of data for the say ninth employee the table will show that this employee's number, regular hours, overtime hours and personal expenses would be found in addresses 19750, 18450, 17150, and 15850 respectively (see table in appendix).

It will be advantageous in the Data Preparation Program to have a dump of grand totals at the end showing regular hours, overtime hours and personal expenses. These totals should balance back to an adding machine tape which should have been taken when auditing the time and expense sheets. The totaling routine should be programmed so that it may be "branched to" after any corrections to the memory table have been made. At this point if everything balances, the Labor Distribution Data Tape may be removed from the punch and the punching routine of the Payroll Data Tape initialized.

This marks the end of the combined operation between the Payroll and Labor Cost Distribution Programs.

B. Payroll Register

The first step in the Payroll Program Package is to obtain the payroll register showing all incomes and deductions for all employees of the current period. This, the Payroll Register Program, necessitates as input the Payroll Master Tape and the Payroll Data Tape as well as the starting check number and period ending date which are entered via the typewriter. The payroll data is stored in four arrays - employee number, regular hours, overtime hours and expenses. The Payroll Master Tape is then read in, the arrays are searched for a match, and the pertinent data from the master and detail are combined to calculate the net pay for each individual. Two lines of printed output appear for each employee showing the standard payroll register items. Accumulative totals are kept for each item and at the end of the report these totals are crossfooted for a check against the accumulated net pay.

In this program as well as in the next there is the problem of half adjusting and truncating a product. Let it be assumed that a straight time employee waged at $4.375 per hour has worked 48.50 hours. Hence

PAY = 4.375 x 48.50 = 212.18750

For obvious reasons the employee should be paid $212.19. Fortran however will carry it as $\bar{2}121875\bar{0}03$. When several such products are accumulated there is going to be a disagreement between the actual sums and those of Fortran. This may be remedied by the following statement:

PAY = RATE x HOURS + 0.005 + 100000.0 - 100000.0

which will be executed as follows step by step

PAY = 4.3750000 x 48.500000 = 212.18750
PAY = 212.18750 + 0.0050000 = 212.19250
PAY = 212.19250 + 100000.00 = 100212.19
PAY = 100212.19 - 100000.00 = 212.19000

When completed the payroll register should be spot checked by the personnel responsible to insure complete accuracy. There is no punched output to this program.

C. Payroll Checks

As soon as the payroll register is checked and accepted the Payroll Checks Program may be used to produce the paychecks. Basically this program is similar to the previous one using the same input (instead of the check number the check date is entered). In addition to the checks this program updates the Payroll Master Tape concurrently with the checks and at the end it produces an Adjusted Pay Rates tape. Such a rate is merely the employee's gross pay divided by the number of hours worked. This will affect only the straight salary and time and a half employees. For example, a straight salary employee who is always paid for 40 hours and has a rate of $4.375 has worked 48.50 hours. To give a more complete picture in the distribution reports all of the 48.50 hours should be considered. Thus this employee's adjusted pay rate is

Adjusted Pay Rate = 4.375 x 40.00/48.50 = 3.6082474

Had the employee being paid time and a half for overtime then his adjusted pay rate would be

Adjusted Pay Rate = (4.375 x 40.00 + (4.375 x 1.5) x 8.5)/48.50 = 4.7996134

This adjusted rate may later be used to distribute dollar increments of the 48.50 hours over the various projects on which the employee worked. This rate can be calculated at the same time as the employee's check and stored in memory. Since the Payroll Master tape is updated during the production of the checks the Adjusted Pay Rates will have to wait until the end of the program to be dumped on tape. These rates are later used with the labor cost distribution programs.

D. Payroll Deductions Register

Because of memory and output limitations not all deductions are itemized in the payroll register. These deductions, bonds, savings, hospitalization, etc. are itemized for each employee by the Payroll Deductions Register Program utilizing the just updated Payroll Master Tape.

E. Labor Distribution Data Sorting

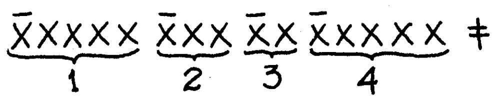
Having completed the payroll phase of the operation we may proceed with the labor cost distribution phase.

It was stated under the Data Preparation Program that data read directly into the program from the time sheets were divided into the Labor Distribution Data Tape and the Payroll Data Tape. Since it is possible for an employee to report time under several activities for several projects time data will have to be sorted in one way or another. Such a sort would be easy for a card machine. For a tape machine though it is a different story. Data could be sorted manually before the Data Preparation Program run but this would defeat payroll automation. As a matter of fact it would be a rather involved affair. Also mentioned earlier was the Project Master Tape containing all projects of the company with their respective costs.

It is possible for the operator to type an erroneous project number that is not included in the Project Master Tape or that a new project had been added without the master being updated. For these reasons two programs were developed in machine language to utilize memory space. These are the Project Completeness Test and The Labor Distribution Data Sorting programs.

The Project Completeness Test program first reads the Project Master Tape and stores the project numbers only; then it reads the Labor Distribution Data Tape and compares each project number with the table previously stored in memory. If a match does not occur the project number on the Labor Distribution Data Tape is printed on the typewriter. At the end of the run the number of projects and number of records are typed. The number of records refers to the records of the Labor Distribution Data Tape containing an activity, hours or expenses and a project number. When this test has been made the Labor Distribution Data Tape may be corrected if need be through a tape correction utility program.

The Labor Distribution Data Sorting Program does a little more than just sort data. First it counts records to insure that no unauthorized records have been dropped or added while correcting the Labor Distribution Data Tape. Second it sorts data. The program inserts the employee number to each of his records, changes each record from an alphameric mode to numeric mode dropping all decimals and arranging them in the format shown below:



where 1: project number
2: employee number
3: activity number
4: hours or personal expenses

Each record contains a total of 16 characters including the record mark. These records are stored in memory beginning in location 19999 and working on downwards. The sorting of the records begins when all records have been stored. It is a simple replaceable sort routine which starts at the top of memory and keeps comparing two adjacent storage cells (records). If the upper is less than or equal to the lower, no replace takes place. If the upper is greater than the lower, the two records are interchanged, a switch is set and the program steps down one record to compare the next two. This continues until one complete pass on the records has been made. At this point, the switch is checked. If it is on (meaning at least one record out of sequence), it is turned off and the comparing routine is repeated. When the switch is finally off after a complete pass, the internal sort is considered terminated. It should be emphasized here that during the compression phase (changing data from alphameric to numeric) where data is edited in the input area and placed in a cell, the comparing field can be set up as one field and still get three breakdown levels - activity, within man, within project - providing the field is arranged as described in the previous paragraph.

Third, at the completion of the sorting routine the program will commence reading the adjusted pay rates which were produced at the end of the payroll check production. The rates and employee numbers are converted into numeric characters and the memory cells are searched for an employee number match. When the match occurs the rate is multiplied by the corresponding hours (no multiplication is performed for activity 20 - personal expenses) and the hours are now replaced by the cost. Each employee number of the Adjusted Pay Rates tape searches the entire memory map as it is possible for one employee to appear in several records.

Fourth, when all hours have been convered into costs the program will type the total direct and indirect labor cost which should agree with that of the payroll register. It is obvious that the adjusted pay rates of personnel other than those of straight time will not be the same with their original base pay. These adjusted pay rates may contain as many as eight significant digits. This problem is similar to that of the payroll register and payroll checks of rounding or truncating products. To compensate for this each product (rate times hours) is half adjusted to the nearest penny and all truncated parts are accumulated to form the truncation or rounding error. This which may be negative is assigned to project one, administration, to activity one, general, and to employee zero, fictitious. Hence the gross payroll figure plus the rounding error of this program should equal that of the payroll register.

Fifth and last the fields of each record are changed back to alphameric and are punched on tape to produce the Sorted Labor Distribution Data Tape.

F. Labor Cost Distribution

This program is essentially a listing with some accumulating of the sorted labor data tape. This tape is now in project sequence. Within each project are the records of each man who worked on that project during the last pay period. Within each man are the operations which he performed on that project as well as the distributed money. There are actually three phases to this program which give three separate reports.

The first report shows by employee number all the employees who worked on each project this current period. All the activities for each employee are accumulated and this accumulated money total is segregated into direct and indirect labor and personal expenses.

The second report of this program is merely a listing of the totals lines for each project. It gives a neat, condensed, summary report of what the projects did this period. This is accomplished by storing project number, total direct labor, total expenses and total indirect labor for each project while producing the first report.

The third report is the project status report showing all projects of the company whether work was done or not on all of them. For this the Project Master Tape is used. As the master is read in the project number from the tape searches the memory (see previous paragraph) for a match. If a match does occur the figures are combined and the master is updated. If no match occurs (no work done on this project this week) the figures remain the same. There may be printout options at this point. One which the user would probably want is a project-to-date report showing three totals for each project. Line 1 shows project totals to-last period, line 2 shows project totals of this period (zeroes may be present) and line 3 shows the updated project totals (sum of lines 1 and 2).

A point of interest to this program as well as to the last of this series is that of the grand totals appearing at the end of the various reports. Fortran handles only eight significan figures thus limiting totals to $999,999.99. If the elements of a total added to more than a million dollars the machine computed total would be a truncated figure sometimes several dollars off the actual sum. This problem may be eliminated using Fortran in the following manner. Let SUM represent the accumulative value of an item and COST the value of any one element of the same item so that

$$\text{SUM} = \sum \text{COST}$$

Before incrementing SUM by COST, COST is compared with the difference between 999,999.99 and SUM. If the cost is smaller than the difference the addition is performed. If not a carry-over factor is incremented by one and the sum reinitialized by the difference between COST and the first difference. Thus in UTO Fortran a procedure could be set up as follows:

```
    BEGIN PROCEDURE 100
    DIF = 999999.99 - COST
    IF(DIF-COST) 111, 110, 110
110 SUM = SUM + COST
    RETURN 100
111 KARRY = KARRY + 1
    SUM = COST - DIF - 0.01
    END PROCEDURE 100
```

where KARRY indicates the millionth carry-over factor. Note the penny (0.01) in the last statement of the procedure to take care of the difference between a million and 999999.99.

G. Labor Cost Breakdown - Bi weekly

This program merely rearranges data from the Sorted Labor Distribution Data Tape. That is, it reads in all the details for a project number, accumulates the amounts by activity number, and punches out a new tape, the Labor Cost Breakdown Tape, which contains the project number followed by each of the operation numbers and their respective monies. This is done for the entire input tape, so that the output tape contains all the project numbers followed by all the activity numbers and amounts. This program generates a new activity code number for each project Activity 21 representing an accumulative total of all the money from all the other operations on that project.

There is also an optional print-out to this program. Any or all projects may be printed showing money spent by each employee under each activity for a project. Because of space limitations and the extend of the arrays only eight employees are to be included on a page. Thus several pages may be needed for one project.

H. Labor Cost Breakdown - Accumulative

This is the last of the Labor Cost Distribution series of programs. As one may recall the third report of the Labor Cost Distribution Program updates a Project Master Tape showing labor breakdown by direct labor, personal expenses and indirect labor. This last program gives the labor breakdown by operation within project. Because there are 21 possible operation under each project, the Activity Master Tape is actually composed of three reels of tape. The first carries all the projects with activities 1 through 7. The second carries all projects with activities 8 through 14 and the third carries all projects with activities 15 through 21. As a reminder activity 21 represents the total cost of activities 1 through 20 inclusive.

This program is executed in three passes. At the beginning of each pass the Labor Cost Breakdown Tape is loaded and the program selects and stores activities 1 - 7, 8 - 14 and 15 - 21 depending whether it is pass one, two or three. Having stored this information the corresponding Activity Master Tape is passed printing the breakdown report and punching the updated Activity Master Tape. The report may take a similar form to that of the project status mentioned under the Labor Cost Distribution Program. The company that developed these programs however chose to eliminate from this report the top line showing project-to-last period totals.

This final report is an extremely comprehensive breakdown of the allocation of funds on the various projects. When a project is completed, the final master records become an invaluable reference to be used in the future for estimating costs of similar jobs and in a way to predict or forecast job progress and man power alocation.

IV. TIMING CONSIDERATIONS AND LIMITATIONS

Below is a list of limitations to these programs:

(a) The number of employees is limited to 130 in any one run. Slight variations in paying systems and in state or local taxations may increase or decrease available memory and thus introduce new limitations on the number of employees.

(b) In the Labor Distribution Data Sorting Program there is room for 1048 records.

(c) An increase in the number of activities will increase the number of passes in the Labor Cost Breakdown Accumulative Program.

(d) The following timings are based on 90 employees working on 35 projects at any one time and reporting their time in about 500 records. The total number of projects being reported on the distribution and breakdown reports are 70. It should be kept in mind that these figures of time are for a bi-weekly payroll.

1. Time and expense sheets are turned in by Friday night.

2. Time and expense sheets are audited Monday morning.

3. Revisions and adjustments to the master tapes .... 1.00 hours

4. Data Preparation Program .... 2.50 "

5. Payroll Register Program .... 0.75 "

6. Payroll Checks Program .... 2.25 "

7. Payroll Deductions Register Program .... 0.50 "

8. Labor Distribution Data Sorting .... 0.75 "

9. Labor Cost Distribution Program

   Report #1 .... 1.00 "

   Report #2 .... 0.25 "

   Report #3 (project status) .... 0.75 "

10. Labor Cost Breakdown - Bi weekly without any print-out .... 0.50 "
(allow 5 minutes per project print-outs)

11. Labor Cost Breakdown - Accumulative .... 2.25 "

12. Average minimum total ....12.50 "

The secret to a smooth and fast run is to avoid last minute changes, and to train the employees to report their time and expenses properly thus eliminating corrections and revisions while a $1600 a month machine is running doing nothing but tape corrections.

V. SUMMARY

It was the purpose of this paper to present guide lines and documentation for implementing a payroll and labor distribution application on a basic paper tape 1620. It takes but a little imagination to develop additional utility programs to handle information developed by these programs to prepare special reports such as Employee Taxable Wage Reports, W-2 Form Information (to the best of the author's knowledge there are no W-2 forms that will fit the basic 1620 typewriter), Sick Leave and Vacation Tally Reports and a number of others. With slight if any at all modification the distribution programs may be used to handle other distributions unrelated to the payroll.

These programs have supplied their developers with up-to-the-minute information on the financial status of all their projects - information which in the past was delinquent and incomplete. It is the authors' belief that such programs as these presented in this paper although not money making will provide a user with invaluable service.

VI. APPENDIX

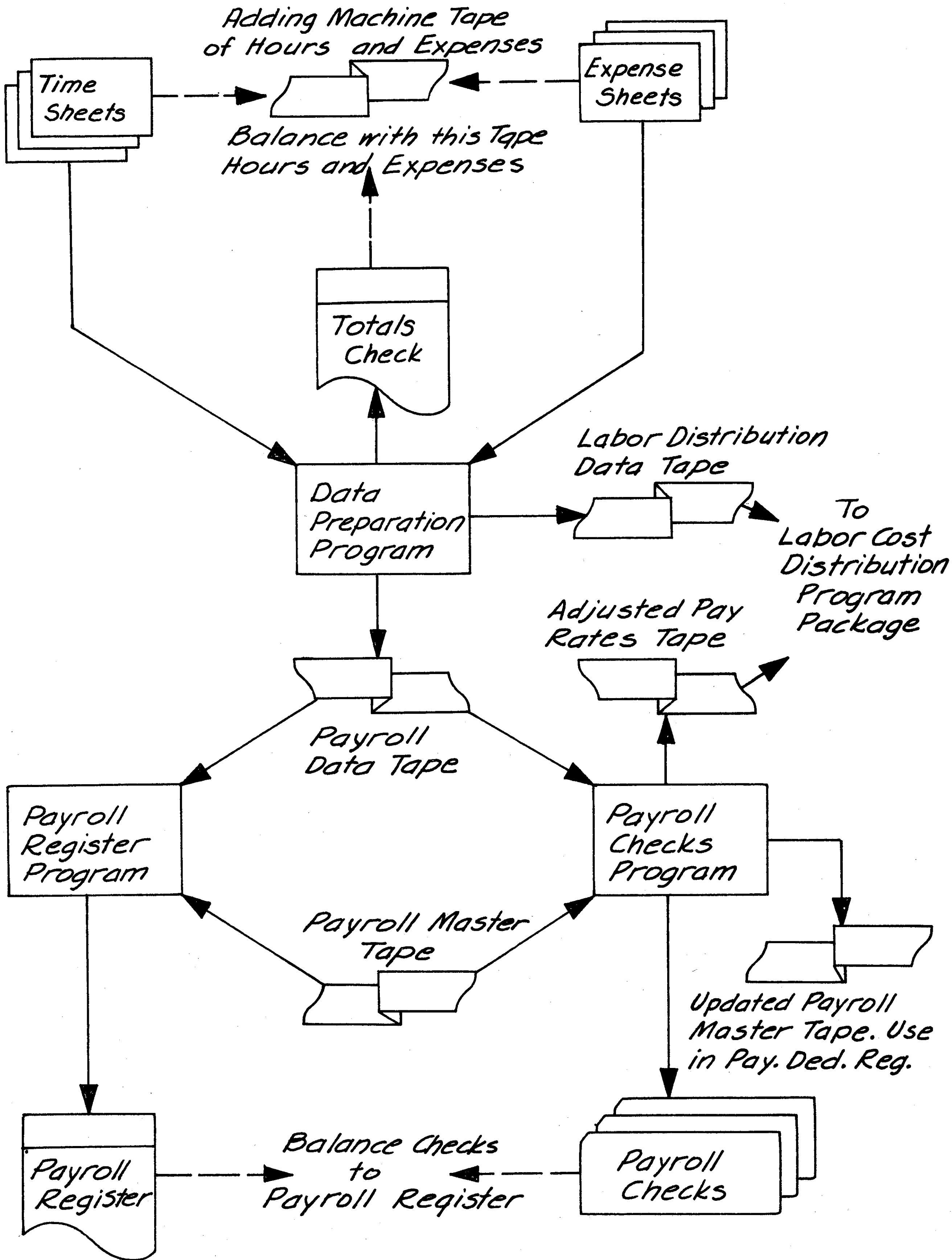
Included in the appendix to this paper is a general flow chart of both program packages and input-output samples of their most important phases.

The back side of the time sheet is not used in either of the packages. In the register of payroll deductions the last column represents the horizontal sums of the other columns. Note that the grand total is obtained twice; one from the vertical and once from the horizontal sums.

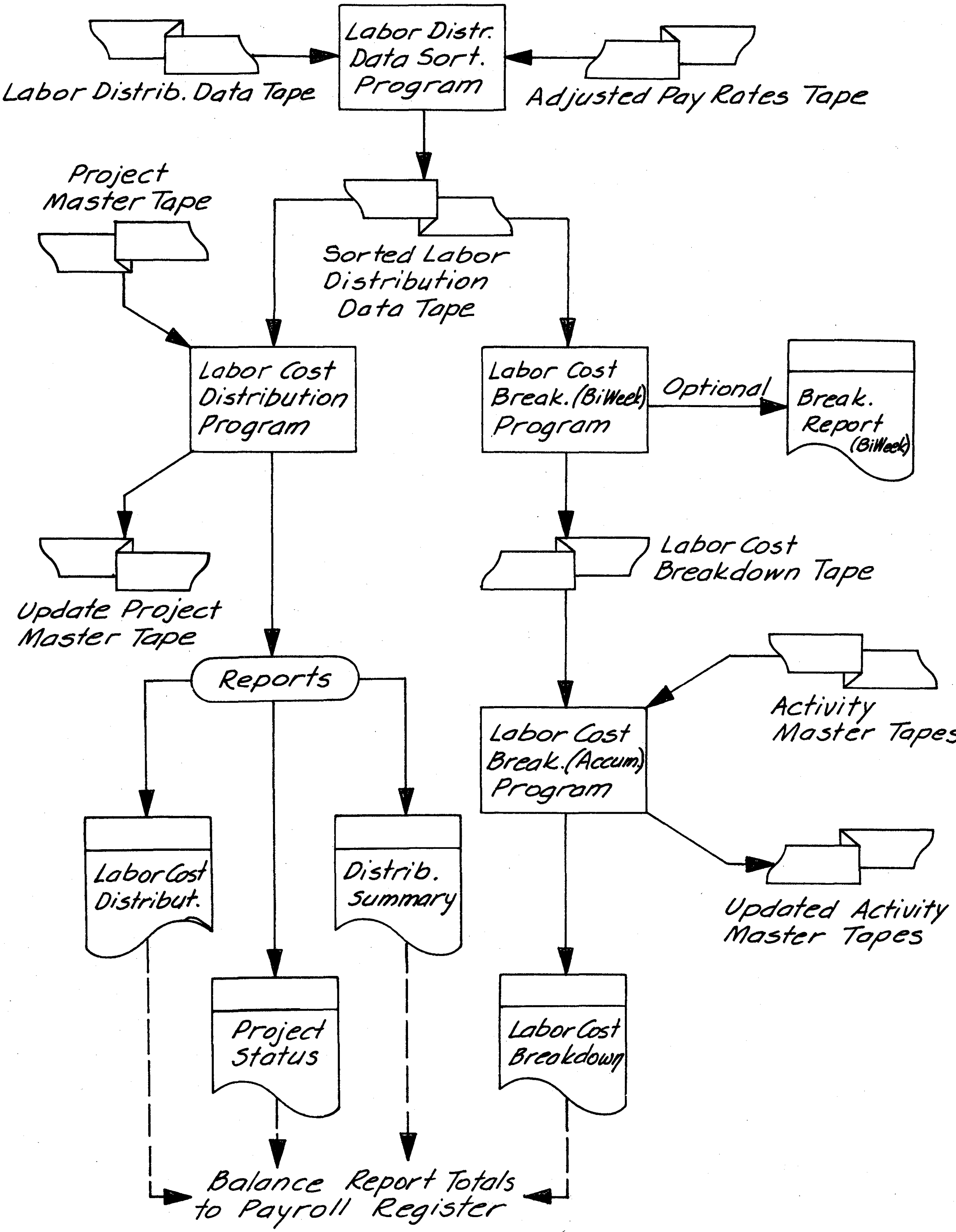
Two projects of the labor distribution report are shown on the same page. Actually they would appear each on a separate sheet the carriage being controlled by the program. Totals of the project summary are obtained at the end but are not shown in the sample. The project status at the bottom shows a sub-total and a total. The developer of the programs wanted to divide the projects into two groups each having a subtotal. The subtotal shown is that of the second group of projects.

The bi-weekly Labor Cost Breakdown report shows a partial listing of project No. 61151. There are only 16 of the 20 activities shown. The number across the top indicate the employee numbers. Had there been more than eight employees a second or even more sheets would have been needed. In such a case the column headed ACC would indicate the accumulative totals to that sheet. The last display shows a partial Accumulative Labor Cost Breakdown report. The numbers 15 - 21 across the top indicate the activity numbers. There are actually two more sheets to this report for activities 1 - 7 and 8 - 14. Note that activity 21 is the sum of the first 20 activities.

GENERAL FLOW CHART
~ PAYROLL ~



GENERAL FLOW CHART
~ LABOR COST DISTRIBUTION ~



ERDMAN & ANTHONY
DAILY TIME REPORT

LOCATION: Rochester, N.Y.
NAME: John J. Johns #1
WEEK ENDING: 11-15-63
TITLE: Draftsman

OPER. NO.	S	S	M	T	W	T	F	TOTAL HOURS	PROJECT NO.
1			8					8	2214.
3				8	8	8		24	2214.
13						2	8	10	1114.
			8	8	8	10	8	42	

I HEREBY CERTIFY that the time reported above is a true and complete statement of my working time for this period.

John J. Johns
Employee Signature   Appd. ____ 1 | 2

88

ERDMAN & ANTHONY
DAILY TIME REPORT

LOCATION: Rochester, N.Y.
NAME: John J. Johns #1
WEEK ENDING: 11-22-63
TITLE: Draftsman

OPER. NO.	S	S	M	T	W	T	F	TOTAL HOURS	PROJECT NO.
3	8		8	8				24	2214.
12						4	4	8	2201.
13					8			8	1114.
18							4	4	
19						4		4	

I HEREBY CERTIFY that the time reported above is a true and complete statement of my working time for this period.

John J. Johns
Empoyee Signature   Appd. ____ 1 | 2

91

(ATTACH RECEIPTS WHEN OBTAINABLE FOR ALL ITEMS)

EXPENSE ACCOUNT 2 Weeks Ending 11-22-63	ERDMAN & ANTHONY Rochester, New York	SHEET OF	CKD BY

(Please print)
Statement of KENETH W. BROWN #2

Mailing Address

Date	Day	Brief statement of itinerary, etc., showing nature of expense items listed in columns right	MI	Mileage and Transp.	Lodging	Meals	Other (Itemize)	Daily Total
	S							
	S							
11/11/63	M	Job Inspection 61151.		4.00	6.50	7.50		18.00
11/12/63	T	Job Inspection 61151.		4.00				4.00
	W							
	T							
	F							
	S							
	S							
	M							
11/19/63	T	Job Inspection 60151.		4.00		3.00		7.00
11/20/63	W	supplies 22-14.					2.28	2.28
	T							
	F							
				12.00	6.50	10.50	2.28	31.28

I HEREBY CERTIFY that this is a true and complete statement of authorized expenses incurred by me in connection with company business for this payroll period.

Employee Keneth W. Brown

Approved

69

ERDMAN AND ANTHONY, CONSULTING ENGINEERS
PAYROLL MASTER TAPE

11/22/63

Quarterly Registers

Empl #	Pay Rate	Depend.	Deductions: Bonds	Savings	Hosp.	Misc.	Pay Code	Income Gross	Withholdings: FICA	Federal	State
1	2.250	0	.00	25.00	.00	.00	3	720.00	26.12	129.60	15.92

Annual Registers

Employee Name	Bogus #	Income Gross	Withholdings: FICA	Federal	State	Personal Expens.
JOHN J. JOHNS		4230.00	153.45	761.40	95.29	.00

Soc. Sec. #	Pension	Life Ins.	N.Y.S. Disab.
120 30 1404	.00	.00	.60

```
  2  4.188 3   .00   .00   .00   .00 1 1340.16    .00 185.08  28.56
KENETH W. BROWN          .             7705.92 174.00 1064.21  164.22  438.29
 94  1 3346   1.00     5.50 .60

  3  3.250 4   .00   .00  9.40   .01 2 1261.00    .00 152.10  21.61
MARY BENSON              .             6909.45 174.00  813.16  108.81  515.68
 126 34 9929   .00      .00 .60

 999  .000 0   .00   .00   .00   .00 0    .00     .00    .00    .00
```

90

18

.2

11

TABLE OF ADDRESSES

The addresses shown in the table represent the address of the left most position of each variable.

ENTRY NUMB.	MAN NUMB.	REG. HOURS	OVT. HOURS	PERS. EXP.	ENTRY NUMB.	MAN NUMB.	REG. HOURS	OVT. HOURS	PERS. EXP.
1	19830	18530	17230	15930	23	19610	18310	17010	15710
2	19820	18520	17220	15920	24	19600	18300	17000	15700
3	19810	18510	17210	15910	25	19590	18290	16990	15690
4	19800	18500	17200	15900	26	19580	18280	16980	15680
5	19790	18490	17190	15890	27	19570	18270	16970	15670
6	19780	18480	17180	15880	28	19560	18260	16960	15660
7	19770	18470	17170	15870	29	19550	18250	16950	15650
8	19760	18460	17160	15860	30	19540	18240	16940	15640
9	19750	18450	17150	15850	31	19530	18230	16930	15630
10	19740	18440	17140	15840	32	19520	18220	16920	15620
11	19730	18430	17130	15830	33	19510	18210	16910	15610
12	19720	18420	17120	15820	34	19500	18200	16900	15600
13	19710	18410	17110	15810	35	19490	18190	16890	15590
14	19700	18400	17100	15800	36	19480	18180	16880	15580
15	19690	18390	17090	15790	37	19470	18170	16870	15570
16	19680	18380	17080	15780	38	19460	18160	16860	15560
17	19670	18370	17070	15770	39	19450	18150	16850	15550
18	19660	18360	17060	15760	40	19440	18140	16840	15540
19	19650	18350	17050	15750	41	19430	18130	16830	15530
20	19640	18340	17040	15740	42	19420	18120	16820	15520
21	19630	18330	17030	15730	43	19410	18110	16810	15510
22	19620	18320	17020	15720	44	19400	18100	16800	15500

11/22/63 24.

ERDMAN AND ANTHONY, CONSULTING ENGINEERS

PAYROLL REGISTER

	REG.H	OVT.H	REG.P	OVT.P	GROSS	FICA	FWT	SWT	DED	EXP	NET
JOHN J. JOHNS 11858.					4443.75						
1	80.00	10.00	180.00	33.75	213.75	7.75	38.48	5.29	25.60	.00	136.63
KENETH W. BROWN 11859.					8040.96						
2	80.00	25.00	335.04	.00	335.04	.00	46.27	7.14	6.10	31.28	306.81
MARY BENSON 11860.					7169.45						
3	80.00	5.00	260.00	.00	260.00	.00	28.08	3.13	10.01	6.34	225.12

	REG.H	OVT.H	REG.P	OVT.P	GROSS	FICA	FWT	SWT	DED	EXP	NET
TOT.	240.00	40.00	775.04	33.75	808.79	7.75	112.83	15.56	41.71	37.62	668.56

92

20

50-18
223

ERDMAN AND ANTHONY
CONSULTING ENGINEERS
ROCHESTER, N. Y.

DATE 11/29/63 24 CHECK NO. 11858

PAY TO THE ORDER OF JOHN J. JOHNS 1.

SAMPLE CHECK

PAY → DOLLARS CENTS 136.63

PAYROLL ACCOUNT
EAST AVE. BRANCH
TO GENESEE VALLEY UNION TRUST CO.
ROCHESTER, N. Y.

OFFICIAL SIGNATURES

⑆0223⑉0018⑆ 510⑈81863⑈3⑈

93

STATEMENT OF EARNINGS AND DEDUCTIONS

REGULAR HRS.	OVERTIME HRS.	RATE	REGULAR PAY	OVERTIME PAY	GROSS PAY		EXPENSES
80.00	10.00	2.250	180.00	33.75	213.75		.00

FED. TAX	F.I.C.A.	STATE TAX	HOSP. INS.	BONDS	SAVINGS	PENSION	N.Y. DISA.	MISC.	NET PAY
38.48	7.75	5.29	.00	.00	25.00	.00	.60	.00	136.63

YEAR-TO-DATE TOTALS

PER. END. DATE	GROSS PAY	FED. TAX	F.I.C.A.	STATE TAX		PENSION
11/22/63	4443.75	799.88	161.20	100.58		

21

ERDMAN AND ANTHONY, CONSULTING ENGINEERS

REGISTER OF PAYROLL DEDUCTIONS

PERIOD ENDING FRIDAY
NOVEMBER 22, 1963

	BONDS	SAVINGS	HOSPITAL-LIZATION	MISC DEDUCTIONS	DISABILITY	GROUP INSURANCE	
JOHN J. JOHNS 1.	.00	25.00	.00	.00	.60	.00	25.60
KENETH W. BROWN 2.	.00	.00	.00	.00	.60	5.50	6.10
MARY BENSON 3.	.00	.00	9.40	.01	.60	.00	10.01
	.00	25.00	9.40	.01	1.80	5.50	41.71

41.71

94

22

ERDMAN AND ANTHONY, CONSULTING ENGINEERS

PROJECT NO. 1.

9/27/63

	LABOR	EXPENSES	S.+V.	TOTALS	
0	-.06	.00	.00	-.06	
1	180.00	.00	.00	180.00	
37	360.00	.00	.00	360.00	
52	147.25	5.70	.00	152.95	
62	39.51	.00	.00	39.51	
63	132.85	.00	.00	132.85	
97	115.15	.00	.00	115.15	
	974.70	5.70	.00	980.40	980.40

ERDMAN AND ANTHONY, CONSULTING ENGINEERS

PROJECT NO. 1114.

9/27/63

	LABOR	EXPENSES	S.+V.	TOTALS	
10	350.00	.00	.00	350.00	
22	216.00	.00	.00	216.00	
35	42.25	.00	.00	42.25	
38	27.00	.00	27.00	54.00	
45	25.00	.00	.00	25.00	
60	107.25	.00	.00	107.25	
80	8.14	.00	.00	8.14	
	775.64	.00	27.00	802.64	802.64

LABOR COST DISTRIBUTION SUMMARY

ERDMAN AND ANTHONY, CONSULTING ENGINEERS

9/27/63

	LABOR	EXPENSES	S.+V.	TOTALS
1.	974.70	5.70	.00	980.40
2.	444.01	.00	.00	444.01
1114.	775.64	.00	27.00	802.64
1115.	117.00	.00	.00	117.00
1152.	1406.88	45.14	.00	1452.02
2201.	778.82	22.37	140.00	941.19
2202.	140.02	.00	.00	140.02
2204.	591.50	.00	270.73	862.23
2205.	1038.36	.00	77.00	1115.36
2206.	295.25	.00	.00	295.25
2208.	325.00	.00	44.00	369.00
2212.	23.75	1.86	.00	25.61
2213.	1292.64	.00	53.25	1345.89
2214.	3318.08	516.17	.00	3834.25
2215.	257.38	.00	27.00	284.38
2218.	.00	9.55	.00	9.55
2221.	165.19	5.55	.00	170.74
2222.	180.73	.00	.00	180.73
2223.	32.55	.00	.00	32.55
3102.	21.25	.00	50.63	71.88
3107.	729.78	102.59	40.63	873.00
3108.	4.25	.00	.00	4.25
59105.	357.16	11.40	.00	368.56
60107.	111.80	.00	.00	111.80
60151.	1666.80	61.39	10.00	1738.19
61102.	1556.76	.00	164.00	1720.76

96

PROJECT STATUS

ERDMAN AND ANTHONY, CONSULTING ENGINEERS

10/11/63

	LABOR	EXPENSES	S.÷\.	TOTALS	
2203.	538.20	.00	.00	538.20	
	.00	.00	.00	.00	
	538.20	.00	.00	538.20	
62208.	2206.13	172.94	128.50	2507.57	
	.00	.00	.00	.00	
	2206.13	172.94	128.50	2507.57	
62213.	4293.42	544.00	180.89	5018.31	
	.00	.00	.00	.00	
	4293.42	544.00	180.89	5018.31	
3101.	313.31	44.45	.00	357.76	
	.00	.00	.00	.00	
	313.31	44.45	.00	357.76	
3102.	2053.95	160.24	50.63	2264.82	
	1384.89	27.92	9.00	1421.81	
	3438.84	188.16	59.63	3686.63	
3103.	1179.89	74.33	76.00	1330.22	
	.00	.00	.00	.00	
	1179.89	74.33	76.00	1330.22	
3104.	61.88	.00	.00	61.88	
	.00	.00	.00	.00	
	61.88	.00	.00	61.88	
3105.	760.47	198.40	55.93	1014.80	
	.00	.00	.00	.00	
	760.47	198.40	55.93	1014.80	
3106.	11838.01	2915.55	112.54	14866.10	
	.00	.00	.00	.00	
	11838.01	2915.55	112.54	14866.10	
3107.	746.28	102.59	40.63	889.50	
	50.38	.00	.00	50.38	
	796.66	102.59	40.63	939.88	
3108.	4.25	.00	.00	4.25	
	.00	.00	.00	.00	
	4.25	.00	.00	4.25	
62204.	2484.90	24.15	214.74	2723.79	
	48.00	.00	.00	48.00	
	2532.90	24.15	214.74	2771.79	
	1 520352.50	46879.50	58380.27	1 625612.27	1 625612.27
	17667.45	745.53	564.11	18977.09	18977.09
	1 538019.95	47625.03	58944.38	1 644589.36	1 644589[illegible]
	1 556023.86	48152.42	60548.30	1 664724.58	1 664724.58
	18998.65	745.53	564.11	20308.29	20308.29
	1 575022.51	48897.95	61112.41	1 685032.87	1 685032.87

ERDMAN AND ANTHONY, CONSULTING ENGINEERS

PROJECT NO. 61151.

6/21/63

LABOR COST BREAKDOWN (BI-WEEKLY)

	3	6	13	21	56	0	0	0	TOT	ACC
1	50.00	.00	46.12	14.25	.00	.00	.00	.00	110.37	110.37
2	.00	.00	.00	.00	7.22	.00	.00	.00	7.22	7.22
3	.00	.00	4.85	.00	.00	.00	.00	.00	4.85	4.85
4	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
5	9.38	5.00	4.85	9.50	.00	.00	.00	.00	28.73	28.73
6	.00	5.00	.00	.00	.00	.00	.00	.00	5.00	5.00
7	209.38	75.00	82.53	57.00	55.34	.00	.00	.00	479.25	479.25
8	.00	5.00	.00	.00	19.25	.00	.00	.00	24.25	24.25
9	.00	.00	131.08	128.25	163.61	.00	.00	.00	422.94	422.94
10	37.50	.00	.00	.00	.00	.00	.00	.00	37.50	37.50
11	.00	.00	.00	4.75	.00	.00	.00	.00	4.75	4.75
12	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
13	.00	5.00	.00	.00	14.44	.00	.00	.00	19.44	19.44
14	56.26	.00	.00	14.25	.00	.00	.00	.00	70.51	70.51
15	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
16	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00

26

98

LABOR COST BREAKDOWN (ACCUMULATIVE) 27.

ERDMAN AND ANTHONY, CONSULTING ENGINEERS

11/08/63

PROJECT	15	16	17	18	19	20	21
1.	.00	.00	.00	.00	.00	.00	928.13
	.00	.00	.00	1364.42	178.46	1263.67	31588.98
2.	.00	.00	.00	4.50	.00	3.25	722.00
	.00	.00	.00	435.40	194.25	15.75	11919.71
58153.	.00	.00	.00	.00	.00	.00	.00
	21014.22	.00	.00	.00	.00	.00	21014.22
60151.	75.00	.00	.00	.00	40.00	13.27	1463.52
	4617.04	92.10	.00	2241.20	1862.66	3437.16	133755.23
5701.	.00	.00	.00	.00	.00	.00	.00
	37761.13	.00	.00	.00	.00	.00	37761.13
58101.	.00	.00	.00	.00	.00	.00	.00
	22737.93	38417.74	.00	.00	.00	.00	269471.35
58104.	.00	.00	.00	.00	.00	.00	.00
	10393.06	4019.15	.00	14.00	.00	.00	88158.71
59105.	.00	.00	.00	.00	.00	.00	235.29
	2806.04	117.75	244.50	1500.36	1513.14	310.76	122273.57
59103.	.00	.00	.00	.00	.00	.00	.00
	27.52	.00	.00	.00	.00	5.85	33.37

99

SPIRE

(System for Personnel Information REtrieval)

Prepared by:

G. J. Reed
Albuquerque Division
ACF Industries, Inc.
December 9, 1963

100

I. INTRODUCTION

The management of ACF Industries, Incorporated realized that manual methods for accessing personnel information were inadequate both with respect to speed and accuracy. Many requests for summarized information either could not be answered or became a costly venture in man hours alone. As an attempt to remedy this problem, a punched card system was designed to hold personnel information in coded form. This card system proved inadequate in speed, type and amount of information stored, and updating capability. With this system as a background, SPIRE was developed.

SPIRE is an acronym for "System for Personnel Information REtrieval." This system was designed for and implemented on a magnetic tape 60K IBM 1620 and has been operational since February, 1963. Since that date, no revisions have been necessary in the type or format of the information.

II. CONTENT

The information content of SPIRE was specifically designed for ACF's use. Personnel information was carefully analyzed and organized to determine what information had been recorded by the previous system and what part of this information had been useful. By categorizing the useful information, it was possible to devise coding techniques which made each item unique in the system and thus recoverable.

The information in SPIRE is broken into the following five categories:

1. Personal — name, date of birth, sex, date of hire, marital status, social security number, etc.

2. Previous Experience — job code, position title, time on job, extent of supervision, etc.

3. ACF Experience — complete file on all pertinent happenings at ACF with the employees status (job title, grade, salary, department, etc.) recorded at the time of the action.

4. Education — includes all completed levels of education and the extent, major, minor, and university, if applicable, for each level.

5. Military Experience — contains the same information as the previous experience category with the branch of service added.

To obtain this information in coded form for all employees, supervisors interviewed their subordinates and coded their education and experience backgrounds. The ACF experience involved so much detail that it was necessary to code each part of the employee's experience using the permanent personnel records. As new employees are hired, their backgrounds are coded and added to the master file.

III. APPLICATIONS

Although SPIRE was developed as a general purpose information system that could be used for the generation of virtually any report involving personnel information, one application appears to be the most notable, that of in-plant recruiting. Prior to the development of SPIRE, to fill a vacant position from within required the personnel in Salary Administration to have an intimate knowledge of the background of all salaried personnel. This, of course, became a virtually impossible task as the number of employees increased. Qualified people could easily have been overlooked, and the preparation of readable resumes for the managers became a monumental task. SPIRE now assures that no qualified individual is overlooked and automatically prepares their resumes. This resume, as printed, contains virtually no coded information, thus providing management with a concise, complete, easily read and understood document on any salaried employee.

Of course, qualification is a matter of definition. A professional analyst takes the vacant position and determines the background necessary to qualify an individual. The analyst specifies acceptable work experience and the minimum years of experience in related fields. He also specifies the acceptable areas of education and the minimum acceptable level. From the years of experience and the education level specified, a minimum number of "points" are computed by the system which will qualify an individual for the vacant position. Individuals may accrue the necessary points in any combination of related experience and education. Thus, if an individual had a masters degree in an applicable field and only a bachelors was required, he could qualify for the position with less experience than that specified. The equating of education and experience used in the SPIRE system is based on the ACF position rating plan. The analyst also specifies the acceptable salary grades for the search. This keeps senior men from "dropping out" for junior jobs for which they would be qualified. In addition to the preciseness with which this search for qualified employees is conducted, a considerable savings in both time and money exists.

The normal time to complete a search for any given position varies between six and eight minutes. This same job accomplished by manual methods would take about six man hours.

Another application which has saved a considerable number of man days is the generation of the quarterly Merit Review Report. This report is by department and contains pertinent information concerning past salary increases for all employees who are to be reviewed during the next quarter. The normal time to complete this run is about 15 to 20 minutes as opposed to approximately 32 man hours.

Other current applications of SPIRE include: creation of summary reports on salary increases for any time period, skills inventories, projections for salary budgeting, preparation of data for salary surveys, current salary status reports, salary distribution studies, and the generation of resumes for specific purposes.

IV. COMPUTER TECHNIQUES

SPIRE is centered about a master information tape. The information stored on this tape is first punched into cards. Each card contains all the pertinent information about a single happening in the employees past. Every card entering the system is identified by the category number and the employee's permanent number. The remainder of the card contains identifying codes for the data as well as alphameric descriptions.

The master tape is blocked by employee. All the information about any employee forms a single tape record. Within this record the information is held in card images sorted by category and date where applicable. This means that the maximum number of records to be read or bypassed will be the number of employees in the system. The employee's current salary grade is carried as part of the first card image so that for many applications a decision can be made as to whether or not a particular record is of interest to the search before actually performing a resume scan. This is particularly useful in the recruitment application.

The accuracy of the information contained in the master file is, of course, quite critical. To provide this accuracy, a great deal of time and effort has been placed in the program used to update the file. A total of 47 different errors are uniquely detectable in the updating data. This program allows total resume replacement or deleting, individual card replacement or deletion within any resume, and the addition of any new information to any given resume.

In general the customary way the master file is used in supplying information, other than complete resumes, is to scan the master tape and write a summary tape of the required information. The information on this summary tape may then be sorted or interrogated in any prescribed manner. Several of these codes that prepare summary tapes have been written. In the majority of special requests, one or more of these summary tapes can provide the necessary data. In these instances it is usually possible to write a program using ACF's magnetic tape FORCOM system which will interrogate the summary tape and prepare the necessary report. This provides a quick, economical way to satisfy demands made on the system.

V. CONCLUSIONS

SPIRE has proved a complete and useful system. It has resulted in large cost savings and has been well accepted by management. This acceptance is born out by the constant increase in requests for information from this system. To date there has been no request that could not be satisfied because the data had not been included in the system.

SPIRE was designed specifically for ACF's needs by ACF personnel and, as such, it may or may not be applicable to any other organization. This system is operational and in fact is far exceeding the original expectations in speed and versatility.

SECTY-STENO 4N HS GRAD PLUS SHORT SPECIALIZED TRNG 1-3YRS EXPR

NAME	PERM.NO.	SEX-MARITAL STATUS	TOTAL POINTS
BROWN S A	61205	FEMALE - MARRIED	71

DATE OF BIRTH - 08/16/37 DATE OF HIRE - 08/01/61 DATE OF REPORT - 01/01/63

***** PREVIOUS EXPERIENCE *****

TIME CODE	JOB TITLE
1-3 YEARS	STENOGRAPHER
3 MOS.- 1 YR.	TYPIST

***** ACF EXPERIENCE *****

104

DATE	JOB TITLE	GRADE	DEPT.	SALARY	PERS.ACT.	RATING	PROM.	SHIFT	NEXT REV.
08/01/61	TYPIST	2N	108	284	NH			1	
04/01/62	STENOGRAPHER	3N	108	314	RP			1	
07/16/62	STENOGRAPHER	3N	108	325	SA			1	1-6 MONTH
09/01/62	STENOGRAPHER	3N	108	345	MI	V. GOOD	6MO-1YR	1	5-ANNUAL

***** EDUCATION *****

EXTENT CODE	YR.COMP.	MAJOR TITLE	MINOR TITLE	UNIV. CODE
EQ. - 1 YR. COL.	1961	SHORTHAND		0
HIGH SCHOOL	1955	COMMERCIAL		0

POSITION REQUEST FOR EDP

AL-314

DATE REQUESTED	DATE REQUIRED	REQUESTED BY	POSITION TITLE AND CODE
01/01/63			1371041C

HAS THIS POSITION BEEN REQUESTED PREVIOUSLY? ☐ YES ☒ NO — IF YES, ARE PREVIOUS REQUIREMENTS ADEQUATE ☐ YES ☐ NO — IF NO, SHOW REQUIREMENTS OR ADDITIONAL REQUIREMENTS BELOW

POSITION REQUIREMENTS

SECTY-STENO 4N HS GRAD PLUS SHORT SPECIALIZED TRNG 1-3 YRS EXPR CC 11-80

EXPERIENCE REQUIREMENTS	NO. EXP CODES	NO. ED CODES	ED EXT	YRS EXP	MIN GRD	MAX GRD
	08	04	2	01	02	03

CODES	TITLE	CODES	TITLE
13732000	TYPIST (PREV.)		
13712000	STENOGRAPHER (PREV.)		
13710000	SECTY - STENO (PREV.)		
13301000	SECTY DEPT HEAD (PREV.)		
1373241C	TYPIST		
1371041C	STENOGRAPHER		
1371041C	SECTY - STENO		
1330142C	SECTY DEPT HEAD		

CODES	TITLE	CODES	TITLE
01100	ACADEMIC HIGH SCHOOL		
13301	SECRETARIAL		
13712	SHORTHAND		
13732	TYPING		

DEPARTMENT NAME AND NUMBER | GRADES | DATE | REVIEW PERIOD

ACF INDUSTRIES

MERIT REVIEW
AL-449 (1-63)

PERM. NO.	NAME	JOB TITLE	GRADE	DATE EMPLOYED	DATE ON JOB	PROM INCR	INITIAL RATE	LAST TWO MERIT INCREASES DATE	AMT.	DATE	AMT.	PRES. SALARY	TYPE OF REVIEW	RECOM. INCREASE AMT.	NEW SALARY	EFFECTIVE DATE	REMARKS
	GRADE =03N MIN =0325 MIDPT =0373 TOR =0404 MAX =0420										DEPT NO. 108						
	SMITH M F 61206	STENOGRAPHER	3N	09/02/59	01/15/62	25		07/01/62	25	08/01/62	30	320	ANNUAL				
	GRADE =04N MIN =0360 MIDPT =0413 TOR =0450 MAX =0465										DEPT NO. 108						
	BROWN S A 61205	SECTY-STENO	4N	08/01/61	01/16/63	30						375	6-MO.				
	WYLER J K 61207	SECTY-STENO	4N	01/01/57	05/15/60	30		08/01/61	25	08/01/62	25	439	ANNUAL				
	GRADE =05N MIN =0400 MIDPT =0460 TOR =0492 MAX =0520										DEPT NO. 108						
	CLOUGH G R 71208	SECTY DEPARTMENT HEAD	5N	01/06/59	03/01/62	40						450	6-MO.				

106

APPROVALS | DEPARTMENT MANAGER | DATE | ADMINISTRATIVE MANAGER | DATE | GENERAL MGR. ALBUQUERQUE DIVISION | DATE | ANALYST | DATE AL-96'S DISTRIBUTED

A COMPUTER PROGRAM FOR THE CALCULATION OF PRIME IMPLICANTS FROM A LIST OF BOOLEAN MINTERMS

THOMAS R. HOFFMAN
UNION COLLEGE
SCHENECTADY, NEW YORK

A Computer Program for the Calculation of
Prime Implicants from a List of Boolean Minterms

I Introduction

Boolean algebra has proven to be a most useful tool in the logical design of digital circuits. In general, the method employed involves four phases:

1- definition of the problem (verbal statements)

2- translation of verbal statements into Boolean equations

3- simplification of the equations

4- implementation of the simplified equations

Step 3 - Boolean simplification - can be done in a number of ways. One method is attributed to Quine.[1] The program to be discussed here enables the 1620 computer to simplify Boolean functions in a manner very similar to that proposed by Quine, obtaining as a result all the prime implicants of the original function.

II The Quine Method[2]

To simplify a Boolean function by the Quine method, it is necessary to express the function as a sum of minterms. The N-letter minterms are then systematically compared with each other two at a time, and all pairs differing in the state of only one letter are combined according to the identity:

$$X A + X \bar{A} = X$$

where A represents the letter eliminated and X is all other letters in the minterms involved.

After all possible comparisons have been made, there is in general a list of terms having N-1 letters. If any minterm did not combine at all during this process, it is a prime implicant. Minterms are checked as they are found to combine, to facilitate spotting of "non-combiners".

Comparisons continue among the N-1 letter terms, with the restriction that only terms containing the same letters can possibly combine. The N-1 letter terms fall in as many as N different sets, and further combination is possible only within these sets.

Successful combination of N-1 letter terms produces N-2 letter terms. Again, uncombinable terms are prime implicants.

This process is continued until no further combinations are possible. Uncombinable terms at any level are prime implicants. The Quine method then goes on to select the simplest set of prime implicants that is logically equivalent to the original function. This last phase is not handled by the program to be discussed - its output is simply a list of all prime implicants.

The reader is referred to Reference 2 for a much more detailed description of the Quine method, along with numerous examples.

III The Program - General

The program to be discussed was written by Mr. H. Huhta of the General Electric Co. as part of his M.S. thesis project at Union College. The required inputs are:

1: KV - the number of variables (cannot exceed 9)

2: KN - the number of minterms in the function (cannot exceed 60)

3: the list of minterms, coded in octal form (see IV A)

The resulting outputs are the prime implicants, coded as described in IV D. The FORTRAN-O language was used. The following paragraphs point up various aspects of the program organization. A detailed example is included, together with Veitch diagram verification of its correctness.

The Union 1620 has only 20 K memory. If both KV and KN are large, the program may not fit. To insure the ability of the machine to cope with any 9-variable problem, a second program was written to perform a single level of reduction. Thus the output, with 9-letter minterms, would be 8-letter terms which would then be reentered to produce 7-letter terms, etc. The second program will not be discussed further in this paper.

The first program has been shown to handle an 8-variable, 24-minterm problem successfully. The exact limits are not yet known.

IV Program Details

(A) Input Coding

Input minterms are coded as fixed-point octal numbers. Each octal digit specifies the state (i.e. - complemented or not) of three variables, hence three octal places will suffice for a 9-variable problem. For example the 5-letter minterm $A \bar{B} C \bar{D} E$ would be entered as 25, using the conventional scheme of representing an uncomplemented letter by 1 and a complemented letter by 0 - but converting to octal (25) rather than decimal (21).

(B) Combinability

Terms at any level are combinable if:

1- they contain the same letters

2- all letters but one are in the same state

Satisfaction of requirement 2 is detected by a two-step process. First, terms must differ in only one octal place, and second, the differing octal digits must differ in only one binary place. The fact that the second of these steps can be carried out at the octal level follows directly from the relationship between the binary and octal number systems. A table L (I) contains the necessary information. This 43-entry table relates Reference Octal Digit M and Compared Octal Digit N. It is addressed by the number N + 5M, and contents of each address is either 4,2,1 or 0 — to indicate the weight of the differing binary place. 0 means: "These two octal digits do not differ in just one binary place, hence they are not combinable."

Terms at any level in the process are always arranged in ascending order prior to comparison. Thus if M is larger than N, it means that the terms being compared differ in at least one other octal position - hence they do not combine. The entries for all cases of $M > N$ are therefore 0 in the table.

(C) Set Identification

As mentioned in II, comparisons at all levels beyond the minterm level take place only within sets. The program must therefore generate a set identification number (S I N) to identify each combined pair. The S I N increment generated by any particular comparison is simply the octal weight of the letter eliminated. As terms pregress through successive levels, an up-to-date S I N is maintained by cumulative addition of the weight of each eliminated variable.

(D) Example - to determine the prime implicants of the function:

$$f = m_3 + m_4 + m_5 + m_7 + m_9 + m_{10} + m_{12} + m_{13}$$

(1) List minterms in ascending order in octal system.

$m_3 \rightarrow 0003$

$m_4 \rightarrow 0004$

$m_5 \rightarrow 0005$

$m_7 \rightarrow 0007$

$m_9 \rightarrow 0011$

$m_{10} \rightarrow 0012$

$m_{12} \rightarrow 0014$

$m_{13} \rightarrow 0015$

(4 digits are shown, because this is the way FORTRAN 0 represents fixed point numbers internally. Leading 0's need not be typed.)

(2) Investigate all possible combinations. Start with 3, and compare each term with all higher-numbered ones.

ex: 3 = 011 three binary differences $\therefore$ not combinable
4 = 100

3 = 011 two binary differences $\therefore$ not combinable
5 = 101

3 = 011 combinable; the most significant binary place
7 = 111 is the one that combines

The decision as to whether or not a pair of octal digits is combinable is made by referring to the table L (I), as discussed in IV-B. In the 3-7 comparison just described, the most significant place is the one that combines, so address 22 (7 + 5 X 3 = 22) of the table contains the number 4 - the octal weight of the eliminated variable. 4 now becomes the S I N of the combined term. It is attached in the 5th place of the lower-valued of the two combined minterms, and the result stored to await processing at the next level (5 places are now possible because the combined term is given a floating point FORTRAN name).

This process continues until all possible combinations are tried. Minterms involved in one or more comparisons are tagged by putting a 9 in the 4th (most significant) place. If any minterms have not entered into any comparisons when the processing of level N is complete, they are printed out. They are prime implicants.

The complete level 4 results appear below:

Initial Level 4	Final Level 4	Transferred to Level 3	Combination of minterms
0003	9003	40003	0003 and 0007
0004	9004	10004	0004 and 0005
0005	9005	100004	0004 and 0014
0007	9007	20005	0005 and 0007
0011	9011	100005	0005 and 0015
0012	0012	40011	0011 and 0015
0014	9014	10014	0014 and 0015
0015	9015		

At this point, 0012 (minterm ten) is outputted - it is a prime implicant. Output coding will be discussed in paragraph (3)

Processing of level 3 starts by arranging the transferred terms into sets having identical S I N's, and putting the terms in each set into ascending order. The result is:

10004 }
10014 } S I N = 1

20005 S I N = 2

40003 }
40011 } S I N = 4

100004 }
100005 } S I N = 10

Digit-by-digit comparisons are now made within each set, just as before. Since the S I N's of two terms being compared are the same, comparison need be made only of the four least significant digits. (The S I N is actually dropped before comparison, by returning to use of fixed-point variables.)

The terms with S I N = 1 combine, since they differ only in the second least significant octal place, and 0 and 1 combine (Table L (I)) to eliminate the variable of weight 1. The new S I N is therefore 10 (generated at level 3) + 1 (generated at level 4) = 11. The term transferred is therefore 110004. Note that the S I N increment of 10 indicates that the variable eliminated had weight eight.

There is only one term having S I N = 2, hence 20005 is outputted at the conclusion of level 3 processing. Similarily, the two terms with S I N = 4 do not combine, hence both 40003 and 40011 are outputted. All three of these terms are prime implicants.

The terms with S I N = 10 combine, since 4 and 5 combine to eliminate the variable of weight 1. The new S I N would be 11, and the transferred term 110004 - identical to that derived from the S I N = 1 set. The program avoids this duplication by a means that will be discussed in paragraph 6.

Since only one term was transferred to level 2, it is of course a prime implicant. The problem is now complete, except for the decoding of the computer output. There are five prime implicants, so far identified as follows:

At level 4: minterm 00012 (note that the S I N of all minterms is 0)

At level 3: terms 20005
40003
40011

At level 2: term 110004

(3) Output Format

Three fixed point numbers are printed out for each prime implicant. They are:

1- the minterm from which the P.I. was derived;
2- the final S I N, to tell which letters have been eliminated;
3- the level, to tell how many letters will appear in the P.I. (this is a check on the others; it contains no new information)

If two P.I.'s come from the same set, the S I N and the level will be printed only once.

The print-out for the example will therefore appear as follows:

12 ← minterm	
0 ← SIN	4 ← level
5	
2	3
3	
11	
4	3
4	
11	2

(4) Output Decoding

The output format of (3) can be translated to literal form as follows (using the first prime implicant to illustrate):

1- write letters for variables: A B C D

2- translate octal minterm to binary: 12 → 1010

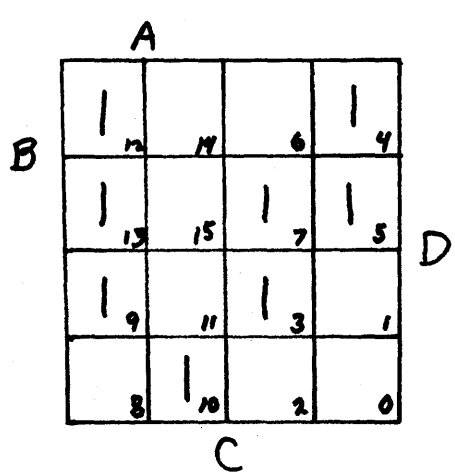
3- translate S I N to binary: 0 → 0000

Now each 0 in binary S I N indicates a letter present in the prime implicant (in this case, all four). The binary minterm now indicates complemented variables by 0, true variables by 1. The prime implicant is therefore $A\ \overline{B}\ C\ \overline{D}$, which is minterm ten, in the usual decimal notation. The output level -4- verifies the fact that the term should contain four letters, which it does. Similarly, the other prime implicants decode as follows:

		A B C D	
5			
2	3	0 1 0 1	→ $\overline{A}\ B\ D$
		0 0 1 0	
3		0 0 1 1	→ $\overline{A}\ C\ D$
4	3	0 1 0 0	
11		1 0 0 1	→ $A\ \overline{C}\ D$
4	3	0 1 0 0	
4		0 1 0 0	→ $B\ \overline{C}$
11	2	1 0 0 1	

(5) Check of Results

The solution of this problem by Veitch diagram is shown below. It is readily verified that the computer solution is correct.



Prime implicants: $A\ \overline{B}\ C\ \overline{D}$ (m_{10})

$B\ \overline{C}$ $(m_4 + m_5 + m_{12} + m_{13})$

$\overline{A}\ B\ D$ $(m_5 + m_7)$

$\overline{A}\ C\ D$ $(m_3 + m_7)$

$A\ \overline{C}\ D$ $(m_9 + m_{13})$

(6) Elimination of Duplication

It is inherent in the Quine system that identical combined terms will arise from different series of comparisons. For example, a term $A \bar{C}$ could arise by eliminating B from $A B \bar{C}$ and $A \bar{B} \bar{C}$, or by eliminating D from $A \bar{C} D$ and $A \bar{C} \bar{D}$. The program eliminates such duplication at all levels by the following procedure:

> When a combinable pair is found, the old S I N and the weight of the variable just eliminated are compared. The term is transferred to the next level of comparison only if the old S I N is <u>less than</u> the weight of the variable just eliminated. For example, in the problem just illustrated, the two terms with S I N = 1 combined to eliminate the variable of weight 10 (decimal <u>eight</u>). Since $1 < 10$, the term was transferred. Later on, the two terms with S I N = 10 combined to eliminate the variable of weight 1 (which would have produced the same term). This term was <u>not</u> transferred because $10 > 1$.

References

(1) "The Problem of Simplifying Truth Functions" by W.V. Quine (American mathematical monthly, p. 521, 1952)

(2) "Logical Design of Digital Computers" by M. Phister (Wiley, 1958) (p. 68-75)

Thomas R. Hoffman

Thomas R. Hoffman
Prof. of Elect. Eng.
Union College

CRITICAL SPEED. STRESS, AND BEARING REACTION CALCULATIONS FOR A GENERAL SHAFT USING NUMERICAL INTEGRATION

BY RALPH B. BATES

CONTENTS

SUMMARY

Critical speed, stresses, and bearing reactions can be calculated on a digital computer for a general two bearing shaft, which can have each bearing located anywhere on the shaft, any number of cross sections, variable loading, and any length.

Besides eliminating the tedious labor of the calculations, the computer provides flexibility. A number of calculations may be rapidly made to optimize design or to check out application variations on a standard design.

Basis of the Calculations

In general, the critical speed calculation is based on Rayleigh's method using the following formula

$$N_C = 187.7\sqrt{\frac{\sum|W_s y_s|}{\sum|W_s y_s^2|}}$$

where N_C is the first critical speed, rpm

W_s is the load at the s position excluding external loads without mass, such as belt pull, pounds.

y_s is the static deflection of the load at the s position due to the loads W_s with the loads reversed between bearings.

The static deflections and bending moments are found by using the same principles of mechanics which are the basis of graphical determination of beam deflections. Numerical integration is used to relate load, shear, bending moment, slope, and deflection. This integration, although numerical, gives exact areas as long as the loads and moments of inertia are constant over the increments of shaft length.

The bearing reactions are found by taking moments about each bearing equal to zero in accordance with statics.

The shaft bending tensile stress, torsional shear stress, and combined shear stress are calculated according to the usual equations of strength of materials.

Method

The method is illustrated in detail by an example calculation. Briefly, it consists of dividing the shaft into increments of length, determining the load and shaft moments of inertia in each increment and the computer calculates critical speed, stresses and bearing reactions. If there is only shaft weight in a number of increments, the O.D. and I.D. of the shaft are sufficient input data and the computer will calculate the load and moments of inertia.

Other Factors in Critical Speed

There are other factors affecting critical speed, such as, shear deflection, gyroscopic effects, bearing length, and bearing flexibility. Bearing support and oil film deflection may be accounted for by adding the bearing deflection to the deflections used in calculating critical speed.

INTRODUCTION

Critical speed, stresses, and bearing reactions for a general shaft may be calculated on a digital computer by utilizing numerical integration. The general shaft has each of two bearings located anywhere on the shaft, any number of cross sections, and any number of loads.

Besides eliminating the tedious labor of the calculations, the computer offers flexibility. It allows the calculation of a number of variations in order to optimize design or evaluate application variations of standard designs.

THEORY

Rayleigh's Energy Method (4) and (7)[1]

Rayleigh's energy method may be used to determine lowest natural frequency of a system. The method consists of assuming a configuration for the system which will approximate the maximum amplitude of the fundamental mode or lowest natural frequency. Based on this configuration, the maximum potential energy is calculated at maximum displacement (zero velocity) and equated to the maximum kinetic energy at the system equilibrium position (maximum velocity). This relationship may then be solved for the lowest natural frequency.

The dynamic deflection curve at maximum displacement for lateral displacement of a beam may be assumed to be very close to the static deflection curve. When the configuration due to static loading of a beam with its own loads is used for calculating the fundamental frequency by Rayleigh's method, the frequency is within the accuracy required for most engineering calculations.

Since Rayleigh's method consists essentially of equating inertia forces of the rotating masses to the restoring forces of the shaft, external forces which do not have mass, such as belt pull, must not be included. Rayleigh found that the system vibrates in a manner which makes the frequency a minimum. This means that the inertia forces of the masses must act in directions causing maximum deflections. In a two bearing shaft the static loads between the bearings must be reversed to produce maximum deflection and minimum frequency.

The potential energy involved in Rayleigh's method is the work done on the beam in moving its loads from the equilibrium position to the dynamic deflection curve at maximum deflection. (This is assumed the same as the static deflection curve without, external loads and with loads reversed between bearings.)
Thus:

$$P.E. = 1/2 \sum \left| W_s y_s \right|$$

The maximum velocity of the weight W_s with amplitude y_s in harmonic motion is py_s - Hence, the maximum kinetic energy of the system is

$$K.E. = \frac{1}{2g} \sum \left| W_s (py_s)^2 \right|$$

---

1. Numbers in parenthesis designate references at end of paper.

The symbols in the previous equations are:

P.E. is potential energy.

W_s is the load at the s position, $s = 1,2,3,\cdots$.

y_s is the deflection of the load at the s position.

K.E. is the kinetic energy.

g is the acceleration of gravity.

p is the natural circular frequency.

The law of conservation of energy requires that the potential energy equal the kinetic energy in undamped systems. Solving for p^2 gives the following result:

$$p^2 = g\sqrt{\frac{\sum |W_s y_s|}{\sum |W_s y_s^2|}}$$

The first critical speed of a shaft may be considered to occur when the circular frequency (ω) of shaft rotation equals the natural frequency of lateral vibration of the rotors (considered as a beam). The rotating shaft is deflected by the small unbalance forces in the rotating system, which cannot be eliminated entirely in actual rotors. These exciting forces have a frequency in a lateral plane equal to the circular frequency of rotation. When the exciting forces have the same frequency as the natural frequency of lateral vibration, resonance occurs, which usually causes undesireable excessive vibration of the machine. Thus:

$$\omega^2 = p^2 = g\sqrt{\frac{\sum |W_s y_s|}{\sum |W_s y_s^2|}}$$

Hence:

$$N_c = 187.7\sqrt{\frac{\sum |W_s y_s|}{\sum |W_s y_s^2|}} = 187.7\sqrt{\frac{|W_1 y_1| + |W_2 y_2| + |W_3 y_3| + \cdots}{|W_1 y_1^2| + |W_2 y_2^2| + |W_3 y_3^2| + \cdots}}$$

where

N_c is the first critical speed, rpm.

W_s is the load at the s position excluding external loads without mass, such as belt pull, pounds.

y_s is the static deflection of the load at the s position (W_s) with the loads reversed between the bearings, inches.

Deflection

The static deflections of the shaft system may be determined by utilising the principles of mechanics which are the basis of graphical determination of beam deflections as shown in Figure 1,(13), (10), and (3).

Bearing Reactions

The bearing reactions are found by taking moments about each bearing equal to zero in accordance with statics.

ℓ is the load per unit length.

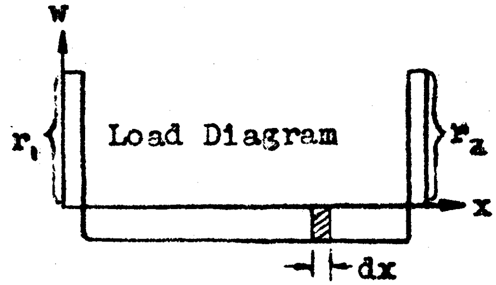
Simply Supported Beam

---

r_1 is the bearing reaction at the left end converted to load per unit length.

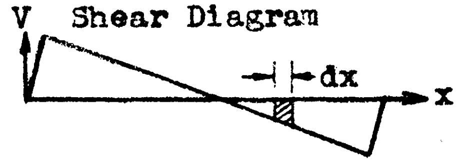
r_2 is the bearing reaction at the right end converted to load per unit length.

w is the load per unit length including the bearing reactions.



V is the shear.

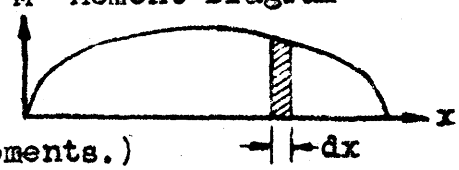
$V = \int w\, dx$ = Area under the w curve; and $\frac{dV}{dx} = w$



M is the bending moment.

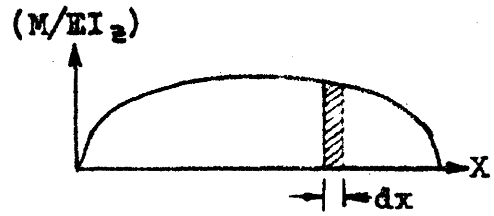
$M = \int V\, dx$ + external moments; and $\frac{dM}{dx} = V$

M = Area under the V curve + M_{ext}.
(Most shafts have zero external bending moments.)



E is the modulus of elasticity.

I_2 is the moment of inertia of the beam cross-sectional area with respect to the axis in the neutral plane.

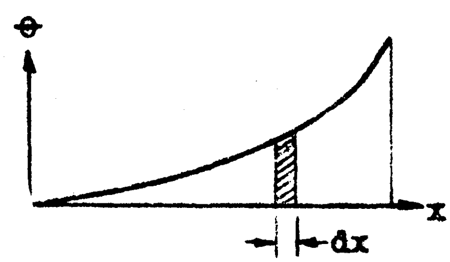


θ is the area under the (M/EI_2) curve.

$S + C_s = \int (M/EI_2)\, dx = \theta$; and $\frac{dS}{dx} = \frac{M}{EI_2}$

$S = \theta - C_s$; and $\frac{dS}{dx} = \frac{d\theta}{dx} = \frac{M}{EI_2}$

S is the slope and C_s is the integration constant.



δ is the area under the θ curve.

$y + C_y = \int S\, dx = \int (\theta - C_s)\, dx = \delta - C_s x$; and $\frac{dy}{dx} = \theta - C_s$

$y = \delta - C_s x - C_y$; and $\frac{dy}{dx} = \frac{d\delta}{dx} - C_s$; so $\frac{d\delta}{dx} = \theta$

y is the deflection, and C_y is the constant.

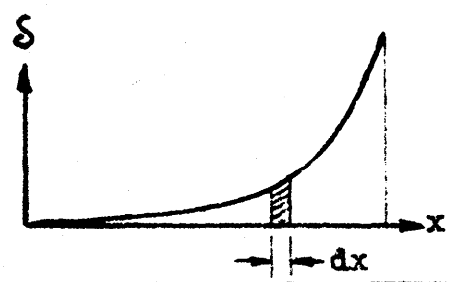


FIGURE 1 - BASIS OF GRAPHICAL DETERMINATION OF DEFLECTIONS

Stresses

The shaft bending tensile stress, torsional shear stress and combined shear stress are calculated according to the usual equations of strength of materials (13).

$$S_{bs} = M_s \frac{c}{I_{xs}} ; \quad S_{ts} = T_s \frac{c}{I_{ps}} ; \quad S_{cs} = \sqrt{S_{ts}^2 + \left(\frac{S_{bs}}{2}\right)^2}$$

where all values are at section s of the shaft:

S_{bs} is bending tensile stress, psi

M_s is bending moment, in lbs.

$\frac{c}{I_{xs}}$ is the radius to the outer shaft surface divided by the moment of inertia of the cross sectional area with respect to radial axis, in.$^{-3}$.

S_{ts} is the shaft torsional shear stress, psi

T_s is the torque in the shaft, in. - lbs.

$\frac{c}{I_{ps}}$ is the radius to the outer shaft surface divided by the moment of inertia of the cross sectional area with respect to the longitudinal axis, in.$^{-3}$

S_{cs} is the shaft maximum combined shear stress, psi

NUMERICAL INTEGRATION

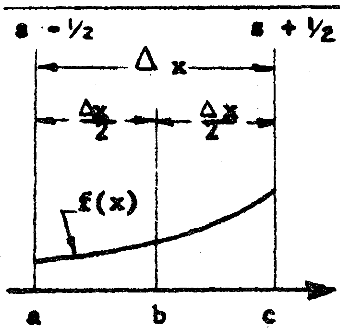
The integrations shown in Figure 1 may be performed by numerical means with a computer. Numerical integration techniques may be approximate, (8) and (15). However, this numerical integration is exact if the load and the moment of inertia are constant over the increment of x, (Δx) and the following formulas are used.

Tayler's Formula (8)

From calculus Tayler's formula with remainder follows:

$$f(x) = f(a) + \frac{f'(a)}{1!}(x-a) + \ldots + \frac{f^{(n)}(a)}{n!}(x-a)^n + \frac{f^{(n+1)}(x_1)}{(n+1)!}(x-a)^{n+1}$$

In Figure 2 if $x = b$ then $(x - a) = \frac{\Delta x}{2}$ and if $x = c$ then $(x - a) = \Delta x$



Δx is the increment of length along the beam.
s is the station line in the center of Δx.
Subscript s designates the value at the station line.
Subscript s - ½ designates the value at the left of Δx.
Subscript s + ½ designates the value at the right of Δx.

FIGURE 2

Shear Formulas

$$V' = \frac{dV}{dx} = w = (\text{constant over } \Delta x) = w_s \; ; \quad V'' = 0$$

$$V_{s+1/2} = V_{s-1/2} + w_s(\Delta x)$$

Bending Moment Formulas

$$M' = \frac{dM}{dx} = V; \; M'' = V' = w_s \; ; \; M^{(3)} = 0$$

$$M_{s+1/2} = M_{s-1/2} + V_{s-1/2}(\Delta x) + \frac{1}{2} w_s (\Delta x)^2$$

$$M_s = M_{s-1/2} + \frac{1}{2} V_{s-1/2}(\Delta x) + \frac{1}{8} w_s (\Delta x)^2$$

Slope Formulas

$\frac{dS}{dx} = \frac{M}{EI_z}$ and since $S = \theta - C_s$; $\frac{dS}{dx} = \frac{d\theta}{dx}$

where EI_{zs} is constant over Δx

$$\theta' = \frac{d\theta}{dx} = \frac{M}{EI_z} \; ; \; \theta'' = \frac{M'}{EI_z} = \frac{V}{EI_z} \; ; \; \theta^{(3)} = \frac{M''}{EI_z} = \frac{w_s}{EI_z} \; ; \; \theta^{(4)} = 0$$

$$\theta_{s+1/2} = \theta_{s-1/2} + \frac{1}{EI_{zs}} \left[M_{s-1/2}(\Delta x) + \frac{1}{2} V_{s-1/2}(\Delta x)^2 + \frac{1}{6} w_s (\Delta x)^3 \right]$$

Deflection Formulas

$\frac{dy}{ds} = S$ and since $y = \delta - C_s x - C_y$ $\quad \frac{dy}{dx} = \frac{d\delta}{dx} - C_s = S$

$\frac{d\delta}{dx} = S + C_s = \theta$ therefore $\delta' = \theta$; $\delta'' = \theta' = \frac{M}{EI_z}$

$$\delta^{(3)} = \theta'' = \frac{V}{EI_{zs}} \; ; \; \delta^{(4)} = \theta^{(3)} = \frac{w_s}{EI_z} \; ; \; \delta^{(5)} = 0$$

$$\delta_s = \delta_{s-1/2} + \frac{1}{2}\theta_{s-1/2}(\Delta x) + \frac{1}{EI_{zs}} \left[\frac{1}{8} M_{s-1/2}(\Delta x)^2 + \frac{1}{48} V_{s-1/2}(\Delta x)^3 + \frac{1}{384} w_s (\Delta x)^4 \right]$$

Error

Therefore, the only error introduced in the integration is due to the approximation which is necessary to keep the loading and moment of inertia constant over the increment of x, (Δx). If desired, the amount of error can be determined by approximating the load with high values and the moment of inertia with low values for one calculation. A second calculation is then made by approximating the load with low values and the moment of inertia with high values. The total error will be less than the difference between results of the two calculations. The error may be reduced by making further calculations with smaller increments of x. This is not required for most engineering calculations with the usual safety factors.

METHOD

The calculation method is illustrated by using the shaft shown in Figure 3a. This is a compressor rotor mounted on a stepped shaft with two bearings. Because the loads need to be regarded differently, the calculations are done in two parts.

Example Calculation - Part 1

First, the bearing reactions and stresses are determined including external loads without mass and with loads in their normal direction (downward) as tabulated in Table 1.

The input data for Part 1 of the calculations is described below:

Column	Symbol	Description
-	Δx	The increment of lengths
-	-	Station numbers of bearing locations.
1	s	Station line number in the center of Δx.
2	ℓ_s	Load per unit length including external loads without mass in normal direction. Concentrated loads must be distributed over the increment Δx. Actual shaft and rotor loads are more often distributed than concentrated.
9	C/I_{xs}	See Page 6.
11	T_s	See Page 6.
12	C/I_{ps}	See Page 6.

The calculations made for Part 1 are described below:

Column	Symbol	Description
3	m_2	✓ Moments about bearing 2 station line as follows $m_2 = \ell_s$ [s(of bearing 2) - s] calculated at each station line, lb/in x station number.
4	m_1	✓ Moments about bearing 1 station line as follows: $m_1 = \ell_s$ [s(of bearing 1) - s] calculated at each station line., lb/in x station number.

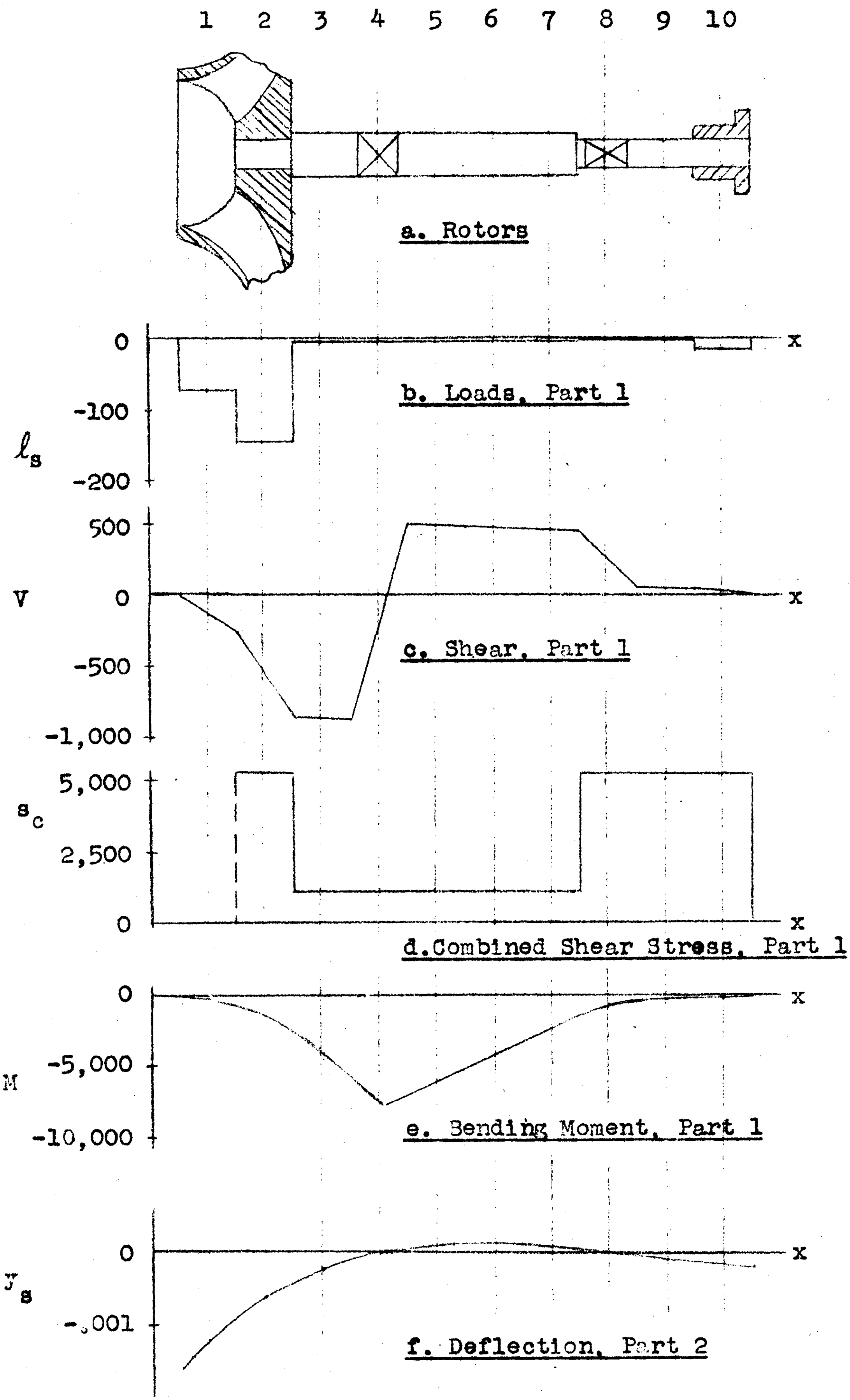


FIGURE 3 - EXAMPLE CALCULATION DIAGRAMS

124

Column	Symbol	Description
-	r_1	Bearing reaction 1, lbs/in $r_1 = - \frac{\sum (m_2 \text{ at each station line})}{s(\text{at bearing } 1) - s(\text{at bearing } 2)}$
-	R_1	Bearing reaction 1, lbs. $R_1 = r_1 (\Delta x)$
-	r_2	Bearing reaction 2, lbs/in $r_2 = + \frac{\sum (m_1 \text{ at each station line})}{s(\text{at bearing } 2) - s(\text{at bearing } 1)}$
	R_2	Bearing reaction 2, lbs. $R_2 = r_2 (\Delta x)$
5	w_s	Load per unit length including bearing reactions $w_s = \ell_s$ except at bearing 1, $w_s = \ell_s + r_1$ and at bearing 2, $w_s = \ell_s + r_2$
6	$V_{s+1/2}$	See Pages 6 and 7.
7	$M_{s+1/2}$	See Pages 6 and 7.
8	M_s	See Pages 6 and 7.
10	S_{bs}	See Page 6
13	S_{ts}	See Page 6
14	S_{cs}	See Page 6

Example Calculations - Part 2

Second, the critical speed is determined excluding external loads without mass and with the loads between bearings reversed (upward) as tabulated in Table 2. Note the shrink fit of the compressor rotor and the coupling hub may be considered to increase the shaft diameter when calculating critical speed, (14).

The input data for Part 2 of the calculation is described below:

Column	Symbol	Description
-	Δx	The increment of length.
-	-	Station numbers of bearing locations.
1	s	Station line numbers in the center of Δx.
2	ℓ_s	Load per unit length excluding external loads without mass. Concentrated loads must be distributed over the increment Δx. Actual shaft and rotor loads are more often distributed than concentrated.
12	$EI_{\chi s}$	$EI_{\chi s}$ at station s. See Page 5.

125

The calculations made for Part 2 are described below:

Column	Symbol	Description
3	m_2	See Part 1.
4	m_1	See Part 1.
-	r_1	See Part 1.
-	r_2	See Part 1.
5	w_s	See Part 1. The inclusion of bearing reactions may be questioned since they are external loads without mass. However, it may be shown that the inclusion of bearing reactions is necessary to make the integration $V = \int w dx$ correct along the length of the shaft.
6	W_s	Load, excluding external loads at station number s, lbs. $W_s = \ell_s (\Delta x)$.
7	$V_{s+1/2}$	See Pages 6 and 7.
8	$M_{s+1/2}$	See Pages 6 and 7.
10	$\theta_{s+1/2}$	See Pages 5, 6 and 7.
11	δ_s	See Pages 5, 6, and 7.
12	y_s	Deflection. See following.
13	$\lvert W_s y_s \rvert$	Absolute value of column 6 x column 12 at station s.
14	$\lvert W_s y_s^2 \rvert$	Absolute value of column 13 x column 12 at station s.
-	N_c	First critical speed. See Page 3. $N_c = 187.7 \sqrt{\frac{\sum \lvert W_s y_s \rvert}{\sum \lvert W_s y_s^2 \rvert}}$

Determination of ys from δ_s

It was shown (Page 5) that

$$y = \delta - C_s x - C_y$$

Since x is the distance along the beam: $x = s (\Delta x)$

Let s_a = the station number at bearing 1
s_b = the station number at bearing 2
δ_a = the δ_s at bearing 1
δ_b = the δ_s at bearing 2

Since the deflection y = 0 at bearing 1 and bearing 2 then

$$\delta_a = C_s s_a (\Delta x) + Cy \text{ and } \delta_b = C_s s_b (\Delta x) + Cy$$

Solving these two equations for C_s and Cy and substituting into the equation for y then

$$y_s = \delta_s - \frac{(s_b \delta_a - s_a \delta_b)}{(s_b - s_a)} - \frac{(\delta_b - \delta_a)}{(s_b - s_a)} s$$

This is the same as

$$y_S = \delta_s - \delta_a - \frac{(\delta_b - \delta_a)(s - s_a)\Delta x}{(s_b - s_a)\Delta x}$$

Hence y is the difference in value between the value of δ_s and the straight line δ_1 which goes through each bearing as shown in Figure 4.

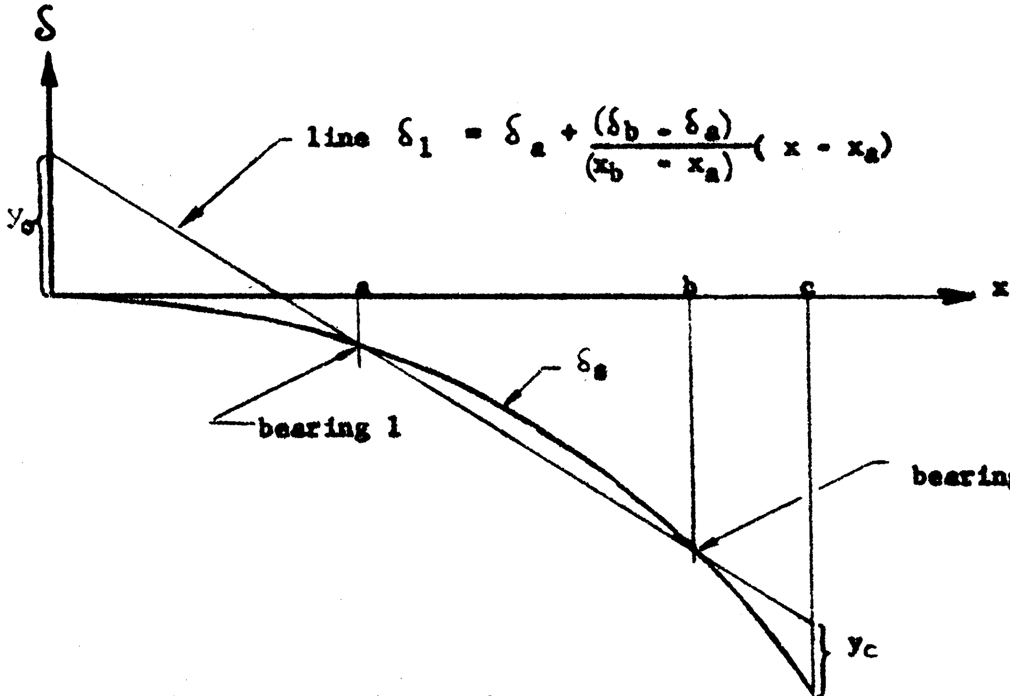


FIGURE 4

REMARKS

Use of a Computer

Because of the tedious labor involved in these general shaft calculations, the use of a computer is necessary. The input data for the computer computations differs slightly from that described in the example, which illustrated the method.

The input data to the computer is as follows:

1. Identification
2. Δ x.
3. Bearing location station numbers.
4. External loads without mass and location station numbers.
5. Horsepower transmitted with start and end station numbers.
6. rpm
7. Modulus of elasticity.

127

In addition either of the following:

8. Loads excluding shaft weight and external loads withour mass.	OR	8' Loads excluding external loads without mass.
9. Shaft O.D. with start and end station numbers.		9' I_{xs}
10. Shaft I.D. with start and end station numbers.		10' I_{ps}
		11' C_s which is 1/2 shaft O.D.

Values such as 5, 9, and 10 which may be the same for a number of stations are easier to put into the computer as a single value from start station number to end station number. External loads, 4, are put into the computer separately so that they may be used for calculations similiar to Part 1 of the example and excluded from calculations similiar to Part 2 of the example. If the shaft has an unusual section so that c, I_x and I_p cannot be calculated from shaft O.D and I.D. then 8', 9', 10', and 11' are the input data of that station instead of 8, 9 and 10.

The calculations done by the computer are similar to those in the example except ℓ_s (due to shaft weight), C/I_{xs}, C/I_{ps}, T_s and EI_{xs} are calculated from input data 5, 6, 7, 8 and 9. The computer uses either input data 4 + 8 + ℓ_s (due to shaft wt) or input data 4 + 8' as ℓ_s in calculations similar to Part 1. In calculations similar to Part 2, the computer uses input data 8 + ℓ_s (due to shaft wt) or input data 8', with the loads between bearings reversed, as ℓ_s.

Advantages

The principal advantages of these general shaft calculations are listed below:

1. The calculations apply to any two bearing shaft with each bearing located anywhere on the shaft with any variable cross sections and loading and with any length.
2. Boundary conditions are evaluated in the calculations by having the first load at station 1 and using zero deflection at each bearing.
3. Designs may be optimized by calculating a number of variations by computer and selecting the best.
4. Variations in standard design, such as a number of belt pulls, may be checked by making a number of calculations by computer.

Accuracy

The accuracy of the calculations are affected by the following:

1. Having the load constant over the increment of length (Δx).
2. Having the shaft cross section moments of inertia constant over the increment of length (Δx).
3. Assuming the static deflection curve is the dynamic deflection in calculating critical speed.

4. Concentrating the load in the increment of length in calculating critical speed.

5. Using Rayleigh's method to calculate critical speed.

The error due to 1 and 2 preceding may be evaluated as indicated on Page 8.

The error due to 4 may be minimized by taking small increments of length (Δx).

The errors due to 3 and 5 may be reduced by recalculating the critical speed using the inertia forces as loads (4)

$$W_{s2} = W_{s1} \, y_{s1} \left(\frac{N_{c1}}{187.7} \right)^2 \quad ; \quad \ell_{s2} = \frac{W_{s2}}{\Delta x}$$

Then calculating a second critical speed value

$$N_{c2} = 187.7 \sqrt{\frac{\sum |W_{s2} y_{s2}|}{\sum |W_{s1} \, y_{s2}^2|}}$$

Subscript 1 refers to values in the first calculation and subscript 2 refers to values in the second calculation.

Note that W_{s1} is used in the calculation of N_{c2} since the kinetic energy is a function of mass while potential energy is a function of the inertia force.

Other Factors in Critical Speed

There are other factors affecting critical speed such as shear deflection, (3) and (5), gyroscopic effects, (2), (3), and (6), bearing length (3), and bearing flexibility (3). Bearing support and oil film flexibility may be accounted for by adding the bearing deflection to the y_s in the calculation of critical speed.

REFERENCES

1. Andersen, R. A. "Flexural Vibrations of Uniform Beams According to the Timoshenko Theory". Jour. Appl. Mech; v. 20, n. 4, (Dec. 1953)

2. Bert, Charles W. "Deflections in Stepped Shafts". Machine Design November 24, 1960 pp. 129-133.

3. Church, A. H. "Centrifugal Pumps and Blowers". New York: John Wiley & Sons, 1944, pp. 291-296

4. Church, A. H. Mechanical Vibrations. New York: John Wiley & Sons, 1957, pp. 204-208.

5. Cowie, Alexander "A Tabular Method for Calculating Deflections of Stepped and Tapered Shafts". Machine Design (August 9, 1956)

6. DenHartog, J. P. Mechanical Vibrations, 4th ed., New York: McGraw-Hill Book Company, p. 265 (1956)

7. Hansen, H. M. and Paul F. Chenea, Mechanics of Vibration. New York; John Wiley & Sons, pp. 44-46, 274-286 (1952)

8. Kaplan, Wilfred Advanced Calculus, Reading, Massachusetts; Addison-Wesley Publishing, 1952, pp. 167-176, 358.

9. MacDuff, John N. and John R. Curreri, Vibration Control. New York; McGraw-Hill Book Company, 1958.

10. Marks, Lionel S. Mechanical Engineers Handbook, 5th ed. New York: McGraw-Hill Company, 1951, pp. 437-439.

11. Mykelstad, "Numerical Analysis of Forced Vibrations of Beams". Journal of Applied Mechanics, v. 20, (Mar. 1953)

12. Prohl, M. A. "A General Method for Calculating Critical Speeds of Flexible Rotors", Journal of Applied Mechanics, v. 13, n. 4, (Dec. 1946)

13. Timoshenko, S. and Gleason H. MacCullough, Elements of Strength of Materials, 3rd ed., New York; D. Van Nostrand Company, 1949, pp. 105-110, 163-165, 271-273.

14. Timoshenko, S. and D. H. Young, Vibrations Problems in Engineering, 3rd ed., New York: D. Van Nostrand Company, 1954, p. 40.

15. Ungar, Eric C. "Numerical Integration Simplifies Complex Design". Product Engineering (March 16, 1959), pp. 50-54.

TABLE 1 - EXAMPLE CALCULATION - PART 1

(Including external loads without mass and with normal direction of loads)

Δx = 4 inches - Bearing No 1 at s = 4 - Bearing No 2 at s = 8

1	2	3	4	5	6	7	8	9	10	11	12	13	14
s	ℓ_s	m_2	m_1	w_s	$V_{s+1/2}$	$M_{s+1/2}$	M_s	$\frac{c}{I_{zs}}$	S_{bs}	T_s	$\frac{c}{I_{ps}}$	S_{ts}	S_{cs}
	#/in	#/in	#/in	#/in	#	in-lbs.	in-lbs.	in^{-3}	psi	in-lbs	in^{-3}	psi	psi
00				0									
0.5	[illegible]	[illegible]	[illegible]		0	0							
1.0	-70	-490	-210	-70			-140						
1.5					-280	-560							
2.0	-144	-864	-288	-144			-1408	0.40	563	26300	0.20	5260	5261
2.5					-856	-2832							
3.0	-4	-20	-4	-4			-4552	0.16	730	26300	0.08	1120	1130
3.5					-872	-6288							
4.0	-4	-16	0	342			-7348	0.16	1175	26300	0.08	1120	1138
4.5					496	-7040							
5.0	-4	-12	+4	+4		-	-6166	0.16	986	26300	0.08	1120	1132
5.5					480	-5088							
6.0	-4	-8	+8	-4			-4136	0.16	661	26300	0.08	1120	1128
6.5					464	-3200							
7.0	-4	-4	+12	-4			-2280	0.16	365	26300	0.08	1120	1122
7.5					448	-1376							
8.0	-2	0	+8	-96			-672	0.40	278	-6300	0.20	5260	5260
8.5					64	-352							
9.0	-2	+2	+10	-2			-228	0.40	91	26300	0.20	5260	5260
9.5					56	-112							
10.0	-14	+28	+84	-14			-28	0.40	11	26300	0.20	5260	5260
10.5					0	0							
TOTAL	-252	-1384	-376	0									

r_1 = + 346 #/in. r_2 = - 94 #/in.
R_1 = 1384 lbs R_2 = -376 lbs.

TABLE 2 - EXAMPLE CALCULATION - PART 2

(Excluding external loads without mass and with loads reversed between bearings)

Δx = 4 inches - Bearing No. 1 at s = 4 - Bearing No. 2 at s = 8

1	2	3	4	5	6	7	8	9	10	11	12	13	14
s	l_s	m_2	m_1	w_s	W_s	$V_{s+\frac{1}{2}}$	$M_{s+\frac{1}{2}}$	EI_{zs} x 10^6	$\theta_{s+\frac{1}{2}}$ x10^4	δ_s x10^6	y_s x10^6	$W_s y_s$ x10^4	$W_s y_s^2$ x10^7
	#/in	#/in	#/in	#/in	#	#	in-lbs	psi	rad.	in.	in.	#/in	#/in^2
00													
0.5						0	0		0				
1.0	-70	-490	-210	-70	-280			1000		-17	-1154	3240	3740
1.5						-280	-560		-75				
2.0	-144	-864	-288	-144	-576			1000		-49	-634	3650	2320
2.5						-856	-2832		-677				
3.0	-4	-20	-4	-4	-16			360		-202	-235	38	9
3.5						-872	-6288		-5717				
4.0	-4	-16	0	330	-16			360		-519	0	0	0
4.5						448	-7136		-13937				
5.0	4	12	-4	4	16			360		-966	105	17	2
5.5						464	-5312		-20887				
6.0	4	8	-8	4	16			360		-1497	126	20	3
6.5						480	-3424		-26067				
7.0	4	4	-12	4	16			360		-2086	89	14	1
7.5						496	-1472		-28832				
8.0	-2	0	8	-108	-8			108		-2726	1	0	0
8.5						64	-352		-31692				
9.0	-2	2	10	-2	-8			108		-3378	-99	8	1
9.5						56	-112		-32525				
10.0	-14	28	84	-14	-56			360		-4030	-200	28	56
10.5						0	0		-32567				
TOTAL	-288	-1336	-424									7015	6132

r_1 = 334 #/in ; r_2 = -106 #/in

$$N_c = 187.7\sqrt{\frac{7015 \times 10^4}{613.2 \times 10^6}} = 6430 \text{ rpm}$$

132

Ralph B. Bates
12-4-63

```
C     CRITICAL SPEED, STRESS, AND BEARING REACTION CALCULATIONS
C             FOR A GENERAL SHAFT USING NUMERICAL INTEGRATION
C                               **********
C     PROGRAM DOES NOT INCLUDE A TRACE ROUTINE
C     NO SWITCH SETTINGS
C                   *****  PART A-1  *****
C     **** DIMENSION STATEMENT
      DIMENSION A(102),B(102),C(102)
C     **** INITIALIZE ARRAYS AND SUMS LOCATIONS
50    DO 51 K=1,102
      A(K)=0.0
      B(K)=0.0
51    C(K)=0.0
      SUMA=0.0
      SUMB=0.0
      SUMC=0.0
C     ** READ IN EXTERNAL LOADS
60    READ    ,I0,W1,I1,W2,I2,W3,I3,W4,I4
      K=I1+1
      A(K)=W1
      K=I2+1
      A(K)=W2
      K=I3+1
      A(K)=W3
      K=I4+1
      A(K)=W4
      IF (I0)60,60,100
C     **** READ NO.1 INPUT DATA
100   READ    ,ISIZE,ICAL,ITYP,M,S1,S2,DX,HP,RPM
      PUNCH,ISIZE,ICAL,ITYP,M,S1,S2,HP,RPM,DX
```

```
C     **** CALCULATE TORQUE..T...THE TURNING MOMENT
      T=63025.*HP/RPM
      M=M+1
      S1=S1+1.
      S2=S2+1.
C     **** READ =2 INPUT DATA
110   READ    ,OD,XID,W,XIX,XIP,J,L
      J=J+1
      L=L+1
      A(1)=0.0
      IF (OD) 114,114,115
C     **** TO DET.IF IX,W ARE GIVEN
115   IF (W)111,111,112
C     **** CALCULATE SHAFT WEIGHT W=PI/4(OD**2-ID**2)*0.285*DX
111   W=0.7853982*(OD**2-XID**2)*0.285*DX
112   IF (XIX)113,113,120
C     **** CALCULATE THE MOMENT OF INERTIA OF THE CROSS SECTIONAL AREA,IX
C          WITH RESPECT TO THE RADIAL AXIS--IX=(OD**4-ID**4)*PI/64
113   XIX=(OD**4-XID**4)*0.049087385
C     **** CALCULATE THE MOMENT OF INERTIA OF THE CROSS SECTIONAL AREA,IP
C          WITH RESPECT TO THE LONGITUDINAL AXIS--IX=IY,SO IP=IX*IY=2*IX
      XIP=2.*XIX
C     **** CALCULATE C/IX(S),THE RADIUS TO THE OUTER SHAFT SURFACE DIVIDED
C          BY THE MOMENT OF INERTIA OF THE CROSS SECTIONAL AREA W.R.TO
C          RADIAL AXIS,IN.**-3
120   CDIX=(OD/2.)/XIX
C     **** CALCULATE C/IP(S),THE RADIUS TO THE OTHER SHAFT SURFACE DIVIDED
C          BY THE MOMENT OF INERTIA OF THE CROSS SECTIONAL AREA W.R. TO
C          THE LONGITUDINAL AXIS,IN**-3
      CDIP=(OD/2.)/XIP
      GO TO 116
C     **** CALCULATE LS(A(K)+LS)=LOAD PER INCREMENT INCLUDING EXTERNAL
C          LOADS WITHOUT MASS IN NORMAL DIRECTION. CONCENTRATED LOADS MUST
```

```
C         BE DISTRIBUTED OVER THE INCREMENT DELTA X. ACTUAL SHAFT AND
C         ROTOR LOADS ARE MORE OFTEN DISTRIBUTED THAN CONCENTRATED
C    **** DIRECTION OF LOAD , DOWN = -
114 CDIX=0.0
    CDIP=0.0
116 DO 130 I=J,L
    PUNCH,CDIX,OD,CDIP,XID
    A(I)=-(A(I)+W)
C    **** CALCULATE M2 (B(J))=MOMENTS ABOUT BEARING 2 STATION LINE
C         M2=LS*((S OF BEARING 2)-S)---WHERE S=STATION NO. SO THAT M2 IS
C         CALCULATED AT EACH STATION LINE
     S=I
    B(I)=A(I)*(S2-S)
C    **** CALCULATE M1 (C(J))=MOMENTS ABOUT BEARING 1 STATION LINE
C         M1=LS*((S OF BEARING 1)-S)---WHERE S=STATION NO. SO THAT M1 IS
C         CALCULATED AT EACH STATION LINE
    C(I)=A(I)*(S1-S)
    SUMC=SUMC+C(I)
    SUMB=SUMB+B(I)
130 SUMA=SUMA+A(I)
    IF (L-M)110,140,140
140 I=S1
    A(I)=A(I)+SUMB/(S1-S2)
    SUMA=SUMA+SUMB/(S1-S2)
    I=S2
    A(I)=A(I)+SUMC/(S2-S1)
    SUMA=SUMA+SUMC/(S2-S1)
C    **** PUNCH OUTPUT DATA---S,W,M2,M1
    DO 150 I=1,M
    S1=I-1
150 PUNCH,S1,A(I),B(I),C(I)
C    **** PUNCH TOTALS OF M2,M1,W
    PUNCH,T,SUMB,SUMC,SUMA
```

```
      GO TO 50
      END
C                    *****  PART A-2  ****
C     **** DIMENSION STATEMENT
      DIMENSION A(102),P(102),X(102)
C     **** INITIALIZE ARRAYS AND SUMS LOCATIONS
100   DO 101 K=1,102
      P(K)=0.0
101   X(K)=0.0
      SUMV=0.0
      SUMD=0.0
      SUME=0.0
      READ   ,ISIZE,ICAL,ITYP,M
      READ   ,S1,S2,HP,RPM
      READ   ,DX
      PUNCH,ISIZE,ICAL,ITYP,M,S1,S2,HP,RPM,DX
C     **** STORE C/IX  IN X(I) ARRAY
C     **** STORE C/IP  IN P(I) ARRAY
      M=M+1
      DO 102 I=1,M
      READ   ,CDIX,OD,CDIP,XID
      X(I)=CDIX
102   P(I)=CDIP
      DO 103 I=1,M
103   READ   ,S,A(I),Z,ZU
      READ   ,T,SUMB,SUMC,SUMA
      S=0.0
      V1=0.0
      D1=0.0
      DO 120 I=1,M
C     **** CALCULATE SHEAR A STATION (S+1/2)---V(S+1/2)=V(S-1/2)+WS*DX
      V2=V1+A(I)*DX
C     **** CALCULATE BENDING MOMENT AT STATION (S+1/2)----
```

```
C         ---M(S+1/2)=M(S-1/2)+V(S-1/2)*DX+1/2*WS*DX**2
      D2=D1+0.5*DX*(V1+V2)
C     **** CALCULATE BENDING MOMENT AT STATION (S)---- M(S)=E(I)
C         ---M(S)=M(S-1/2)+1/2*V(S-1/2)*DX+1/8*WS*DX**2
      E1=D1+0.125*DX*(3.*V1+V2)
C     **** CALCULATE BENDING TENSILE STRESS,PSI---SBS=MS*C/IX
      SB=E1*X(I)
      IF (SB)104,105,105
104   SB=-SB
C     **** CALCULATE SHAFT TORSIONAL SHEAR STRESS,PSI---STS=T*C/IP
105   ST=T*P(I)
C     **** CALCULATE THE SHAFT MAX. COMBINED SHEAR STRESS,PSI---
C         ---SCS=(ST**2+(SBS/2)**2)**.5
      SC=(ST**2+0.25*SB**2)**0.5
C     **** PUNCH OUTPUT DATA---V(S+1/2),M(S+1/2),MS,C/IX,SBS,T,C/IP,STS,SC
      S=I-1
      IF (X(I))110,110,111
110   Q=0.
111   PUNCH,S,E1,X(I),SB,Q,P(I),ST,SC
      Q=T
      S=S+.5
      PUNCH,S,V2,D2
      V1=V2
120   D1=D2
      GO TO 100
      END
C     REARRANGE THE OUTPUT DATA OF 62-0015A1,2 INTO TABLE
      DIMENSION A(102),B(102),C(102),T(102),V(102),D(102)
100   READ,ISIZE,ICAL,ITYP,M
      READ,S1,S2,HP,RPM
      READ,DX
      PUNCH,ISIZE,ICAL,ITYP,M,S1,S2,HP,RPM,DX
      K=M+1
```

```
C     **** READ,C/IX,OD,C/IP,ID
      DO 110 I=1,K
  110 READ,R1,R2,R3,R4
C     **** READ S,WS,M2,M1
      DO 120 I=1,K
      READ,S  ,R1,R2,R3
      PUNCH,S  ,R1,R2,R3
      S=S+.5
  120 PUNCH,S
      S=S-S
      READ,R4,SUMB,SUMC,SUMA
      PUNCH,S  ,SUMA,SUMB,SUMC
      R1=SUMB/(S1-S2)
      R2=SUMC/(S2-S1)
      READ,ISIZE,ICAL,ITYP,M
      READ,S1,S2,HP,RPM
      READ,DX
C     ****  A(I)=C/IP,B(I)=STS,C(I)=SCS
      DO 130 I=1,K
      READ,S1,SM,CDIX,SB
      READ,T(I),A(I),B(I),C(I)
      READ,S,V(I),D(I)
  130 PUNCH,S1,SM,CDIX,SB
      DO 131 I=1,K
      S=I
      S=S-.5
  131 PUNCH,S,V(I),D(I)
      S=S-S
      PUNCH,S,R1,R2
      DO 140 I=1,K
  140 PUNCH,T(I),A(I),B(I),C(I)
      GO TO 100
      END
```

```
C                  *****  PART B-1  *****
C    **** DIMENSION STATEMENT
     DIMENSION A(102),X(102),Y(102)
C    **** INITIALIZE WORING AREAS
90   DO 91 I=1,102
     A(I)=0.0
     X(I)=0.0
91   Y(I)=0.0
     V1=0.0
     D1=0.0
     F1=0.0
     G1=0.0
     SUMB=0.
     SUMC=0.
C    **** READ INPUT DATA NO.1
     READ    ,ISIZE,ICAL,ITYP,M,S1,S2,HP,RPM,DX
     PUNCH,ISIZE,ICAL,ITYP,M,S1,S2,HP,RPM,DX
C    **** READ C/IX (CDIX)
     M=M+1
     S1=S1+1.
     S2=S2+1.
     DO 101 I=1,M
     READ   ,X(I),OD,CP,CX
     IF (X(I))101,101,100
100  X(I)=(OD/2.)/X(I)
101  CONTINUE
C    **** READ STATION,LS,M2,M1
     DO 105 I=1,M
     READ    ,S,A(I),B,C
     S=S+1.
C    **** CALCULATE LS=LOAD PER INCREMENT EXCLUDING EXTERNAL LOADS
C         WITHOUT MASS AND WITH LOADS REVERSED BETWEEN BEARINGS.
C         CONCENTRATED LOADS MUST BE DISTRIBUTED OVER THE INCREMENT DX.
```

```
C         ACTUAL SHAFT AND ROTOR LOADS ARE MORE OFTEN DISTRIBUTED THAN
C         CONCENTRATED.
      IF (S-S2)102,104,104
  102 IF (S-S1)104,104,103
  103 A(I)=-A(I)
      B=-B
      C=-C
  104 SUMB=SUMB+B
      PUNCH,I,A(I)
  105 SUMC=SUMC+C
      READ,T,B,C,SA
      K=S1
      B=A(K)-B/(S1-S2)
      A(K)=B+SUMB/(S1-S2)
      PUNCH,K,A(K)
      K=S2
      C=A(K)-C/(S2-S1)
      A(K)=C+SUMC/(S2-S1)
      PUNCH,K,A(K)
      DO 170 K=1,M
C     **** CALCULATE SHEAR AT STATION (S+1/2)---V(S+1/2)=V(S-1/2)+WS*DX
      V2=V1+A(K)*DX
C     **** CALCULATE BENDING MOMENT AT STATION (S+1/2)---
C             ---M(S+1/2)=M(S-1/2)+V(S-1/2)*DX+1/2*WS*DX**2
      D2=D1+0.5*DX*(V1+V2)
C     **** CALCULATE THE AREA UNDER THE (M/EIZ) CURVE AT STATION (S+1/2)--
C             --F(S+1/2)=F(S-1/2)+(2M(S-1/2+1/2V(S-1/2)+M(S+1/2))*DX/(3*EIX)
      IF (X(K))110,110,120
  110 F2=F1
      G2=G1+F1*DX
      Y(K)=G1+.5*DX*F1
      GO TO 130
  120 F2=F1+(2.*D1+0.5*V1+D2)*DX/(3.*X(K)*30.E06)
```

```
C     **** CALCULATE THE AREA UNDER THE F CURVE AT STATION (S+1/2)---
C          G(S+1/2)=G(S-1/2)+DX*(F(S-1/2)+(12M(S-1/2)+DX(3V(S-1/2)
C                        +V(S+1/2)))*DX/24EI)
      G2=G1+DX*(F1+(12.*D1+DX*(3.*V1+V2))*DX/(24.*X(K)*30.0E06))
C     **** CALCULATE THE AREA UNDER THE F CURVE AT STATION S---
C     ---G =G(S-1/2)+1/2DX(F1+(M(S-1/2)+1/48*DX(7V(S-1/2)+V(S+1/2)
      Y(K)=G1+.5*DX*(F1+(D1+.02083333*DX*(7.*V1+V2))*DX/(4.*X(K)*30.E06))
C     **** CALCULATE DEFLECTION AT STATION S---GS1=G1,G=GS2
130   PUNCH,V1,V2,D1,D2,F1,F2,G1,G2,Y(K)
      V1=V2
      D1=D2
      F1=F2
      G1=G2
170   CONTINUE
      K=S1
      G1=Y(K)
      A(K)=B
      K=S2
      G2=Y(K)
      A(K)=C
C     **** CALCULATE ABS. SUM OF WS*YS--WS=WS*DX
      SUMP=0.0
      SUMK=0.0
      DO 240 K=1,M
      S=K
      Y(K)=Y(K)-G1-(G2-G1)*(S-S1)/(S2-S1)
      SUMP=SUMP+(A(K)*DX)*Y(K)
      SUMK=SUMK+(A(K)*DX)*Y(K)*Y(K)
      PUNCH,SUMP,SUMK,Y(K)
      IF (SUMP)210,220,220
210   SUMP=-SUMP
220   IF (SUMK)230,240,240
230   SUMK=-SUMK
```

```
240 CONTINUE
C     **** CALCULATE THE FIRST CRITICAL SPEED,RPM
C           NC=187.7*(SUM OF WY/SUM OF WY**2)**.5
      XNC=187.7*(SUMP/SUMK)**.5
C     **** PUNCH OUTPUT--YS AND NC
      R1=SUMB/(S1-S2)
      R2=SUMC/(S2-S1)
      PUNCH,R1,R2,XNC
      DO 250 I=1,M
      K=I-1
250   PUNCH,K,Y(I)
      GO TO 90
      END
```

THREE DIMENSIONAL SURFACE FIT

M-151

December 9, 1963

D. G. Kitzinger

TABLE OF CONTENTS

INTRODUCTION

An ever present problem in engineering is the need to represent the three dimensional array of data by a mathematical equation. Many physical quantities can be described only in three-space. Computer storage limitations prevent table look-up of a large number of these quantities, expecially if their range of values is extensive. M-151 uses nth order multiple interpolation together with extensive transformation of variables to describe most three dimensional functions. The method used has the advantage of separating the three dimensional characteristics of the array into components described in two dimensions, hence easy to define by the user. Since many functions (exponentials, sinusoidals, etc.) are not best described by the polynomials used in interpolation, the first approximation to the function might be sadly deficient. Better successive approximations to the function can be easily made using the appropriate output option accompanied by transformation of variables. An error analysis assists the programmer in choosing the best fit of a function, and in the case of second and third order interpolation, the best fit is selected from a number of possible fits.

ANALYSIS

Let us represent a three dimensional array by $x = f[Y(y), Z(z)]$, where $Y(y)$ and $Z(z)$ are polynomials in y and z, respectively. Let us limit the order of these polynomials to n and m so that

$$Y(y) = a_0 + a_1 y + \cdot \cdot \cdot + a_n y^n = \sum_{i=0}^{n} a_i y^i \qquad (1)$$

$$Z(z) = b_0 + b_1 z + \cdot \cdot \cdot + b_m z^m = \sum_{j=0}^{m} b_j z^j \qquad (2).$$

At the ith value of y,

$$Z_i = \sum_{j=0}^{m} b_{ji} z^j$$

Define:

$$x = f(Y, Z_i) = \sum_{i=0}^{n} Z_i y^i = \sum_{i=0}^{n} \sum_{j=0}^{m} b_{ji} z^j y^i \qquad (3).$$

$$\text{Then, } a_i = \sum_{j=0}^{m} b_{ji} z^j \qquad (4).$$

Note that x is a smooth function of y and z with (n + 1)(m + 1) coefficients and hence requiring the same number of known members of the array. In general, an (n + 1)(m + 1) order matrix would have to be solved to obtain values for the unknown coefficients.

Referring to equations (3) and (4), note that both a_i and x vary in the same way with changes in z. Thus, by studying the behavior of x with changing z, the behavior of a_i with changing z is also known. By holding y constant, a_i is a function of z alone. If values of a_i are known at each of (m + 1) values of z, the b_{ji} values in equation (4) are defined by the appropriate matrix solution. If z is held constant at z_i and if (n + 1) values for $x = g(y, z_i)$ are known, (n + 1) a_{ij} coefficients are determined. Define a restriction that allows easy determination of the a_{ij} and b_{ji} coefficients: y takes on, at most, (n + 1) values; z takes on, at most, (m + 1) values; and the value of x is defined for all allowed values of y and z. At particular z_i,

$$x_i(y, z_i) = a_{0i} + a_{1i}y + \cdots + a_{ni}y^n \qquad (5),$$

which is a form that allows solution for (n + 1) values of a_{ij}, when the (n + 1) values of y are used.

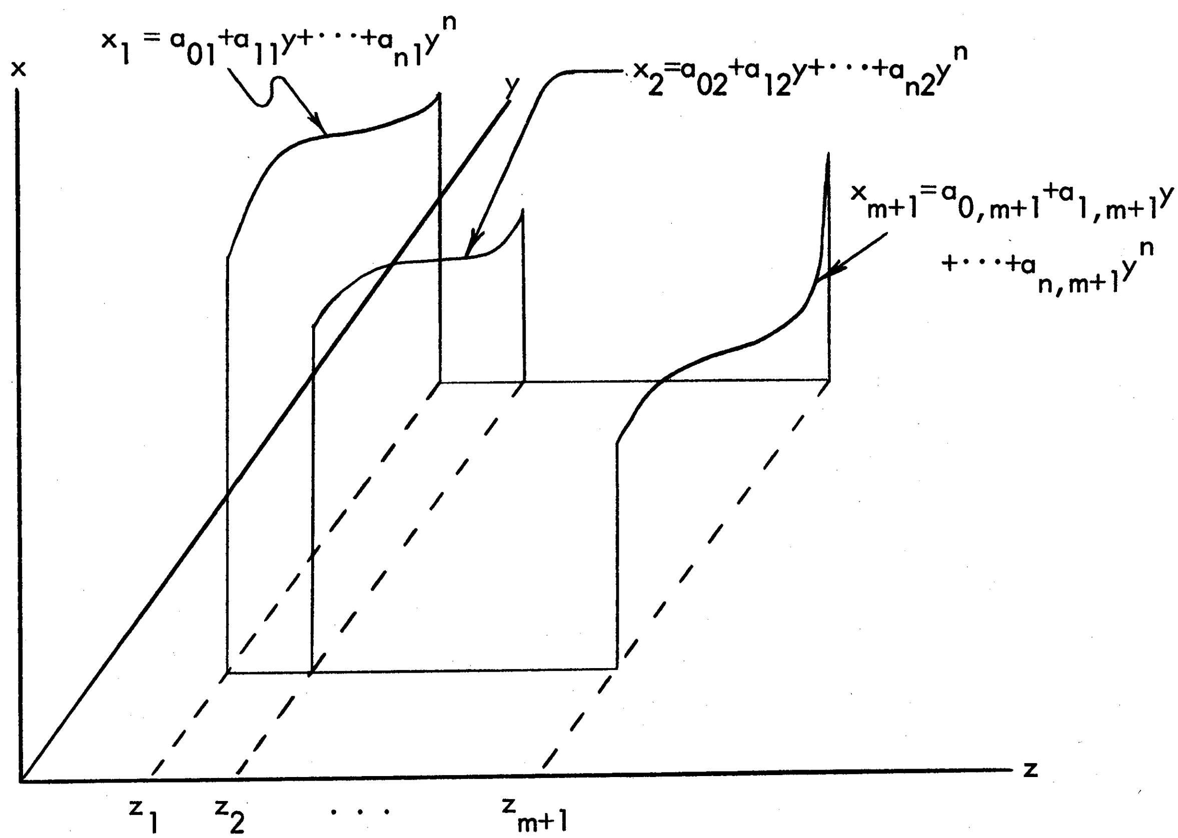


FIGURE 1

Choose the (m + 1) values of z successively, then an array of $a_i(z_j)$ values results: $i = 0, 1, 2, \cdots, n$; $j = 0, 1, 2, \cdots, m$. Fit the (m + 1) values of a_i by

$$a_i(z) = b_{0i} + b_{1i}z + \cdots + b_{mi}z^m \qquad (6),$$

with the result that the b_{ji} coefficients in equation (3) are evaluated. Equation (3) defines a function of y and z that passes through (n + 1)(m + 1) data points. Furthermore, the function of y and z is smooth, and if planes parallel to the y or z axis and perpendicular to the yz plane are chosen, they intersect the x = g(y, z) surface in lines described by polynomials of order n and m, respectively. For practical purposes, n and m may be selected by choosing their values dependent on the study of the z = constant plane intersections and the y = constant plane intersections with the surface x = g(y, z). Note that experimental data are often determined most easily by holding all variables constant except one. This procedure guarantees compliance with the restrictions imposed regarding values of x at each combination of allowable y and z.

The usefulness of the assumed fit given by equation (3) is limited when the surface to be fit is best described using exponential, trigonometric, fractional exponent and hyperbolic functions. Although it is true that most useful engineering functions can be described by infinite series approximated by the form given in equation (3), it would be better to use the appropriate functional transformation directly. For instance, assume that a surface is known to be of the form $x = f(y) = \left[\sin(7y + 3)\right]\left[a_0 + a_1 y\right]$. Let $x_1 = f(y_1)$ and $x_2 = f(y_2)$ and define $g(y) = \sin(7y + 3)$, $x_1' = x_1/g(y)$, $x_2' = x_2/g(y)$ and, in general, $x' = x/g(y)$. Then fit $x' = a_0 + a_1 y$ by a first order polynomial using x_1' and x_2' as data points.

Having determined the values of a_0 and a_1, x is defined by $x'g(y)$. The expression for x is valid not only at y_1 and y_2, but over the entire region in which g(y) is a suitable transformation. It should be noted that the data points x_1 and x_2 could have been fitted by $x = a_0 + a_1 y$ without considering g(y). However, only the two data points and the points where $(7y + 3) = \sin^{-1}(1)$ would be satisfied by $x = a_0 + a_1 y$.

Theoretically, provided the matrix solution is exact, the polynomial fit given by equation (3) should fit exactly the (n + 1)(m + 1) primary data points. If the function to be described is of the form $x = g(y) \sum_{i=0}^{n} \sum_{j=0}^{m} b_{ji} z^j y^i$, a study of the primary data points will not give information concerning the suitability of g(y) as the appropriate transformation of variables, since all primary data points are fitted, independent of the form of g(y). Hence, secondary data points not used in the determination of the b_{ji} coefficients must be studied to arrive at g(y). M-151 allows secondary data points for either arbitrary y or z, but not both arbitrary. Either y or z must be an "allowed" value, in the sense that defines primary data points. The other coordinate, z or y respectively, may be arbitrary.

To help decrease the importance of the user in deciding on allowed values of y and z for primary data points, over-definition of the problem is allowed in the cases of second and third order fitting. As many as seven allowed values may be chosen and the program will use all combinations of seven points taken three and four at a time, respectively for second and third order fits.

SURFACE FITTING TECHNIQUE

1. Select the dependent variable, x

2. Plot families of curves
   a. y = constant plotted on the xy plane
   b. z = constant plotted on the xy plane

3. Assume n and m for fitting y and z such that $x = \sum_{i=0}^{n} a_i y^i$ and $x = \sum_{i=0}^{m} b_i z^i$ on the xz and xy planes, respectively. If there is uncertainty concerning the best value of n and m, all possible values should be used in separate cases, using the average error and maximum error features of the code to decide between cases. It is not crucial that the very best case be used, if transformation of variables can be used to force a best fit. For instance, x = sin(7y + 3) is fit best in certain regions by $x = \sum_{i=0}^{2} a_i y^i$ without transformation of variables, but with transformation of variables x/sin(7y + 3) = x' = 1 is fit exactly by a zero order fit: $x' = a_0 = 1$. In the interest of developing a technique independent of curve fitting experience, it is better to use $x = \sum_{i=0}^{2} a_i y^i$, provided that the region of definition will remain small. However, any curve fitting experience at the user's disposal should be used.

4. To effect more accurate fits, use the output option in which transformed data is output ready for further transformations. This option prints the transformed values of x (for example, x/g(y) is printed if the transformation is x' = x/g(y) and x is printed if there is no transformation), the fit of the transformed variable, the difference including sign between x' and its fit, and the ratio x'/(fit of x'). Referring to figure 2: x = f(y), the fit of x = g(y), and h(y) = x - (fit of x).

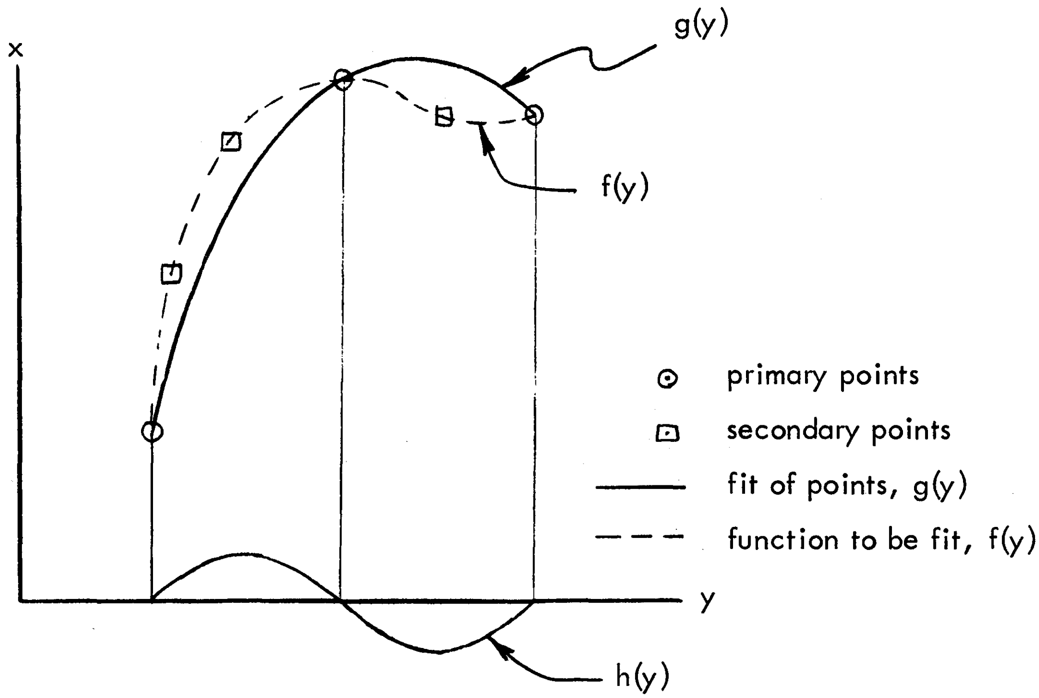


FIGURE 2

The transformation that effects an exact solution is f(y) = h(y) + g(y). Since h(y) can be approximated by a sinusoidal with appropriate amplitude and period, the sinusoidal transformation is made on all data points using the program to make the change: x' = x - h(y). Upon rerunning this case with the transformation included, the printout will include x', the fit of x', x' - fit of x', and the ratio x'/(fit of x'). Once again, h'(y) = x' - fit of x' = f'(y) - g'(y). To determine our fit, x' = h'(y) + g'(y) and x = x' + h(y) = h(y) + h'(y) + g'(y). Correspondingly, the ratio x/(fit of x) may be used by plotting this ratio and describing it just as h(y) was described. Code capability allows as many as ten transformations to be made. It is necessary, in order to achieve an improvement in the fit, that the transformation of variables be such that the original data points will not be seriously shifted by the transformation. As an example, consider the translation without distortion of the fit of the array to more closely approximate secondary data points in the array. If this translation is fit once again, the fit of the translated array will be parallel to the original fit with no improvement over the original fit in regard to secondary data points in the array.

TRANSFORMATION OF VARIABLES

The utility of transformation of variables was demonstrated in the section on surface fitting technique. Basically, there are two types of transformations available in this code: functional transformations that deal with the manipulation of complete generated functions of x, y and z; and transformations of individual variables x, y and z to form the functions used in functional transformation. Functional transformations available are listed in Table 1 and transformation of variables are listed in Table 2.

TABLE 1

FUNCTIONAL TRANSFORMATIONS

NVT	NMODE = 1	NMODE = 2
1	None	None
2	$f(x)$	$f(x)$
3	$xg(y)$	$xh(z)$
4	$x + g(y)$	$x + h(z)$
5	$x/g(y)$	$x/h(z)$
6	$x[g(y) + k(y)]$	$x[h(z) + l(z)]$
7	$x/[g(y) + k(y)]$	$x/[h(z) + l(z)]$
8	$xg(y)h(z)$	$x/[g(y)h(z)]$
9	$x[g(y) + h(z)]$	$x/[g(y) + h(z)]$
10	$xg(y)/h(z)$	$xh(z)/g(y)$

TABLE 2

TRANSFORMATION OF VARIABLES

Let v take on the value x, y or z. v shall be transformed.

TRANSFORMATION	INVERSE TRANSFORMATION
NT = 1 None	None
NT = 2 $a + be^{cv+d}$	$\left(\ln\left[\frac{(a + be^{cv+d}) - a}{b}\right] - d\right)\Big/c$
NT = 3 $a + bv^{c}$	$\left[\frac{(a + bv^{c}) - a}{b}\right]^{1/c}$
NT = 4 a + b sin (cv + d)	$\frac{\sin^{-1}\left(\frac{[a + b \sin (cv + d) - a]}{b}\right) - d}{c}$
NT = 5 $a + bc^{dv}$	$\frac{\ln\left[\frac{(a + bc^{dv}) - a}{b}\right]}{d \ln c}$
NT = 6 a + b ln (cv + d)	$\frac{\exp\left(\frac{[a + b \ln (cv + d)] - a}{b}\right) - d}{c}$

(Table 2, Continued)

NT = 7 $a + b \ln_c (v + d)$	$c^{\left(\frac{\left[a + b \ln_c (v + d)\right] - a}{b}\right)} - d$
NT = 8 $a + b \ln \frac{c}{v + d}$	$ce^{-\left(\frac{\left[a + b \ln \frac{c}{v + d}\right] - a}{b}\right)} - d$
NT = 9 $a + \frac{b}{(v + c)^d}$	$\left\{\frac{b}{\left[a + \frac{b}{(v + c)^d}\right] - a}\right\}^{1/d} - c$
NT = 10 $a + b \sinh(cv + d)$	$\left(\ln \left[\left\|1 + \left(\frac{[a + b \sinh(cv + d)] - a}{b}\right)^2\right\|^{1/2} + \frac{[a + b \sinh(cv + d)] - a}{b}\right] - d\right) \Big/ c$
NT = 11 $a + b \cos(cv + d)$	$\frac{\cos^{-1}\left(\frac{[a + b \cos(cv + d)] - a}{b}\right) - d}{c}$
NT = 12 $a + b \tan^{-1}(cv + d)$	$\frac{\tan\left(\frac{[a + b \tan^{-1}(cv + d)] - a}{b}\right) - d}{c}$
NT = 13 $a + b \cosh(cv + d)$	$\left(\pm \ln \left[\left\| -1 + \left(\frac{[a + b \cosh(cv + d)]}{b} - a\right)^2\right\|^{1/2} + \frac{[a + b \cosh(cv + d)] - a}{b}\right] - d\right) \Big/ c$

with the restriction that $(cv + d) > 1$.

INPUT PARAMETERS

AT, BT, CT, DT — These are the coefficients a, b, c, and d defined in Table 2. Referring to Table 1, these coefficients apply to each single function transformation, but apply only to g(y) if NMODE = 1 or h(z) if NMODE = 2, when a transformation is used that involves two distinct changes of variables. (An example is NVT = 6 and NMODE = 1 from Table 1, where g(y) and k(y) are distinct changes of variables having distinct characteristic coefficients.)

AT2, BT2, CT2, DT2 — These are the coefficients a, b, c, and d defined in Table 2. Referring to Table 2, these coefficients apply only when a two function transformation is used. When NMODE = 1, only k(y) or h(z) is described; and when NMODE = 2, only l (z) or g(y) is described.

DC — x values defined as primary (see analysis). Values of x are allowed only for (YC, ZC) coordinate pairs. To fill input cards, nonvalid values of DC are set equal to 1.000E-9. The order in which DC values are placed on cards is defined by the order in which ZC values are read. YC is held constant for any DC input card.

DY — x values defined as secondary (see analysis). Arbitrary values of z are allowed for each YC value. To fill input cards, nonvalid values of DY are set equal to 1.000E-9. The order in which DY values are placed on cards is defined by the order in which Z values are read. YC is held constant for any DY card and packets of DY and Z cards are read in the same order as YC values were read.

DZ — x values defined as secondary (see analysis). Arbitrary values of y are allowed for each ZC value. To fill input cards, nonvalid values of DZ are set equal to 1.000E-9. The order in which DZ values are placed on cards is defined by the order in which Y values are read. ZC is held constant for any DZ card and packets of DZ and Y cards are read in the same order as ZC values were read.

KO — output selector: KO = 1 - only the b_{ij} coefficients are printed; KO = 2 - b_{ij} coefficients, x, y, z, fit of x, x - (fit of x), average per centage error, and maximum per centage error are printed; KO = 3 - same as KO = 2 with the addition of the transformed x = x', fit of x', x' - (fit of x'), and x'/(fit of x') printed. Table 3 shows a fourth order function fitted with a first order approximation using KO = 1, 2, and 3.

MO — order of polynomial Z defined in equation (2), where MO = m. MO may be as large as ten, but calculational error is such that it is suggested that MO be kept less than seven.

MC — (MO + 1) except for second and third order fits where MC may be as large as seven. There are MC values of ZC. MC $\leq$ 12

NC — (NO + 1) except for second and third order fits where NC may be as large as seven. There are NC values of YC. NC $\leq$ 12

NC2 — the number of secondary values of x to be read in for each value of YC. There are NC2 arbitrary z values. NC2 is $\leq$ 18.

NC3 — the number of secondary values of x to be read in for each value of ZC. There are NC3 arbitrary y values. NC3 is $\leq$ 18.

NE — operator that governs the type of error analysis used to distinguish between possible fits when there are multiple fits allowed. Consequently, NE is useful only for second and third order fits where over-definition is possible (see analysis).

NE	TYPE FIT
1	numerical average
2	product
3	numerical average X product
4	average^2/maximum error
5	product with minimum error stripped out
6	average with maximum stripped out
7	product with maximum stripped out
8	product with maximum and minimum stripped out

NMODE — an operator, together with NVT, that is used to define transformations in Table 1.

NO — order of polynomial Y defined in equation (1), where NO = n. NO may be as large as ten, but calculational error is such that it is suggested that NO be kept less than seven.

NPASS — number of transformations allowed sequencially. Referring to the section on techniques, NPASS takes on the value 2 if x" transformations are made.

NT — defined in Table 2 and treated similar to coefficients AT, BT, CT and DT.

NT2 – defined in the same way as NT but treated similar to coefficients AT2, BT2, CT2 and DT2.

NVT — an operator, together with NMODE, that is used to define transformations in Table 1.

Y — an arbitrary value of y which, together with a ZC, defines a secondary point, x.

YC — the value of y used to define primary data points in the DC array and secondary data points in the DY array.

Z — an arbitrary value of z which, together with a YC, defines a secondary point, x.

ZC — the value of z used to define primary data points in the DC array and secondary data points in the DZ array.

INPUT TECHNIQUE

Cards 3 and 4 below are read in repetitively in packages consisting of one card 3 followed by one card 4 provided MC $\leq$ 6 or by two card 4s when MC > 6. NC of these packages are read in. Card 5 is repeated MC times. Cards 6, 7 and 8 form a package that is read in NC times. If NC2 = 0, cards seven and eight are not included in the package. If 0 < NC2 $\leq$ 6, just one card 7 and card 8 follow NC2. If 6 < NC2 $\leq$ 12, the sequence is 6, 7, 8, 7, 8. If 12 < NC2 $\leq$ 18, the sequence is 6, 7, 8, 7, 8, 7, 8. Cards 9, 10 and 11 form a package that is read in MC times. This package is handled the same as the 6, 7 and 8 package. Cards 13, 14, 15 and 16 form a package that is read in NPASS times. If NVT (I) < 6, cards 14 and 15 are omitted.

Card	Description	Format
1	55 symbol comments card	55H
2	NO, MO, NC, MC, NE, KO	6I4
3	YC(I)	E8.4
4	DC(I,J), DC(I, J + 1), DC(I, J + 2), DC(I, J + 3), DC(I, J + 4), DC(I, J + 5)	6E8.4
5	ZC(J)	E8.4
6	NC2	I4
7	DY(I, J), DY(I, J + 1), DY(I, J + 2), DY(I, J + 3), DY(I, J + 4), DY(I, J + 5)	6E8.4
8	Z(I, J), Z(I, J + 1), Z(I, J + 2), Z(I, J + 3), Z(I, J + 4), Z(I, J + 5)	6E8.4
9	NC3	I4
10	DZ(J, I), DZ(J +1, I), DZ(J + 2, I), DZ(J + 3, I), DZ(J + 4, I), DZ(J + 5, I)	6E8.4
11	Y(J,I), Y(J + 1, I), Y(J + 2, I), Y(J + 3, I), Y(J + 4, I), Y(J + 5, I)	6E8.4
12	NPASS	I4
13	NVT(I), NMODE(I), NT(I)	3I4
14	NT2(I)	I4
15	AT2(I), BT2(I), CT2(I), DT2(I)	4E8.4
16	AT(I), BT(I), CT(I), DT(I)	4E8.4

KO = 1

XTHREE DIMENSIONAL SURFACE FIT CODE, M-151

M-151 DEMONSTRATION PROBLEM, ORDER=1X1

COEFFICIENTS OF THE POWER SERIES IN Z

```
B( 1, 1)     =     -20.3577E-01
B( 1, 2)     =      10.8032E-04


B( 2, 1)     =      17.8064E-04
B( 2, 2)     =     -65.3496E-08
```

KO = 2

XTHREE DIMENSIONAL SURFACE FIT CODE, M-151

M-151 DEMONSTRATION PROBLEM, ORDER=1X1

COEFFICIENTS OF THE POWER SERIES IN Z

```
B( 1, 1)     =     -20.3577E-01
B( 1, 2)     =      10.8032E-04


B( 2, 1)     =      17.8064E-04
B( 2, 2)     =     -65.3496E-08
```

```
                                   TABULATED   CALCULATED
        Y           Z                  X            X        DIFFERENCE

50.0000E-00 27.0000E+02 82.1100E-02 82.1100E-02    .0000E-[illegible]
            50.0000E+02 32.3000E-01 32.2999E-01 10.0000E-08
10.0000E+02 27.0000E+02 82.1100E-02 82.1099E-02 10.0000E-08
            50.0000E+02 17.8900E-01 17.8899E-01 20.0000E-08
50.0000E-00 30.0000E+02 90.9100E-02 11.3530E-01-22.6204E-02
            33.0000E+02 10.1400E-01 14.4950E-01-43.5508E-02


            37.0000E+02 11.9000E-01 18.6844E-01-67.8447E-02
            41.0000E+02 14.7100E-01 22.8738E-01-81.6386E-02
            47.0000E+02 23.8000E-01 29.1579E-01-53.5795E-02
            49.0000E+02 29.0900E-01 31.2526E-01-21.6265E-02
10.0000E+02 30.0000E+02 89.5400E-02 94.7347E-02-51.9478E-03
            33.0000E+02 97.2700E-02 10.7359E-01-10.0895E-02
            37.0000E+02 10.9000E-01 12.4192E-01-15.1926E-02
            41.0000E+02 12.3700E-01 14.1025E-01-17.3256E-02
            47.0000E+02 15.5600E-01 16.6275E-01-10.6752E-02
            49.0000E+02 17.0400E-01 17.4691E-01-42.9172E-03
```

AVERAGE PERCENTAGE ERROR = .12221486E+02

MAXIMUM PERCENTAGE ERROR = .36310770E+02

KO = 3

XTHREE DIMENSIONAL SURFACE FIT CODE, M-151

M-151 DEMONSTRATION PROBLEM, ORDER=1X1

COEFFICIENTS OF THE POWER SERIES IN Z

```
B( 1, 1)     =     -20.3577E-01
B( 1, 2)     =      10.8032E-04


B( 2, 1)     =      17.8064E-04
B( 2, 2)     =     -65.3496E-08
```

THIS OPTION OUTPUTS THE FOLLOWING ON ALTERNATE LINES--

```
ORIGINAL DATA
                                       TABULATED    CALCULATED
      Y              Z                     X             X          DIFFERENCE

TRANSFORMED DATA
   TABULATED   CALCULATED
      XT           XC            XT-XC         XC/XT
```

```
50.0000E-00 27.0000E+02 82.1100E-02 82.1100E-02    .0000E-99
82.1100E-02 82.1100E-02    .0000E-99 10.0000E-01
            50.0000E+02 32.3000E-01 32.2999E-01 10.0000E-08
32.3000E-01 32.2999E-01 10.0000E-08 99.9999E-02
10.0000E+02 27.0000E+02 82.1100E-02 82.1099E-02 10.0000E-08
82.1100E-02 82.1099E-02 10.0000E-08 99.9999E-02
            50.0000E+02 17.8900E-01 17.8899E-01 20.0000E-08
17.8900E-01 17.8899E-01 20.0000E-08 99.9999E-02
50.0000E-00 30.0000E+02 90.9100E-02 11.3530E-01-22.6204E-02
            33.0000E+02 10.1400E-01 14.4950E-01-43.5508E-02
            37.0000E+02 11.9000E-01 18.6844E-01-67.8447E-02
            41.0000E+02 14.7100E-01 22.8738E-01-81.6386E-02
            47.0000E+02 23.8000E-01 29.1579E-01-53.5795E-02
            49.0000E+02 29.0900E-01 31.2526E-01-21.6265E-02
10.0000E+02 30.0000E+02 89.5400E-02 94.7347E-02-51.9478E-03
            33.0000E+02 97.2700E-02 10.7359E-01-10.0895E-02
            37.0000E+02 10.9000E-01 12.4192E-01-15.1926E-02
            41.0000E+02 12.3700E-01 14.1025E-01-17.3256E-02
            47.0000E+02 15.5600E-01 16.6275E-01-10.6752E-02
            49.0000E+02 17.0400E-01 17.4691E-01-42.9172E-03
```

AVERAGE PERCENTAGE ERROR = .12221486E+02

MAXIMUM PERCENTAGE ERROR = .36310770E+02

TABLE 3

156

SAMPLE PROBLEM

A practical problem involving the fit of hydrogen thermal conductivity as a function of temperature and pressure was met with success using M-151. The technique of regression analysis proved unwieldy in solving the same problem; lack of success being attributed to inexperience in picking suitable form for the fit plus the inherent limitations of regression analysis, such as inaccuracies resulting from the need to solve for large numbers of coefficients simultaneously. Some of the data points used are presented in Table 4. Referring to the section on Surface Fitting Technique, thermal conductivity was made the independent variable, pressure = y and temperature = z. The families of curves were not plotted since it was decided to choose many combinations of n and m and study the resulting error analysis. Table 5 presents input used when n = m = 4. Step 4 in the technique section was not used extensively since good results were achieved with the 4 x 4 order fit. Because of the volume of output with the 4 x 4 case, only the error analysis will be presented together with the errors from other cases in Table 6. Table 3 presents examples of output available.

TABLE 4

THERMAL CONDUCTIVITY OF HYDROGEN

($X10^6$)

TEMPERATURE, °R

Pressure, psi. P \ T	2700	3100	3500	3900	4300	4600	4800	5000
50	.8211	.9425	1.094	1.312	1.687	2.166	2.626	3.230
100	.8211	.9352	1.073	1.262	1.571	1.947	2.301	2.760
200	.8211	.9277	1.041	1.212	1.451	1.721	1.965	2.273
300	.8211	.9264	1.047	1.203	1.430	1.682	1.907	2.191
400	.8211	.9250	1.043	1.194	1.409	1.643	1.850	2.109
500	.8211	.9237	1.039	1.184	1.388	1.604	1.793	2.026
700	.8211	.9210	1.031	1.166	1.345	1.526	1.678	1.862
1000	.8211	.9204	1.029	1.159	1.327	1.491	1.627	1.789

TABLE 5

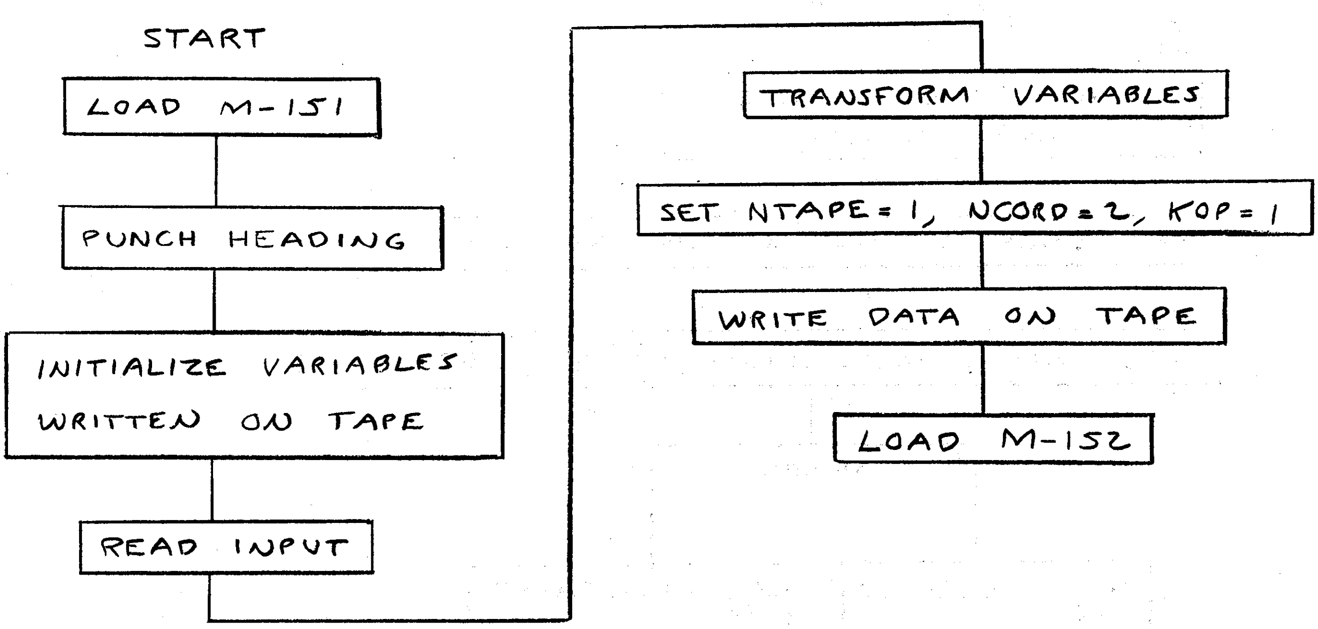
```
M-151 DEMONSTRATION PROBLEM, ORDER=4X4
   4    4    5    5    2    3
50.
.8211     1.094     1.687     2.626     3.23      1.000E-9
100.
.8211     1.073     1.571     2.301     2.76      1.000E-9
300.
.8211     1.047     1.43      1.907     2.191     1.000E-9
500.
.8211     1.039     1.388     1.793     2.026     1.000E-9
1000.
.8211     1.029     1.327     1.627     1.789     1.000E-9
2700.
3500.
4300.
4800.
5000.
   6
.9091     1.014     1.19      1.471     2.38      2.909
3000.     3300.     3700.     4100.     4700.     4900.
   6
.9047     1.0       1.158     1.396     2.112     2.516
3000.     3300.     3700.     4100.     4700.     4900.
   6
.8992     .9838     1.119     1.304     1.788     2.041
3000.     3300.     3700.     4100.     4700.     4900.
   6
.8974     .9789     1.107     1.276     1.694     1.903
3000.     3300.     3700.     4100.     4700.     4900.
   6
.8954     .9727     1.09      1.237     1.556     1.704
3000.     3300.     3700.     4100.     4700.     4900.
   0
   0
   0
   0
   0
   0
```

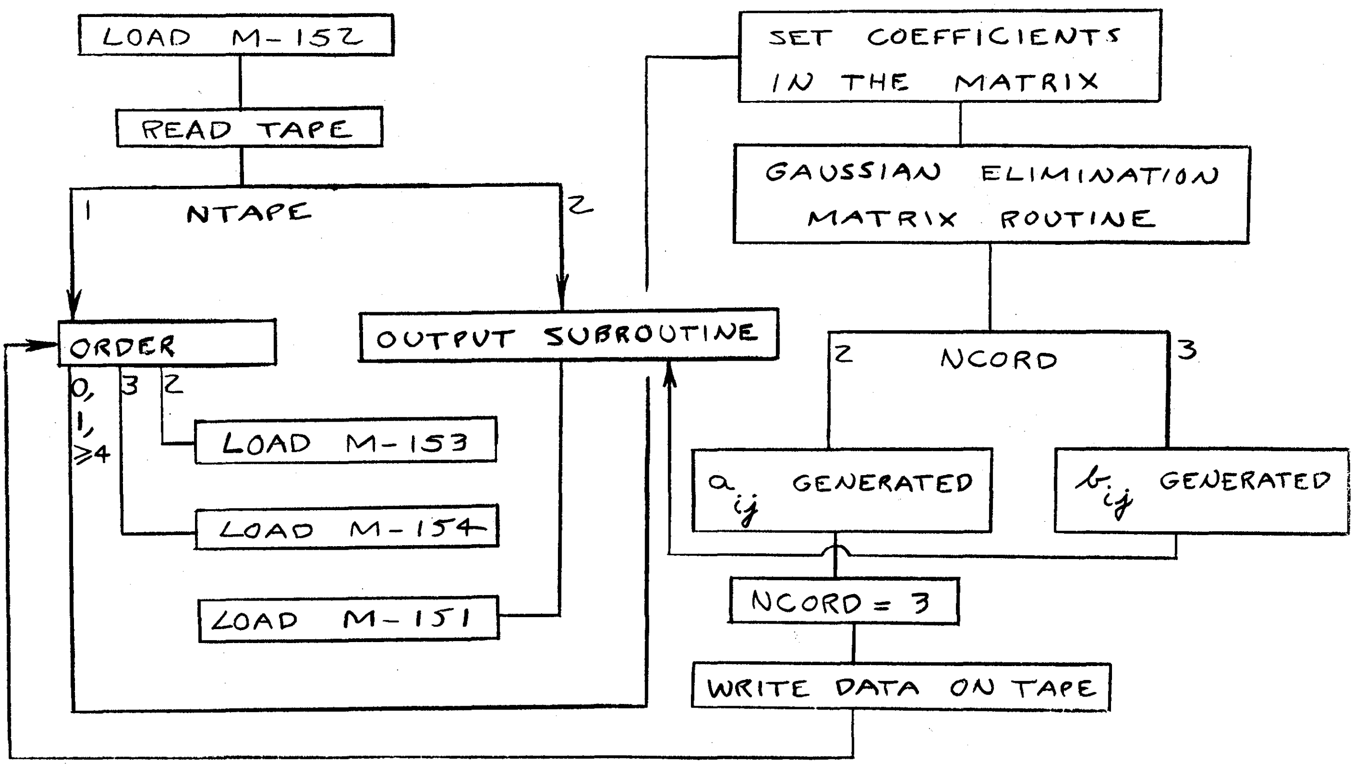
TABLE 6

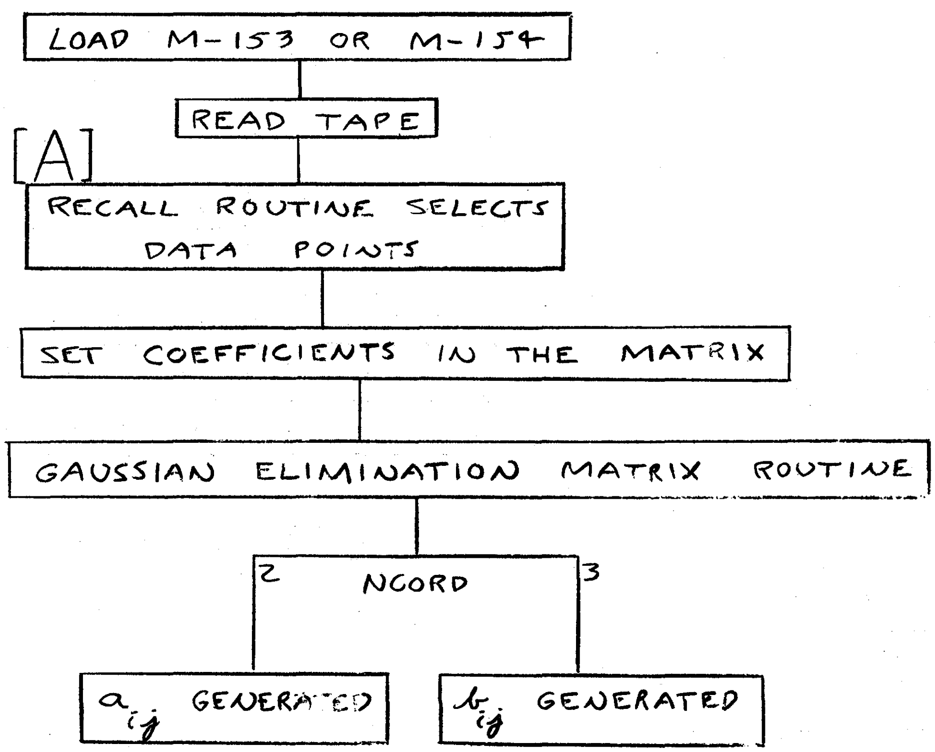
ERROR ANALYSIS

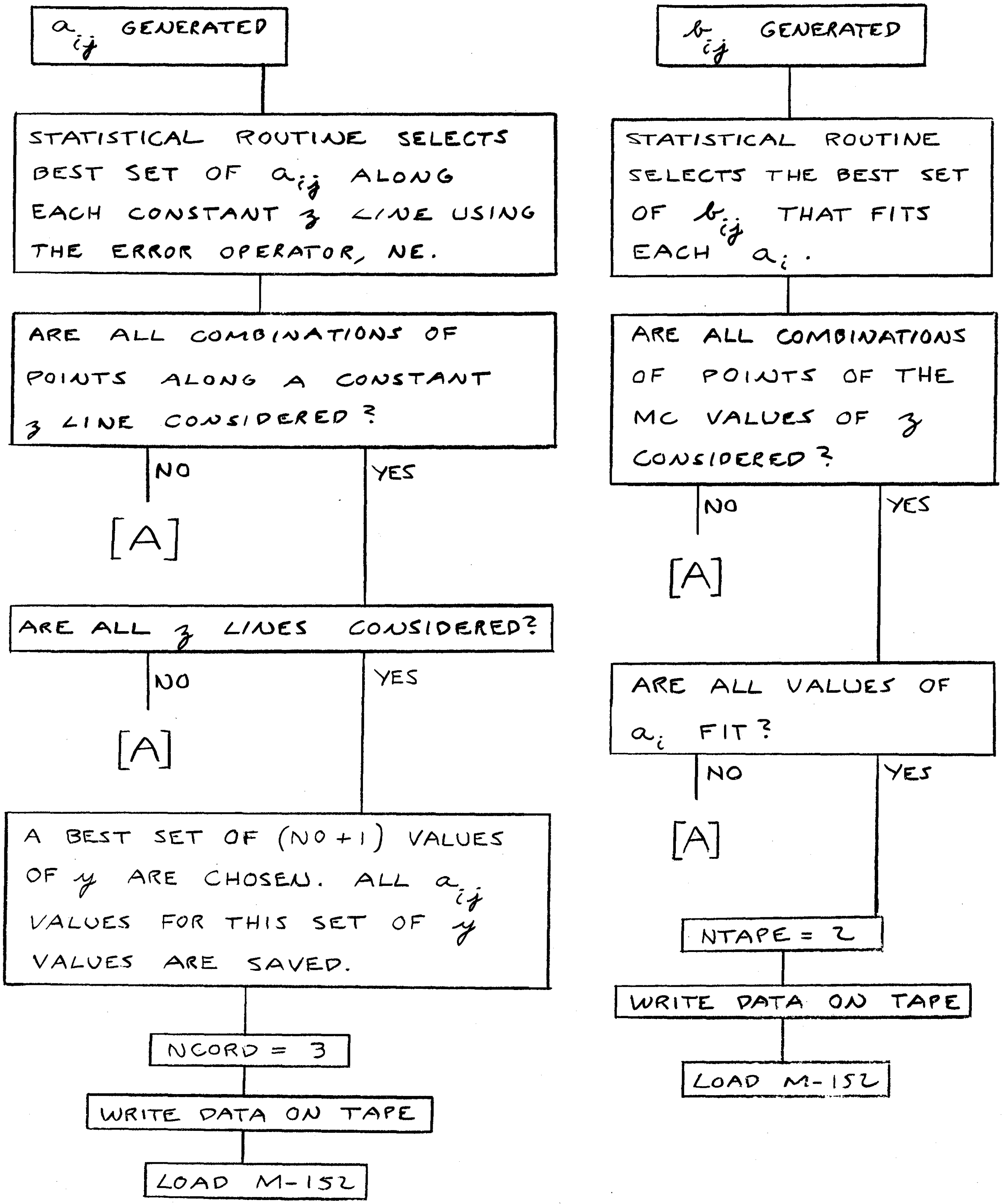
Order (n x m)	Average % Error	Maximum % Error
1 x 1	12.22	36.31
1 x 2	7.755	18.20
2 x 2	8.681	18.82
2 x 3	2.645	13.25
3 x 2	9.439	21.38
3 x 3	2.653	9.839
4 x 4	.04456	.3445
5 x 5	.07827	.7006
6 x 6	2.201	38.67
7 x 7	51.29	1211.

FLOW CHART




LOAD M-152
READ TAPE
NTAPE
1
2
ORDER
0, 1, ≥4
3
2
LOAD M-153
LOAD M-154
LOAD M-151
OUTPUT SUBROUTINE
SET COEFFICIENTS IN THE MATRIX
GAUSSIAN ELIMINATION MATRIX ROUTINE
NCORD
2
3
a_{ij} GENERATED
b_{ij} GENERATED
NCORD = 3
WRITE DATA ON TAPE


LOAD M-153 OR M-154
READ TAPE
[A]
RECALL ROUTINE SELECTS DATA POINTS
SET COEFFICIENTS IN THE MATRIX
GAUSSIAN ELIMINATION MATRIX ROUTINE
NCORD
2
3
a_{ij} GENERATED
b_{ij} GENERATED


a_{ij} GENERATED
STATISTICAL ROUTINE SELECTS BEST SET OF a_{ij} ALONG EACH CONSTANT z LINE USING THE ERROR OPERATOR, NE.
ARE ALL COMBINATIONS OF POINTS ALONG A CONSTANT z LINE CONSIDERED?
NO
YES
[A]
ARE ALL z LINES CONSIDERED?
NO
YES
[A]
A BEST SET OF (NO+1) VALUES OF y ARE CHOSEN. ALL a_{ij} VALUES FOR THIS SET OF y VALUES ARE SAVED.
NCORD = 3
WRITE DATA ON TAPE
LOAD M-152
b_{ij} GENERATED
STATISTICAL ROUTINE SELECTS THE BEST SET OF b_{ij} THAT FITS EACH a_i.
ARE ALL COMBINATIONS OF POINTS OF THE MC VALUES OF z CONSIDERED?
NO
YES
[A]
ARE ALL VALUES OF a_i FIT?
NO
YES
[A]
NTAPE = 2
WRITE DATA ON TAPE
LOAD M-152

MAXIMUM LIKELIHOOD RESOLUTION OF TWO MIXED NORMAL DISTRIBUTIONS

Reimut Wette

Biomathematics Unit, Division of Research,
The University of Texas M.D. Anderson Hospital and
Tumor Institute, Houston, Texas

Abstract:

Samples exhibiting particular deviations from an assumed normal distribution can, in certain situations, be interpreted on the assumption that the parent distribution consists of two normally distributed subpopulations, with different means and/or variances, and mixed in a finite proportion. The statistical problem is, then, to estimate the respective parameters. In the general case, where no simplifying assumptions (e.g. equality of variances) can be made, five parameters have to be estimated, viz. two means, two variances (or standard deviations), and the proportion of mixture. The author's estimation method of choice was the maximum likelihood scoring system. The derived estimator is of the iterative type, and modified insofar as the information matrix is estimated as the dyadic square of the score vector. Numerical execution of this iterative estimation procedure requires initial estimates of the five parameters; these are, at present, obtained manually from a normit graph of the data. A generalized distance from origin of the score vector (i.e. the quadratic form of the estimated variance-covariance matrix) is used as a criterion to exit from the iteration cycles.

The procedure was programmed for the IBM 1620 (with 1311) in FORTRAN II-D. The main program consists of two parts: 1.- Routines for control transfer set-up, standard form data input or transfer to either of two optional non-standard data input subprograms. 2.- The resolution procedure proper, routines for optional intermediate and other output, and transfer to two optional subprograms.

The program provides, at present, the following I/O (punched cards) and test options (under parameter card control):

Data input:
(1) Ungrouped or grouped data not exceeding 200 variates or classes.
(2) Grouping of ungrouped raw data; class interval and grouping range computed from initial estimates.
(3) Grouping of fixed-interval valued ungrouped data.

Output and test options:
(1) Output of grouped data frequency distribution.
(2) Output of expected frequency and cumulative frequency distribution together with observed distribution (grouped data only).
(3) Chi-squared test for normality against skewness and kurtosis (cumulant test), with programmed bypass of resolution peocedure when not significant.
(4) Intermediate iteration output of estimated information and variance-covariance matrix, parameter estimates and corrections, and convergence criterion.

Input options 2 and 3 and output options 2 and 3 are subprograms called as LOCALs on a 40k machine. A data trailer card controls exit to MONCAL or return linkage to part 1 of the main program. Maximum core storage used is 35,604 cores (Monitor I, modif. 2, all-in-core subroutines), program package (2 main/link, 4 subprograms) disk storage is 435 sectors. Interation cycle running time is about 2.7 seconds per point (variate or frequency class).

Convergence in this iterative estimation procedure depends on the goodness of the initial estimates, on sample size and structure. Experience gathered so far indicates that convergence, if attainable, is comparatively fast (less than 10 iterations). Convergence could not be attained in a few instances of small and ill-behaved samples, of size around 20 and less, from which reliable initial estimates could not be obtained.

The author gratefully acknowledges the extensive assistance of Mr. Lawrence E. Newton, Jr., in programming and testing different versions of the method in the Computer Science Laboratory of the M.D. Anderson Hospital and Tumor Institute.

<u>Note</u>: Methodological and programming details will be published elsewhere.

EVALUATION OF TWO METHODS OF FINDING SIGNIFICANT CONTRIBUTORS IN MULTIPLE REGRESSION ANALYSIS

M. J. Garber and Richard H. Hill[1/]

With the advent of computers in recent years tremendous strides have been made in the ability to reduce to manageable numbers vast accumulations of data. In evaluation of experimental results the major emphasis is now on punching and proofing the observed values with the assurance that a generalized computer program is probably on the shelf ready for use. In the majority of cases this is true, but thoughts and concepts have expanded along with the increase in ability to do arithmetic quickly and accurately. We are, of course, now finding problems even large scale computers cannot solve in a reasonable period of time.

One such problem is multiple regression involving many independent variables. A number of years ago the first of approximately 60 such problems was handed to the senior author. One phase of the experiment dealt with growth in a greenhouse of citrus seedlings in 102 nonfumigated old citrus soils. A total of 35 measurements was made on each. These included chemical, physical, and biological properties of the soil, plant composition, and relative growth. Locating statistically the best single contributor and the best set of 33 of the 34 independent variables is relatively a trivial operation. The best single contributor is the variable with the highest coefficient of determination ($r^2_{x_i \cdot y}$). The best set of 33 (or n-1) contributors is easily found by entering all n variables into the regression equation, and deleting the variable with the smallest

1/ University of California, Riverside, and Informatics, Inc.

absolute partial correlation ($|r_{yx_i \cdot x_j x_k ---}|$). Mr. Hill (then at the University of California, Los Angeles) wrote the MISLE program for the 709.

The best set of two, the best set of n-2, and all best sets between these extremes present a very different problem. The best set of two will not necessarily contain the best single contributor, and the best set of n-2 could contain the variable not included in the best set of n-1. For a statistically efficient analysis all best sets should be calculated. But this presents a formidable problem. For example, the final regression equation for the above problem contained 10 independent variables. Finding the best set of 10 among 34 variables can be quite a chore, even for a computer. The number of sets to be evaluated is 34!/(10!24!) or 131,128,140. Assuming that a computer can solve these at the rate of one per second, over 36,424 hours are required for finding the best set of 10.

A number of questions immediately come to mind:

(1) Is the best set of 10 significantly better than the one selected by this procedure?

(2) How does the best set of 10 compare statistically with the best set of 11, or any other best set?

(3) As the correlation coefficients are statistics, and not parameters, is the additional labor worth doing?

After some soul searching it was decided that initially the following three-phase procedure would be used:

(1) MISLE Program (presently UCRBL 0052, 1620 Library File No. 6.0.37)

All independent variables are entered into the regression equation, the inverse matrix of sums of squares and cross products being calculated in the process. All b_i^2/c_{ii} are calculated, where b_i is the partial regression coefficient of y on the ith independent variable, and c_{ii} is the major diagonal element for that variable in the inverse matrix. As Fisher (1) has shown b_i^2/c_{ii} is the variance (equivalent to a sum of squares with one degree of freedom) which would drop into the error sum of squares of y if the variable were deleted.

The variable with the smallest b^2/c is deleted, and the remaining n-1 independent variables are entered into the regression equation. This procedure is continued until only one variable remains.

An example of such analysis is illustrated in Table 1 in terms of the squares of partial correlation coefficients. Note that, in general, the simple correlation coefficients, the squares of which are shown in the second line of the table, do not reflect in any way the magnitudes of the squares of the partial correlation coefficients as variables are deleted.

The leftmost 10 variables were found to be statistically significant contributors (0.05 level of probability), and it should be noted that four of these showed nonsignificant simple correlation with the dependent variable. The magnitude of R^2 was 0.5387.

Variable 34 has the largest simple correlation. Yet, in combination with other variables its contribution is relatively small. On the other hand, Variable 31 which was not significantly directly associated with the dependent variable is relatively a large contributor in the presence of other variables.

(2) Stepwise Regression (UCRBL 0018, 1620 Library File No. 6.0.031, a modification of the BIMD 9 developed at UCLA by the Biostatistics group)

Here the procedure is to begin with the variable most highly correlated with y. From the remaining variables the one with the highest (absolute) partial correlation is selected for entry (assuming that it meets the F criterion), and the magnitudes of the two contributions are evaluated. Either both are retained or one is deleted (a parameter card entered with the data contains the minimum F value required to enter a variable into the regression equation and the minimum F value required for retention of a variable). This process of entering, evaluating, and deleting variables continues until the criteria can no longer be met.

For the problem being considered here nine variables were found to be statistically significant contributors (R^2 = .5205) when grouped. Eight of these variables were also selected by the MISLE program.

(3) MISLE Runs

(a) The eleven variables selected by either or both of the above programs were fed into the computer for another MISLE evaluation. Ten of the variables were found to be significant contributors, and the first variable of the eleven to be deleted was the one unique to the Program 18 run. There was thus no significant gain.

(b) It was then decided to make another run, this time including the eleven above and certain additional variables. Examination of the F values ($\geq$ 1) led to the choice of the two variables last deleted by Program 52 and the succeeding three which would have been entered by Program 18. Two of these were common to both runs.

Of the fourteen variables entered for the final run 10 were found to be significant. Seven of these were among the variables found significant by both programs, two common, but in the noncritical area, and the tenth being a significant contributor found by Program 18. The magnitude of R^2 was .5433.

The 10 variables statistically significant in this run were the ones concluded to be real contributors (paper by Martin, Harding, and Garber (2)).

The four curves are shown in Figure I. The curve of multiple R^2 calculated by Program 52 (labeled 52I) is almost smooth, running essentially horizontally until approximately six variables have been deleted. The drop off becomes quite steep as additional variables are deleted. It should be noted that the last remaining variable is not the one most highly correlated with y. The latter, Variable 34, was dropped after the 26th run.

The curve of multiple R^2 calculated by Program 18 begins by entering Variable 34 then rising in a somewhat oscillatory manner to essentially match the 52I curve after 9 variables have been entered.

The short spur labeled 52IIa is the graphic result of the MISLE run with 11 variables.

The short section of curve (52IIb) exhibits a sharper break than either of the first two.

The remaining two curves in the figure are from the 52I run. The curve of "Cumulated b^2/c" shows the addition to the residual sum of squares (all variables initially entered into the regression equation) as variables are deleted. The curve of "Residual Mean Squares" falls off from its initial value, then rises. Recalling that the residual mean square is the quotient of the residual sum of squares by the residual degrees of freedom it is seen in this example that in the initial phases of variable deletion the denominator is increasing relatively more rapidly than the numerator. Only after approximately half the number of variables has been deleted does the curve begin to rise.

A summary of the evaluation of 55 experiments is given in Table 2. For the left side of the table the F criterion was set at 1.00. For the right side of the table the 0.05 level of probability was the criterion.

In eight of the 55 experiments Program No. 18 was the better of the two programs. In four experiments Program No. 52 gave better results. In one experiment each of the programs provided information both significant and unique.

In one experiment the order of entering variables indicated the advisability of a third run. The outcome was the set of four variables selected by both programs.

For the remaining 41 experiments both programs gave the same results in the initial runs. The third run was unnecessary.

In summary, there is no clear cut indication of the superiority of one method over the other. For our purposes at Riverside we will continue to evaluate large problems by both the deletion and the stepwise methods.

LITERATURE CITED

1. Fisher, Sir R. A. (1948). Statistical Methods for Research Workers. 10th Edition. Hafner Publishing Company.

2. Martin, J. P., R. B. Harding, and M. J. Garber. (1961). Relation of Soil Properties and Plant Composition to Growth of Citrus Seedlings in 100 Nonfumigated and Fumigated Old Citrus Soils. Soil Science 91(5):317-323.

	1[1]	32	31	11	6	8	2	29	34	24	20	27	22	16	17	23	19	9	15	5	4	30	18	28	33	13	7	26	3	21	12	14	10	25
	12[2]	11	1	7	0	5	4	0	16	0	1	12	11	0	0	0	4	3	10	4	2	3	1	1	0	6	11	0	1	0	2	1	6	3
6184[3]	8[4]	13	10	0	3	2	1	4	5	5	7	4	1	1	1	3	2	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0
6184	8	13	10	0	3	3	1	4	5	6	7	4	1	1	1	3	2	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	
6184	8	13	10	3	3	8	1	4	5	6	7	4	1	1	1	3	2	2	1	0	1	1	0	0	1	1	0	0	0	0	0	0		
6184	8	13	10	4	5	8	1	5	5	6	8	4	1	1	1	3	2	2	1	0	1	1	0	0	1	1	0	0	0	0	0			
6182	8	14	10	7	6	8	3	5	5	6	7	4	1	1	1	3	2	2	1	0	1	1	0	0	1	1	0	0	0	0				
6179	9	14	10	7	5	8	3	5	5	6	8	4	2	1	1	3	3	2	1	0	1	1	0	1	1	1	0	0	0					
6175	9	14	10	7	5	8	3	5	5	6	8	4	2	3	2	3	2	2	1	0	1	1	0	1	1	1	0	0						
6171	9	14	10	7	7	8	3	5	5	6	8	4	2	3	2	3	2	2	1	0	1	1	0	1	1	1	0							
6164	9	15	10	7	7	10	3	6	5	6	9	5	2	4	2	3	2	2	2	1	1	1	1	1	0	0								
6146	10	15	12	9	7	9	5	6	6	6	8	4	1	4	2	3	2	2	2	1	1	1	1	0	0									
6131	11	15	13	8	7	9	5	6	6	6	8	5	1	3	2	3	2	2	2	1	1	1	0	0										
6120	11	15	13	8	7	9	5	6	7	6	8	5	1	4	2	2	2	2	2	1	1	1	0											
6104	11	15	13	9	8	9	5	5	8	6	8	5	1	3	2	2	2	2	2	1	1	1												
6067	10	15	13	8	8	10	5	6	8	6	7	5	2	4	3	2	2	2	2	1	1													
6035	10	15	13	8	10	9	5	6	8	7	7	5	1	4	3	2	2	1	1	0														
6018	9	15	13	8	10	9	5	6	8	6	8	5	2	4	3	2	2	1	1															
5967	10	17	12	7	12	8	7	7	6	5	8	4	3	3	2	2	1	0																
5948	10	17	11	7	12	8	9	6	6	6	8	5	3	4	2	2	2																	
5876	15	16	12	7	11	7	8	7	8	5	7	4	2	3	3	1																		
5817	14	15	11	8	11	8	8	8	9	5	6	3	2	3	2																			
5714	12	18	11	8	10	9	8	7	7	5	4	3	2	0																				
5693	12	18	13	7	10	9	7	7	7	5	4	3	1																					
5644	11	17	14	6	10	8	7	6	6	4	4	3																						
5489	14	17	14	10	10	8	8	7	6	5	2																							
5387	13	17	14	10	11	9	8	6	5	4																								
5171	12	17	17	11	11	9	7	6	3																									
5006	16	24	21	12	9	7	7	6																										
4682	12	25	19	9	9	6	5																											
4426	14	22	16	7	6	4																												
4194	28	19	17	6	5																													
3866	26	15	13	7																														
3434	26	16	12																															
2565	16	16																																
1154	12																																	

LEGEND

1. Variable number (Variables were deleted from right to left)
2. Simple $r^2_{yx_i}$ (Two decimal digits)
3. Multiple $R^2_{y.ijk---}$ (Four decimal digits)
4. Partial $r^2_{yi.jk---}$ (Two decimal digits)

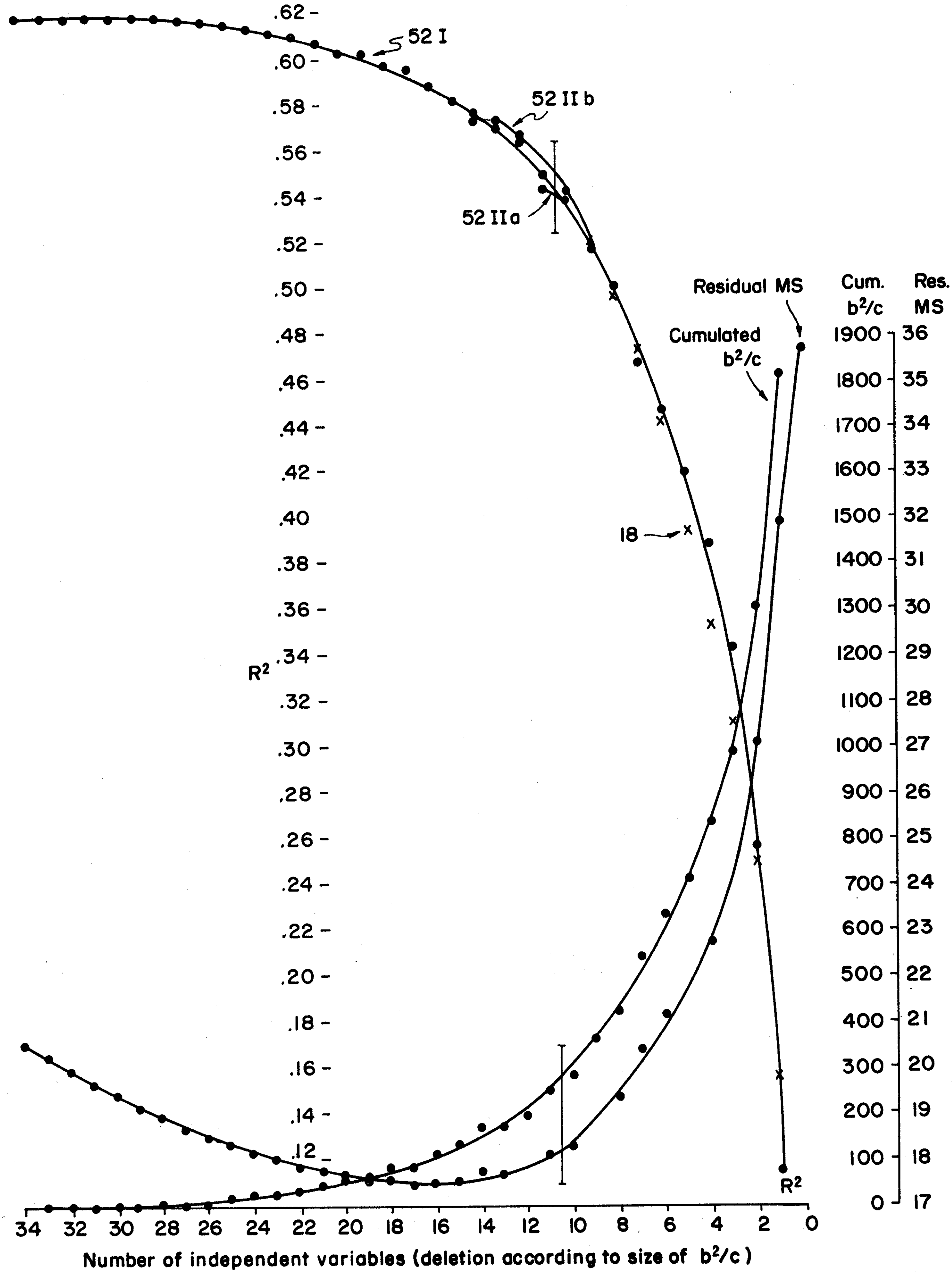
Table 1. Computer results (34 independent variables, 102 sets of observations).

Table 2. Summary of evaluation of 55 experiments.

No. of Indep. Var.	Degrees of Freedom	No. of Variables Selected by		No. of Var. in Common	No. of Variables Unique to		Final Selection			
		0052	0018		0052	0018	No. of Variables	No. Common	No. Unique to 0052	No. Unique to 0018
34	101	12	12	10	2	2	10	9		1
32	44	7	3		7	3	2			2
32	44	9	9	8	1	1	9	8		1
31	44	5	6	3	2	3	6	3		3
31	44	4	4	2	2	2	4	2		2
30	44	7	5	2	5	3	3	1		2
29	44	4	6	3	1	3	6	3		3
9	15	2	2	1	1	1	2	1		1
31	44	3	2		3	2	3		3	
31	44	6	5	4	2	1	6	4	2	
10	72	2	1		2	1	2		2	
6	72	2	2		2	2	2		2	
33	101	6	6	4	2	2	5	3	1	1
32	44	4	5	4		1	4	4		

Summary of 41 additional experiments:

No. of independent variables: 6 to 32
No. of degrees of freedom: 24 to 152
No. of significant contributors selected in common by 0052 and 0018: 0 to 5


52 I
52 II b
52 II a
18
Residual MS
Cumulated b²/c
Cum. b²/c
Res. MS
R²
Number of independent variables (deletion according to size of b²/c)

IBM

ELECTRONIC CIRCUIT ANALYSIS PROGRAM
FOR THE IBM 1620

APPLICATION PROGRAM BULLETIN

IBM 1620 ELECTRONIC CIRCUIT ANALYSIS

The Electronic Circuit Analysis system is a series of programs to help electronic engineers greatly extend their design capabilities through the use of the IBM 1620 Data Processing System. Previous computer experience is not required.

Electronic circuit performance requirements vary between industries and products. In some instances, the design objective is extremely high reliability; in others, it is a balance between cost, reliability, and consumer acceptability. The design effort includes determining the tolerance required of each individual circuit component in order to obtain an overall circuit performance within certain tolerances. Since the relationship between component tolerance and circuit tolerance is not a direct one, the analysis of the possible combinations of components is a time-consuming and tedious task to perform manually.

The IBM 1620 Electronic Circuit Analysis system provides the user with a high degree of confidence in circuit design, since many variations of circuit performance can be "tested". The system is user-oriented, providing means for easy communication and complete user control during the solution of the problem.

To use the system, the engineer starts with a schematic representing the equivalent circuit. He then numbers the nodes and branches of this network. A special input language is used to describe the network, its physical characteristics, and various user options such as type of output, type of analysis to be performed and circuit parameters to be varied. The input statements are then punched into cards in free form for entry into the 1620. The system interprets the input statements, determines the network topology, generates the network equations and performs the desired analysis. During the course of solution, the engineer can easily intervene at the console and modify system parameters, output desired, or many of the input language instructions by merely typing modified language statements on the console typewriter. Thus, the system allows either intervention for manual modification or completely automatic modification and solution. In this manner, the engineers can simulate their normal mode of design, which is experimental at the outset and iterative on a fixed design in the final stages.

The system consists of four sections: the monitor, the AC program, the DC program and the transient analysis program. The monitor reads and interprets the input cards, obtains the program requested (AC, DC or transient) from the auxiliary 1311 Disk Storage Drive, and transfers control to that program. Upon completion of the analysis, control is returned to the monitor, which reads any other input cards. Programs to be run may be stacked one behind the other, and solutions developed without manual intervention.

Some other features of the system are:

1. Networks of up to 21 nodes and 60 branches may be processed.

2. Nonlinear circuit elements are handled in the transient analysis through linear approximation. This is accomplished by automatic switching of circuit elements from one value to another in order to change the characteristic of the model being used.

3. The numerical stability of the transient solution is not affected by the size of the time step selected. Thus, the user can select large time steps in order to quickly obtain an indication of the transient behavior. In addition, the steady state solution can be obtained by selection of a single, sufficiently large time step.

The outputs available to the user are:

- Node voltages
- Branch currents
- Branch voltages
- Power dissipation
- Switching times

Copies of this and other IBM publications can be obtained through IBM Branch Offices. Address comments concerning the contents of this publication to IBM, Technical Publications Department, 112 East Post Road, White Plains, N. Y. 10601

- Sensitivity coefficients which indicate the relative effect of variations in the circuit elements on circuit output parameters
- Standard deviation of the node voltages
- "Worst case" analysis, which yields the maximum and minimum variations of node voltages when the worst possible combination of circuit element tolerances occurs
- Coefficients of the circuit equations generated by the system
- Node incident matrix indicating the topology of the circuit that the system generated.

A technique of solution is used in the AC/DC programs that substantially reduces the overall problem solution time when multiple solutions are required because of the varying of circuit parameters. The technique requires only a single nominal solution of the system equations. Subsequent solutions that would normally be required when circuit parameters are varied are obtained through a matrix update technique of the nominal solution matrix.

MACHINE CONFIGURATION

The minimum system required is a 40K IBM 1620 Data Processing System, with a 1311 Disk Storage Drive, a 1622 Card Read Punch, Automatic Divide and Indirect Addressing.

PRECISION

The standard floating point format of the IBM 1620 is used throughout, which carries eight significant digits. This precision results in a solution for the largest problem accommodated that is well within acceptable design accuracy.

PROGRAM LANGUAGE AND SYSTEM

The program is written in FORTRAN II D with Monitor I.

Printed in U. S. A. H20-4222

IBM
International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, New York 10601

ABSTRACT

The Electronic Circuit Analysis Program for the IBM 1620 is an engineering tool to aid in the design of reliable electronic circuits. The system provides facilities for complete DC, AC and transient analysis of an electrical network. The user can easily determine the effect of parameter variations on circuit performance that might arise due to effects of environment, aging and production tolerances. The manual calculations required to analyze circuit performance are tedious and time consuming for all but the most trivial circuits. The use of the computer relieves the designer of this restriction and greatly expands his capability.

FEATURES

-- provides user with a high degree of confidence in circuit design since many variations of circuit performance can be "tested" which could not have been physically duplicated in any reasonable amount of time.

-- System is user oriented, providing means for easy communication and complete user control over course of problem solution.

-- Simple input language used to describe circuit topology and component characteristics.

-- Automatic generation of network equations.

-- Handles network of up to 21 nodes and 60 branches.

-- Automatic modification of parameters or selective modification through console.

-- Transient analysis of non-linear circuits through automatic piece-wise linear approximation.

-- Worst-Case analysis of circuit design performed.

-- Obtains sensitivity of circuit performance to various parameters.

-- Numerical stability of transient solution not affected by size of time step selected.

-- Voltage and current dependent switching of element values is provided in transient program.

DESCRIPTION OF APPLICATION

The problem of reliable electrical circuit performance faces many designers in various industries. It is especially critical in the Aerospace Industries where extremely high reliability is desired. In other industries (e. g., electrical and communications) the problem is one of attaining a desired reliability that the consumer will tolerate.

The question to be answered in design of these circuits is what is the tolerance required of each individual circuit component in order to obtain an overall cricuit performance within certain tolerances. The relationship between individual component tolerance and overall circuit performance is not a direct one. Thus, it cannot be said that to obtain a maximum of ± 1% variation of a given circuit output parameter, all components must be at least 1% components (tolerances of ± 1% from nominal value). In fact, it may be that some components will require a ± .01% tolerance others can be as high as ± 10%.

The analysis of the possible combinations of components is a tedious, if not impossible task to perfrom manually, and it is not practical to build and test a large number of variations of any given circuit. Thus, the digital computer provides a means for solving this widespread problem.

Use

The Electronic Circuit Analysis Program is written in Fortran II D, works under Monitor-1 on the 1620 and resides on the 1311 disk. It is comprised of four programs: A main line program which interprets the input language and calls in the specific program requested, the AC program, the DC program and the Transient program. With the input language, the user describes his circuit, indicates the program he wants, the output he wants and which parameters he wants varied. The main line program determines the circuit topology, generates the circuit equations, calls in the proper program which then provides the desired solution.

MACHINE REQUIREMENTS

The minimum requirements for the Electronic Circuit Analysis Program is a 1620 with 40K storage, a 1311 disk drive, 1622 Reader Punch, Automatic Divide and Indirect Addressing.

ANALYSIS PROCEDURE

The procedure employed by the user to effectively utilize the IBM 1620 Electronic Circuit Analysis Program is outlined in Figure 1. The critical ingredient of the procedure is the generation of an appropriate equivalent circuit for the electron devices in the circuit under examination (tubes, diodes, transistors). These devices must be replaced by an equivalent circuit consisting of passive elements, current sources and voltage sources. The use of the Electronic Circuit Analysis Program allows the user to incorporate a level of sophistication into the equivalent circuits that renders a hand analysis of the result totally impractical. For a discussion of equivalent circuits for electron devices, the reader is referred to:

> Sheffler, H. S., Terry, F. R., "Description and Comparison of Computer Methods of Circuit Analysis", North American Aviation, Inc., Autonetics Division, Manual EM-6839.

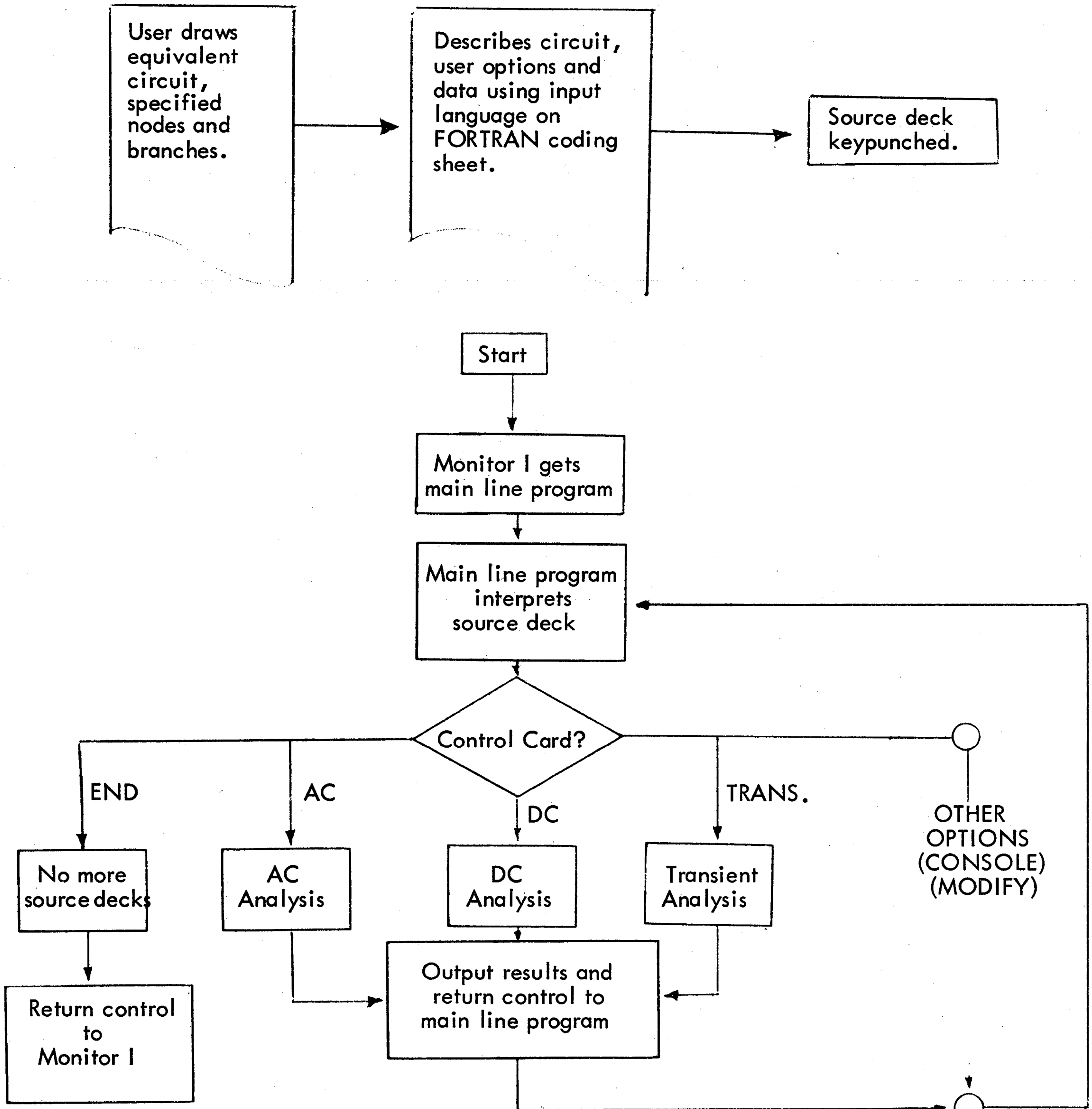


Figure 1. Analysis Procedure

INPUT LANGUAGE

SPECIFICATIONS

I. Introduction

The need for and usefulness of an integrated system of programs to do network analysis has already been shown. In order for the system to be most effective in carrying out its task, it must "speak" to the user in his own jargon. The circuit design engineer must be able to visualize a direct connection between the output which he obtains from the system and the input data presented to the system.

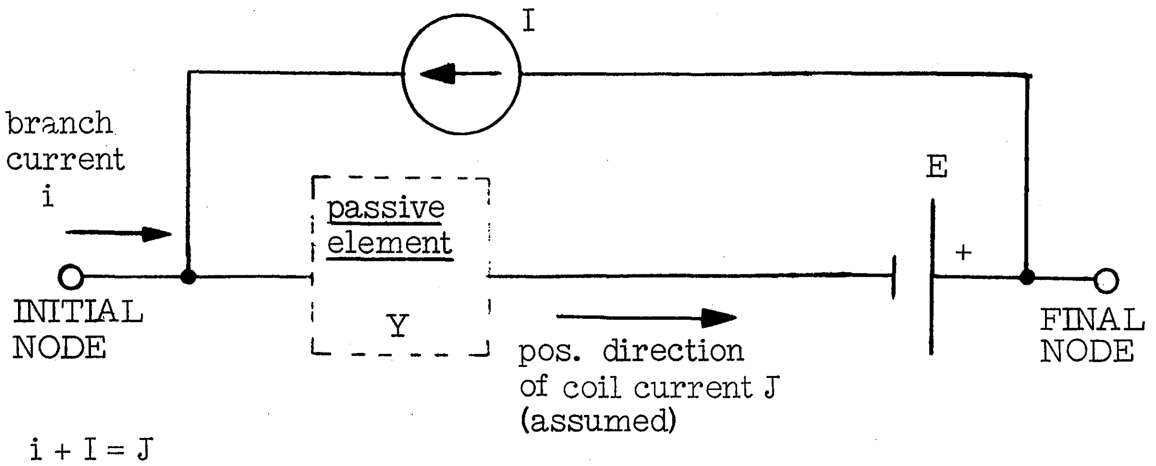
For this reason the SNAP language program was developed. The input data which must be prepared by the engineer is as close as possible to the format the engineer would normally use to carry out hand analyses of his circuit.

The input data could, perhaps, be handled in fixed-fields on data cards as FORTRAN II-D normally does. But the engineer's task seems overbearing when one considers that a 20 node-60 branch circuit could contain over 470 separate items to be input. This in itself could discourage the use of the system by all but the most earnest engineers. The present input language does not require the engineer to learn something entirely new to him, but instead allows him to use directly what he is already familiar with. The teaching of the language to prospective users is, to coin a phrase, a "SNAP."

II. Preliminaries

The approach a person would take in order to prepare data for the SNAP language program would be, as mentioned above, very close to the normal hand analysis procedure. The engineer would prepare from a schematic diagram an equivalent circuit using the basic building blocks (i. e., resistors, capacitors, inductors, voltage and current sources, transconductances, and mutual inductances) in place of special devices such as transistors, diodes, vacuum-tubes, etc.. The engineer must decide upon element values including tolerances, if needed, for D. C., alternate values in the case of switching in transient and phase angles and frequency for A. C. This particular phase -- the gathering of actual and representative values -- is usually the most difficult for any analysis.

Ingenious uses of the so-called "standard-branch" can in many cases lead to a simplification in the input data. The standard branch is defined as having each end terminate at a separate node and the branch between these two nodes can be one of the combinations shown in Figure 1.



Y can be any of the following:

1. Resistor or Conductor
   (expressed in ohms or mhos)

2a. Capacitor - (farads)

2b. Capacitor with leakage
   (farads and ohms, or mhos)

3. Inductor (henries)

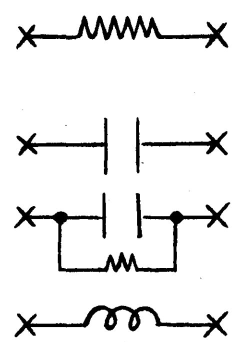


Figure 1

The assumed positive directions are as shown in the figure and will appear positive in sign on the data cards when oriented as shown. Note that the assumed positive direction of branch current goes from the INITIAL node to the FINAL node. The specification as to which node is initial and which is final is important in many cases and is part of the input information.

Once the engineer has represented his schematic in terms of the standard branch and has decided upon positive current directions, he then consecutively numbers the nodes and branches, as well as transconductances and switches.

The common node (usually ground) is referred to as node zero. This done, he is ready to write down the input data to be key-punched.

III. Specific Input Card Types

The engineer is now prepared to make use of the specific input card types with which he can properly describe his circuit to the SNAP system. There are three types of SNAP language statements: COMMANDS, CONTROL and DATA. A COMMAND specifies what analysis program is to be called and when to call it. A DATA statement specifies the circuit configuration and all of the circuit parameters. The class of statements called CONTROL cards instruct SNAP on what to do, but in a more specific sense than a command.

A. COMMANDS

1. DCNODE

   This command specifies that a DC analysis is to be made on the data which follows it.

2. ACNODE

   An AC analysis is to be made on the data which follows this card.

3. TRANSIENT

   A transient analysis is to be performed.

4. MODIFY

This command indicates that the data that follows modifies the previous analysis and that the previous analysis is to be either repeated or continued with the modified data incorporated. (Note: only 1 MODIFY command is allowed before any EXECUTE command).

5. EXECUTE

This command will cause execution of the commands preceding it which have not yet been executed.

6. CONSOLE

This command causes execution of the commands preceding it as with the EXECUTE command, but after execution is completed, will re-define the input device as the CONSOLE typewriter for manual entry.

The EXECUTE command will not re-define the input device. The SNAP program initially expects input via the card reader and will continue to do so with EXECUTE commands until a CONSOLE card is read. Thereafter, EXECUTE commands will leave the console typewriter as the input device.

7. READER

This command is similar to the CONSOLE command except it defines the card-reader as the input device after execution of the preceding unexecuted commands.

8. END

This command defines the end of a SNAP run. It causes the program to return control to the 1620 Monitor I Supervisor. It would not be used until a user were completely through with circuit analysis.

9. 1USER

This command as well as the two following can serve to call special user routines which can be compiled on the 1620 Monitor I, added to the SNAP package and called by use of these commands.

10. 2USER

11. 3USER

Examples of commands are shown in Figure 2 and are summarized in Table I.

1 Card Column	7
	DCNODE
B1	N=(0,1),R=1.R
D2	N-(53
	EXECUTE
	END

Figure 2.

Command	
DCNODE	Calls D.C. Analysis Program
ACNODE	Calls A.C. Analysis Program
TRANSIENT	Calls Transient Analysis
MODIFY	Denotes data to follow as modifying previous analysis and creates linkage to repeat previous analysis with modified data
EXECUTE	Signals end of data for analysis called for and starts execution
CONSOLE	Same as execute, except after execution returns control to typewriter
READER	Same as execute, except after execution returns control to card reader
END	Signals end of SNAP jobs
1 USER 2 USER 3 USER	These commands can call special user compiled programs.

Table 1.

B. DATA STATEMENTS

There are two general classes of data statements. The first classification specifies data from the equivalent circuit such as circuit topology, element values and initial conditions. The second group specifies data which applies to the circuit in a more general way. In this classification lie the frequency to be used in an A C circuit, the time step to be used for a Transient solution, as well as solution criteria such as is needed for Transient.

1. Equivalent Circuit Data Statements

These data statements contain information of two general types. The first type identifies the kind of data statement and its serial number and the second type identifies the actual data by prefacing the data with a data label.

a. Data Statement Card Identifiers

1. B - identifies a branch card.

Branch cards contain information about the individual branches within the circuit to be analyzed.

2. T - identifies a bransistor card.

A transistor card specifies a transconductance or current gain between two branches in the equivalent circuit.

3. M - identifes a mutual card.

Mutual cards specify the values of mutual inductance which exists between two inductive branches.

4. S - identifies a switch card.

Switch cards contain information about the switches in a circuit to be analyzed by the transient program.

b. Data Labels

Twelve different types of data could possibly be found on a data statement card. The following list associates the parameter with the particular identifier and indicates on which type of data statement card a given parameter may occur.

1.	Nodal data:	"N"	(B-cards only)
2.	Branch data:	"B"	(T-M-or S cards)
3.	Resistance data:	"R"	(B-cards only)
4.	Conductance data:	"G"	(B-cards only)
5.	Voltage data:	"E"	(B-cards only)
6.	Current data:	"I"	(B-cards only)
7.	Inductance data:	"L"	(B-or-M cards)
8.	Capacitance data:	"C"	(B-cards only)
9.	Initial Capacitor Voltage:	"EO"	(B-cards only)

10. Initial Inductor Current: "IO" (B-cards only)

11. Transconductance data: "GM" (T-cards only)

12. Current Gain data: "BETA" (T-cards only)

Examples of Data statements are shown in Figure 3.

Card columns 1, 7

1	7
	TRANSIENT
B1	N = (0,1), R = 1.E + 6
B2	N = (1,2), C = 5.3E - 6, EO = 150.
B3	N = (2,5), G = .00125
B4	N = (5,6), L = 5.7E - 3, IO = .0015
T1	B = (1,3), BETA = 65.
T2	B = (2,3), GM = 1.E -2
S1	B = 5, (7, 1, 9), OFF
	DELTA = .5E - 6
	EXECUTE
	END

Figure 3

2. General Data Statements

These data statements contain only two items: a data label and a value.

a. "FREQUENCY"

This data card specifies the frequency to be used in an AC analysis. Frequency is specifies in cycles.

Note: All of the following data cards are for use with the transient analysis program.

b. "DELTA"

Specifies the time increments to use in the transient analysis program.. This would normally be specified in units of seconds.

c. "MAJOR"

Specifies the number of time steps which should pass before an output should occur. The user may want to integrate with a small time step but may only want output at every tenth step. MAJOR is assumed to be 1 by the program unless otherwise specified.

d. "1 ERROR"

Specified the maximum allowable error in the substitution check. A check is made to see that all of the currents into each node sum to zero. Normally, due to round-off, they will not sum to zero. 1 ERROR allows the user to control the limit beyond which a message will be output indicating that the criterion has not been met. 1 ERROR is assumed to be 1 milliamp unless otherwise specified.

e. "2 ERROR"

When the transient analysis program is searching for the exact time at which a switch should be actuated, 2 ERROR provides a tolerance on how exact this time should be found. Unless specified by a 2 ERROR card, the program uses a value of .001 indicating that the switching time will be found to within. 1% of the time step.

f. "3 ERROR"

Accuracy in results can be increased by decreasing the time step under certain conditions of switching when inductors are involved in a circuit. 3 ERROR specifies the reduction to be made. Its value is assumed to be 1.0 by the program unless otherwise specified. This will inhibit this feature which is the normal procedure for most circuits.

g. "INITIAL TIME"

Specifies the starting time printout. DELTA will increment time from this base. This can be useful when re-analyzing a certain portion of a transient response when this portion occurs after time equals zero. The program assumes 0.0 unless otherwise specified.

h. "FINISH"

Specifies the time at which to terminate a transient analysis. This can allow the user to leave the computer unattended knowing that his job will not run endlessly. The value of FINSIH is initialized to $1.0 * 10^{49}$ unless otherwise specified by a FINISH card.

i. "SHORT"

j. "OPEN"

These two cards are available to the user to specify quantities used to calculate the Time = 0.0 voltages and currents. At time zero capacitors act as short circuits and inductors act as open circuits. These two values are substituted in place of the actual values when the transient analysis program is calculating these initial conditions. The Program assumes values of .01 for short and 1.0×10^7 for open unless cards are input to the contrary. If the user switches these values, that is, if he sets short to $1.0 \cdot 10^7$ and open to .01 he can obtain a steady state solution corresponding to the state of the circuit at $t = \infty$.

Examples of the use of Data Cards are shown in the sample problems in APPENDIX A.

C. The following Control Cards may be used with the SNAP Program (See TABLE III for a summary):

1. "PUNCH"

The SNAP Program is initialized to print the output data, but the entering of this control card will redefine the card punch as the output device.

2. "PRINT"

This card is used if a user wants both PRINT and PUNCH as output options. The punch control card will inhibit printing unless this card is used.

3. "SENSITIVITY"

The D.C. analysis program is prepared to calculate the sensitivity of each node voltage with respect to a small change in each circuit parameter. In addition, the actual partial derivatives are output. This control card initiates this output.

4. "WORST-CASE"

When given the proper tolerance data, the D.C. Program will make use of the partial derivatives mentioned above to calculate the worst-case maximum and mimimum node voltages. This card allows the user to get this information.

Note: the partial derivatives are necessary in order to calculate these worst-case conditions, therefore, the SENSITIVITY card is not required as input information. The program will automatically calculate the partial derivatives and sensitivities.

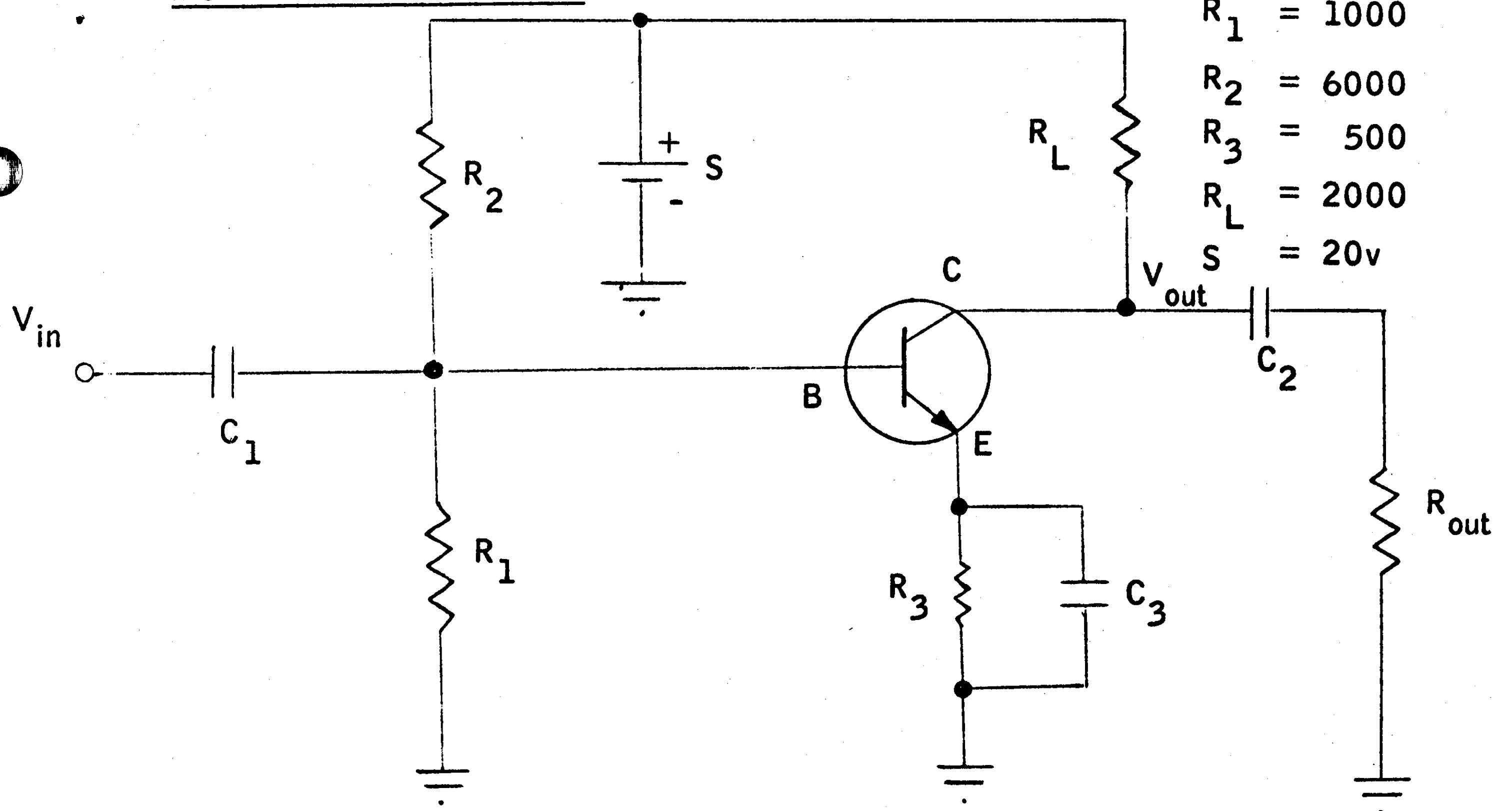
5. "STADEV"

The DC Analysis will calculate the standard deviation of each node voltage when the parameter tolerances are input and this card is entered. The sensitivities and partial derivatives are also required for these calculations so that the SENSITIVITY micro-data card is again not required.

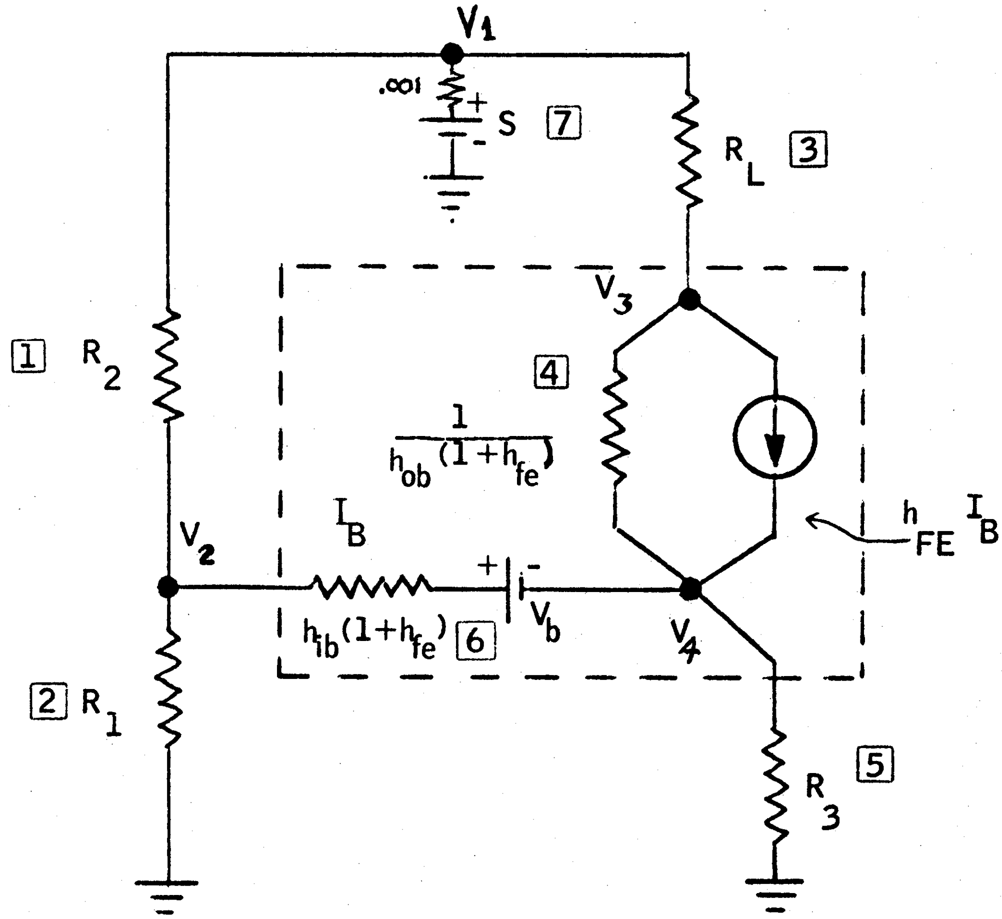
Control Card	Use	Analysis
PUNCH	Enables results to be punched. Inhibits Printing.	ALL
PRINT	Used only when punch inhibits print and printed output is also desired.	ALL
SENSITIVITY	Enables DC to perform sensitivity and partial derivative calculations.	D. C.
WORST-CASE	Enables Worst-Case calculations as well as sensitivity.	D. C.
STADEV	Enables Standard deviation calculations as well as sensitivity.	D. C.

Table III

DC ANALYSIS EXAMPLE



Single Stage Common Emitter Amplifier



Equivalent Circuit of Single Stage Common Emitter Amplifier

```
        DC ANALYSIS
C
C
C        COMMON EMITTER AMPLIFIER
C
C
B1      N=(1,2),R=6000.0
B2      N=(2,0),R=1000.0
B3      N=(1,3),R=2000.0
B4      N=(3,4),R=43000.0
B5      N=(4,0),R=500.0
B6      N=(2,4),R=487.5,E=-0.58
B7      N=(1,0),R=0.01,E=-20.0
T1      B=(6,4),BETA=42.0
        SENSITIVITY
        EXECUTE
EXECUTE PROGRAM.
DC CALLED.
DETMA
TOPOLOGICAL MATRIX DIMENSIONED    7  BRANCHES BY    4   NODES.


BRANCH   NODES
   1      1- 4          1         -1          0          0
   2      1- 4          0          1          0          0
   3      1- 4          1          0         -1          0
   4      1- 4          0          0          1         -1
   5      1- 4          0          0          0          1
   6      1- 4          0          1          0         -1
   7      1- 4          1          0          0          0
TRIPRO
SPARSE
NODAL CONDUCTANCE MATRIX DIMENSIONED    4   NODES
BY    4   NODES.


NODE ROW NODE COLS


    1       1 - 4 0.10000067E 03-0.16666667E-03-0.50000000E-03 0.
    2       1 - 4-0.16666667E-03 0.32179487E-02 0.             -0.20512821E-02
    3       1 - 4-0.50000000E-03 0.86153845E-01 0.52325581E-03-0.86177100E-01
    4       1 - 4 0.             -0.88205127E-01-0.23255814E-04 0.90228382E-01
EFCURR K= 1
LMVBM K= 1
EQUIVALENT CURRENT VECTOR


NODE NO.      CURRENT


    1       0.20000000E 04
    2       0.11897436E-02
    3       0.49969229E-01
    4      -0.51158973E-01
MISER
NODAL SOLUTION MATRIX


NODE ROW NODE COLS
```

201

```
    1       1 - 4 0.99999590E-02-0.14312440E-01 0.99640164E-02 0.91912452E-02
    2       1 - 4 0.13764431E-02 0.82326112E 03 0.87008461E 00 0.19547309E 02
    3       1 - 4 0.47343888E-02-0.31369275E 04 0.19925266E 04 0.18317456E 04
    4       1 - 4 0.13467984E-02 0.80399203E 03 0.13641357E 01 0.30664095E 02
PMVBM
NODE VOLTAGE SOLUTION VECTOR


NODE NO.      VOLTAGE


    1      0.19999928E 02
    2      0.27758129E 01
    3      0.11591434E 02
    4      0.21495623E 01
LMVBM K= 2
BRANCH VOLTAGE VECTOR


BRANCH NO.      VOLTAGE


     1       0.17224115E 02
     2       0.27758129E 01
     3       0.84084945E 01
     4       0.94418712E 01
     5       0.21495623E 01
     6       0.62625059E 00
     7       0.19999928E 02
COIL VOLTAGE VECTOR


BRANCH NO.      VOLTAGE


     1       0.17224115E 02
     2       0.27758129E 01
     3       0.84084945E 01
     4       0.94418712E 01
     5       0.21495623E 01
     6       0.46250604E-01
     7      -0.71763992E-04
EFCURR K= 2
BRANCH CURRENT VECTOR


BRANCH NO.      CURRENT


     1       0.28706858E-02
     2       0.27758129E-02
     3       0.42042472E-02
     4       0.42042457E-02
     5       0.42991246E-02
     6       0.94873034E-04
     7      -0.71763993E-02
COIL CURRENT VECTOR


BRANCH NO.      CURRENT


     1       0.28706858E-02
     2       0.27758129E-02
     3       0.42042472E-02
```

```
     4       0.42042457E-02
     5       0.42991246E-02
     6       0.94873034E-04
     7      -0.71763993E-02
POWER LOSS IN BRANCHES


BRANCH NO.        POWER


     1       0.49445022E-01
     2       0.77051372E-02
     3       0.35351389E-01
     4       0.39695946E-01
     5       0.92412360E-02
     6       0.43879351E-05
     7       0.51500706E-06
PARSHL     1    1


PARTIAL DERIVATIVES

NODE VOLTAGE WITHRESPECT TO RES. IN BRANCH    1


NODE NO.   PARTIALS          SENSITIVITY

   1    0.11632210E-07    0.69793257E-06
   2   -0.39388667E-03   -0.23633200E-01
   3    0.15008578E-02    0.90051467E-01
   4   -0.38466744E-03   -0.23080046E-01
PARSHL     2    1


PARTIAL DERIVATIVES

NODE VOLTAGE WITHRESPECT TO RES. IN BRANCH    2


NODE NO.   PARTIALS          SENSITIVITY

   1   -0.39728654E-07   -0.39728653E-06
   2    0.22852188E-02    0.22852188E-01
   3   -0.87075237E-02   -0.87075235E-01
   4    0.22317314E-02    0.22317314E-01
PARSHL     3    1


PARTIAL DERIVATIVES

NODE VOLTAGE WITHRESPECT TO RES. IN BRANCH    3


NODE NO.   PARTIALS          SENSITIVITY

   1    0.75555890E-10    0.15111178E-08
   2   -0.18261319E-05   -0.36522638E-04
   3   -0.41885271E-02   -0.83770541E-01
   4   -0.28647506E-05   -0.57295010E-04
PARSHL     4    1


PARTIAL DERIVATIVES

NODE VOLTAGE WITHRESPECT TO RES. IN BRANCH    4
```

NODE NO.	PARTIALS	SENSITIVITY
1	0.75556279E-10	0.32489199E-07
2	-0.18261312E-05	-0.78523641E-03
3	0.15720065E-04	0.67596278E-02
4	-0.28647495E-05	-0.12318423E-02

PARSHL 5 1

PARTIAL DERIVATIVES

NODE VOLTAGE WITHRESPECT TO RES. IN BRANCH 5

NODE NO.	PARTIALS	SENSITIVITY
1	0.79028616E-07	0.39514307E-06
2	0.16807263E-03	0.84036316E-03
3	0.15749805E-01	0.78749022E-01
4	0.26365753E-03	0.13182876E-02

PARSHL 6 1

PARTIAL DERIVATIVES

NODE VOLTAGE WITHRESPECT TO RES. IN BRANCH 6

NODE NO.	PARTIALS	SENSITIVITY
1	0.17422980E-08	0.84937024E-08
2	0.37502359E-05	0.18282400E-04
3	0.34721164E-03	0.16926567E-02
4	-0.88989997E-04	-0.43382623E-03

PARSHL 7 1

PARTIAL DERIVATIVES

NODE VOLTAGE WITHRESPECT TO RES. IN BRANCH 7

NODE NO.	PARTIALS	SENSITIVITY
1	-0.71763699E-02	-0.71763697E-06
2	-0.98779051E-03	-0.98779049E-07
3	-0.33975864E-02	-0.33975864E-06
4	-0.96651630E-03	-0.96651628E-07

PARSHL 1 2

PARTIAL DERIVATIVES

NODE VOLTAGE WITHRESPECT TO GAIN BRANCH 6 TO BRANCH 4

NODE NO.	PARTIALS	SENSITIVITY
1	-0.35741134E-04	-0.30792361E-07
2	0.86383292E 00	0.74422528E-03
3	-0.74362179E 01	-0.64065876E-02
4	0.13551408E 01	0.11675059E-02

PARSHL 1 3

PARSHL 2 3

PARSHL 3 3

PARSHL 4 3

PARSHL 5 3

PARSHL 6 3

PARTIAL DERIVATIVES

NODE VOLTAGE WITHRESPECT TO VOLTAGE SOURCE IN BRANCH 6

NODE NO.	PARTIALS	SENSITIVITY
1	0.18364523E-04	-0.15821743E-05
2	0.39528981E-01	-0.34055737E-02
3	0.36597508E 01	-0.31530160E-00
4	-0.93799041E 00	0.80811480E-01

PARSHL 7 3

PARTIAL DERIVATIVES

NODE VOLTAGE WITHRESPECT TO VOLTAGE SOURCE IN BRANCH 7

NODE NO.	PARTIALS	SENSITIVITY
1	0.99999591E 00	-0.99999592E 02
2	0.13764431E-00	-0.13764431E 02
3	0.47343888E-00	-0.47343888E 02
4	0.13467984E-00	-0.13467984E 02

PARSHL 1 4

PARSHL 2 4

PARSHL 3 4

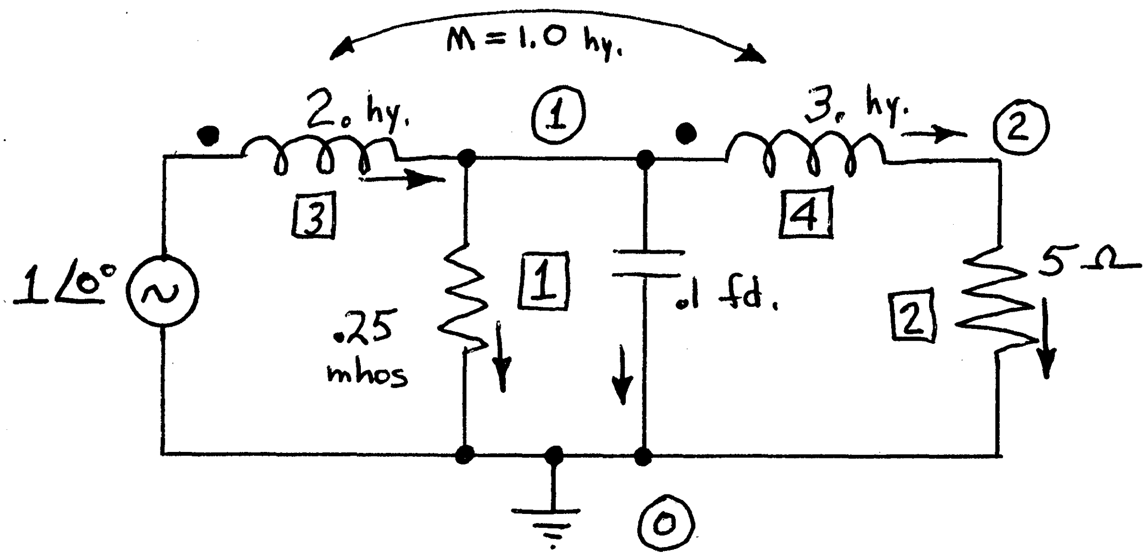
PARSHL 4 4

PARSHL 5 4

PARSHL 6 4

PARSHL 7 4

A. C. EXAMPLE



1		7
		ACNODE
B1		N (1,0), G = .25, C = .1
B2		N (2,0), R = 5.0
B3		N (0,1), L = 2.0, E = 1/0
B4		N (1,2), L = 3.0
M1		B (3,4), M = 1.0
		FREQUENCY = 1.59
		EXECUTE

```
        ACNODE
B1      N=(1,0),G=.25,C=.1
B2      N=(2,0),R=5.
B3      N=(0,1),L=2.,E=1./0.
B4      N=(1,2),L=3.
M1      B=(3,4),M=1.
        FREQUENCY= 1.59
        EXECUTE
 EXECUTE PROGRAM.
 AC CALLED.



 NODE VOLTAGE SOLUTION VECTOR

        MAGNITUDE         PHASE
        8.146E-03      -1.740E+02
        1.044E-01       9.049E+01



 BRANCH VOLTAGE VECTOR

        MAGNITUDE         PHASE
        8.146E-03      -1.740E+02
        1.044E-01       9.049E+01
        8.146E-03       5.936E-00
        1.055E-01      -9.390E+01
```

TRANSIENT ANALYSIS EXAMPLE

6. LCR Voltage Regulator

The objective of the circuit in this example is to regulate the voltage V3 at 12.0 volts. This is done by utilizing the cut-off and saturation characteristics of a transistor and a diode. If V3, is less than 12.0v, a comparator senses this and turns the transistor on. This cuts off the diode and builds up the voltage on the capacitor. When $V3 \leq 12.0$ volts the transistor is turned off. The inductor tends to resist a change in current. The return path for the current is through the diode which goes from the blocked condition to full conduction. The voltage drops on the capacitor, which is sensed by the comparator. There is a hysteresis voltage drop associated with the comparator of 0.3 volts. Hence, when $V3 - 11.7 \leq 0$, the comparator turns the transistor back on and the process is repeated.

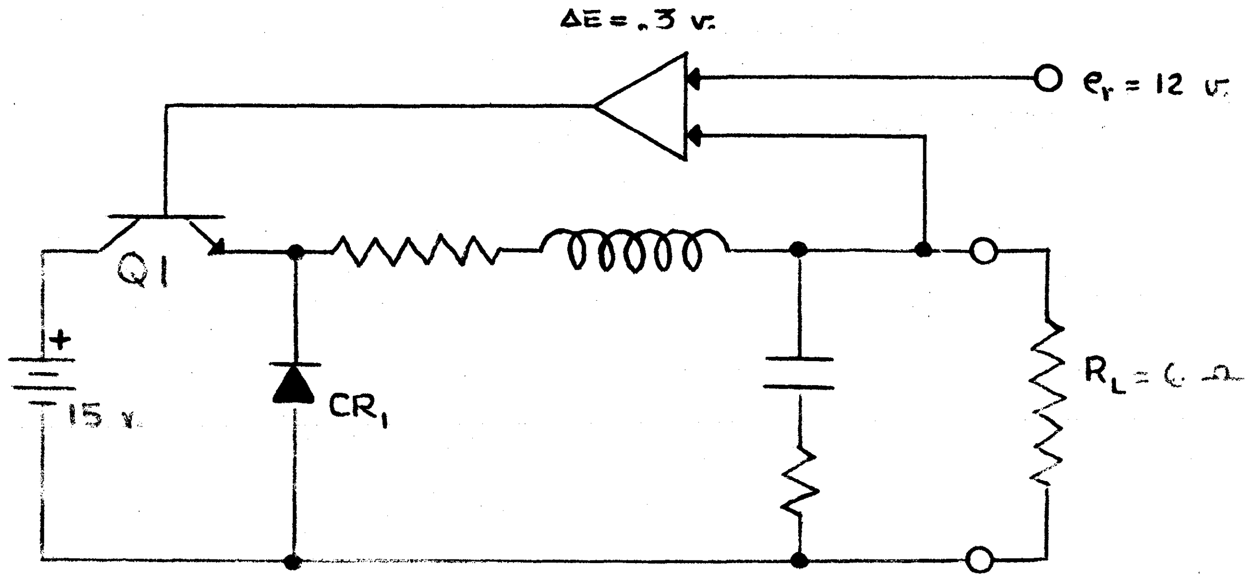


Fig. 6a LCR Voltage Regulator - Schematic Diagram

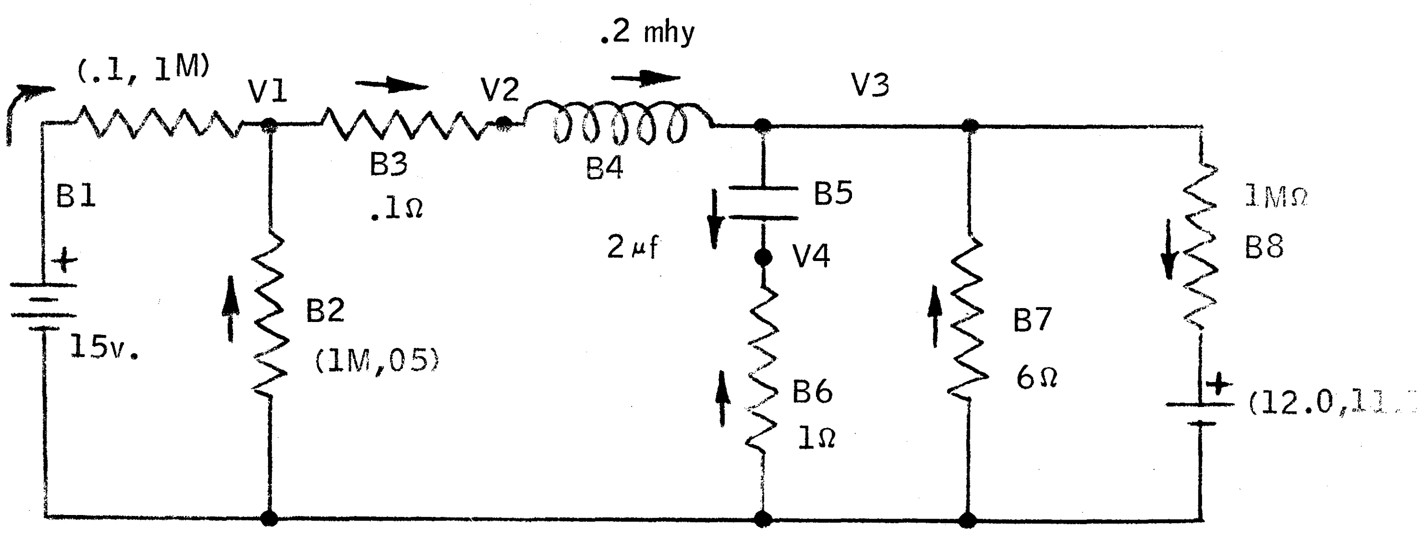


Fig. 6b LCR Voltage Regulator - Equivalent Circuit

```
          TRANSIENT
C
C
C         LCR VCLTAGE REGULATOR
C
C
B1        N=(0,1),R=(.1,1.E6),E=(15.,15.)
B2        N=(0,1),R=(1.E6,.5)
B3        N=(1,2),R=.1
B4        N=(2,3),L=.2E-3
B5        N=(3,4),C=.2E-5
B6        N=(0,4),R=1.
B7        N=(0,3),R=6.
B8        N=(3,0),R=(1.E6,1.E6),E=(-12.,-11.7)
S1        B=8,(1,2,8),OFF
          DELTA=1.0E-6
          MAJOR=20
          1ERROR=1.0E-2
          FINISH=200.0E-6
          EXECUTE
EXECUTE PROGRAM.
TR CALLED.


    T =   0.

V   1 =   0.1500E 02
V   2 =   0.1500E 02
V   3 =   0.1167E-04
V   4 =   0.1155E-04


I   1 =   0.1788E-04
I   2 =  -0.1500E-04
I   3 =   0.
I   4 =   0.1500E-05
I   5 =   0.1155E-04
I   6 =  -0.1155E-04
I   7 =  -0.1945E-05
I   8 =  -0.1200E-04


    T =   0.2000E-04

V   1 =   0.1487E 02
V   2 =   0.1474E 02
V   3 =   0.4436E 01
V   4 =   0.5578E 00


I   1 =   0.1297E 01
I   2 =  -0.1487E-04
I   3 =   0.1297E 01
I   4 =   0.1297E 01
I   5 =   0.5578E 00
I   6 =  -0.5578E 00
I   7 =  -0.7394E 00
I   8 =  -0.7564E-05
```

T = 0.4000E-04

V 1 = 0.1480E 02
V 2 = 0.1459E 02
V 3 = 0.9586E 01
V 4 = 0.4516E-00

I 1 = 0.2049E 01
I 2 = -0.1480E-04
I 3 = 0.2049E 01
I 4 = 0.2049E 01
I 5 = 0.4516E-00
I 6 = -0.4516E-00
I 7 = -0.1598E 01
I 8 = -0.2414E-05

T = 0.5378E-04

V 1 = 0.1477E 02
V 2 = 0.1454E 02
V 3 = 0.1200E 02
V 4 = 0.3031E-00

I 1 = 0.2303E 01
I 2 = -0.1477E-04
I 3 = 0.2303E 01
I 4 = 0.2303E 01
I 5 = 0.3031E-00
I 6 = -0.3031E-00
I 7 = -0.2000E 01
I 8 = -0.4399E-10

SWITCH 1 IS ON

T = 0.5378E-04

V 1 = -0.1152E 01
V 2 = -0.1382E 01
V 3 = 0.1200E 02
V 4 = 0.3026E-00

I 1 = 0.1615E-04
I 2 = 0.2303E 01
I 3 = 0.2303E 01
I 4 = 0.2303E 01
I 5 = 0.3026E-00
I 6 = -0.3026E-00
I 7 = -0.2000E 01
I 8 = 0.3025E-06

T = 0.6000E-04

V 1 = -0.9455E 00
V 2 = -0.1135E 01

```
V  3 =  0.1183E 02
V  4 = -0.8035E-01

I  1 =  0.1595E-04
I  2 =  0.1891E 01
I  3 =  0.1891E 01
I  4 =  0.1891E 01
I  5 = -0.8035E-01
I  6 =  0.8035E-01
I  7 = -0.1971E 01
I  8 =  0.1279E-06


   T =  0.6116E-04

V  1 = -0.9080E 00
V  2 = -0.1090E 01
V  3 =  0.1170E 02
V  4 = -0.1340E-00

I  1 =  0.1591E-04
I  2 =  0.1816E 01
I  3 =  0.1816E 01
I  4 =  0.1816E 01
I  5 = -0.1340E-00
I  6 =  0.1340E-00
I  7 = -0.1950E 01
I  8 =  0.3791E-10

SWITCH    1    IS UFF


   T =  0.6116E-04

V  1 =  0.1482E 02
V  2 =  0.1464E 02
V  3 =  0.1170E 02
V  4 = -0.1338E-00

I  1 =  0.1816E 01
I  2 = -0.1482E-04
I  3 =  0.1816E 01
I  4 =  0.1816E 01
I  5 = -0.1338E-00
I  6 =  0.1338E-00
I  7 = -0.1950E 01
I  8 = -0.3011E-06


   T =  0.7991E-04

V  1 =  0.1479E 02
V  2 =  0.1458E 02
V  3 =  0.1200E 02
V  4 =  0.8745E-01
```



```
I  1 =  0.2087E 01
I  2 = -0.1479E-04
```

```
I   3 =   0.2087E 01
I   4 =   0.2087E 01
I   5 =   0.8745E-01
I   6 =  -0.8745E-01
I   7 =  -0.2000E 01
I   8 =   0.1895E-10

SWITCH     1    IS ON


    T =   0.7991E-04

V   1 =  -0.1044E 01
V   2 =  -0.1252E 01
V   3 =   0.1200E 02
V   4 =   0.8730E-01

I   1 =   0.1604E-04
I   2 =   0.2087E 01
I   3 =   0.2087E 01
I   4 =   0.2087E 01
I   5 =   0.8730E-01
I   6 =  -0.8730E-01
I   7 =  -0.2000E 01
I   8 =   0.3007E-06


    T =   0.8000E-04

V   1 =  -0.1041E 01
V   2 =  -0.1249E 01
V   3 =   0.1200E 02
V   4 =   0.8142E-01

I   1 =   0.1604E-04
I   2 =   0.2081E 01
I   3 =   0.2081E 01
I   4 =   0.2081E 01
I   5 =   0.8142E-01
I   6 =  -0.8142E-01
I   7 =  -0.2000E 01
I   8 =   0.2987E-06


    T =   0.8386E-04

V   1 =  -0.9145E 00
V   2 =  -0.1097E 01
V   3 =   0.1170E 02
V   4 =  -0.1209E-00

I   1 =   0.1591E-04
I   2 =   0.1829E 01
I   3 =   0.1829E 01
I   4 =   0.1829E 01
I   5 =  -0.1209E-00
I   6 =   0.1209E-00
I   7 =  -0.1950E 01
I   8 =  -0.3254E-10
```

SWITCH 1 IS OFF

```
    T =  0.8386E-04

V   1 =  0.1482E 02
V   2 =  0.1463E 02
V   3 =  0.1170E 02
V   4 = -0.1207E-00

I   1 =  0.1829E 01
I   2 = -0.1482E-04
I   3 =  0.1829E 01
I   4 =  0.1829E 01
I   5 = -0.1207E-00
I   6 =  0.1207E-00
I   7 = -0.1950E 01
I   8 = -0.3011E-06

    T =  0.1000E-03

V   1 =  0.1479E 02
V   2 =  0.1459E 02
V   3 =  0.1192E 02
V   4 =  0.7623E-01

I   1 =  0.2063E 01
I   2 = -0.1479E-04
I   3 =  0.2063E 01
I   4 =  0.2063E 01
I   5 =  0.7623E-01
I   6 = -0.7623E-01
I   7 = -0.1987E 01
I   8 = -0.7665E-07

    T =  0.1016E-03

V   1 =  0.1479E 02
V   2 =  0.1458E 02
V   3 =  0.1200E 02
V   4 =  0.8494E-01

I   1 =  0.2085E 01
I   2 = -0.1479E-04
I   3 =  0.2085E 01
I   4 =  0.2085E 01
I   5 =  0.8494E-01
I   6 = -0.8494E-01
I   7 = -0.2000E 01
I   8 = -0.2325E-10
```

SWITCH 1 IS ON

T = 0.1016E-03

V 1 = -0.1042E 01
V 2 = -0.1251E 01
V 3 = 0.1200E 02
V 4 = 0.8479E-01

I 1 = 0.1604E-04
I 2 = 0.2085E 01
I 3 = 0.2085E 01
I 4 = 0.2085E 01
I 5 = 0.8479E-01
I 6 = -0.8479E-01
I 7 = -0.2000E 01
I 8 = 0.3007E-06

T = 0.1056E-03

V 1 = -0.9136E 00
V 2 = -0.1096E 01
V 3 = 0.1170E 02
V 4 = -0.1228E-00

I 1 = 0.1591E-04
I 2 = 0.1827E 01
I 3 = 0.1827E 01
I 4 = 0.1827E 01
I 5 = -0.1228E-00
I 6 = 0.1228E-00
I 7 = -0.1950E 01
I 8 = 0.5162E-10

SWITCH 1 IS OFF

T = 0.1056E-03

V 1 = 0.1482E 02
V 2 = 0.1463E 02
V 3 = 0.1170E 02
V 4 = -0.1226E-00

I 1 = 0.1827E 01
I 2 = -0.1482E-04
I 3 = 0.1827E 01
I 4 = 0.1827E 01
I 5 = -0.1226E-00
I 6 = 0.1226E-00
I 7 = -0.1950E 01
I 8 = -0.3010E-06

T = 0.1200E-03

V 1 = 0.1480E 02
V 2 = 0.1459E 02
V 3 = 0.1184E 02
V 4 = 0.6492E-01

```
I  1 =  0.2039E 01
I  2 = -0.1480E-04
I  3 =  0.2039E 01
I  4 =  0.2039E 01
I  5 =  0.6492E-01
I  6 = -0.6492E-01
I  7 = -0.1974E 01
I  8 = -0.1576E-06


   T =  0.1235E-03

V  1 =  0.1479E 02
V  2 =  0.1458E 02
V  3 =  0.1200E 02
V  4 =  0.8534E-01

I  1 =  0.2085E 01
I  2 = -0.1479E-04
I  3 =  0.2085E 01
I  4 =  0.2085E 01
I  5 =  0.8534E-01
I  6 = -0.8534E-01
I  7 = -0.2000E 01
I  8 = -0.2384E-12

SWITCH     1    IS ON


   T =  0.1235E-03

V  1 = -0.1043E 01
V  2 = -0.1251E 01
V  3 =  0.1200E 02
V  4 =  0.8518E-01

I  1 =  0.1604E-04
I  2 =  0.2085E 01
I  3 =  0.2085E 01
I  4 =  0.2085E 01
I  5 =  0.8519E-01
I  6 = -0.8518E-01
I  7 = -0.2000E 01
I  8 =  0.3007E-06


   T =  0.1275E-03

V  1 = -0.9136E 00
V  2 = -0.1096E 01
V  3 =  0.1170E 02
V  4 = -0.1228E-00

I  1 =  0.1591E-04
I  2 =  0.1827E 01
I  3 =  0.1827E 01
I  4 =  0.1827E 01
```

I 5 = -0.1228E-00
I 6 = 0.1228E-00
I 7 = -0.1950E 01
I 8 = 0.5484E-10

SWITCH 1 IS OFF

T = 0.1275E-03

V 1 = 0.1482E 02
V 2 = 0.1463E 02
V 3 = 0.1170E 02
V 4 = -0.1227E-00

I 1 = 0.1827E 01
I 2 = -0.1482E-04
I 3 = 0.1827E 01
I 4 = 0.1827E 01
I 5 = -0.1227E-00
I 6 = 0.1227E-00
I 7 = -0.1950E 01
I 8 = -0.3010E-06

T = 0.1400E-03

V 1 = 0.1480E 02
V 2 = 0.1460E 02
V 3 = 0.1177E 02
V 4 = 0.5081E-01

I 1 = 0.2013E 01
I 2 = -0.1480E-04
I 3 = 0.2013E 01
I 4 = 0.2013E 01
I 5 = 0.5081E-01
I 6 = -0.5081E-01
I 7 = -0.1962E 01
I 8 = -0.2296E-06

T = 0.1454E-03

V 1 = 0.1479E 02
V 2 = 0.1458E 02
V 3 = 0.1200E 02
V 4 = 0.8534E-01

I 1 = 0.2085E 01
I 2 = -0.1479E-04
I 3 = 0.2085E 01
I 4 = 0.2085E 01
I 5 = 0.8534E-01
I 6 = -0.8534E-01
I 7 = -0.2000E 01
I 8 = 0.1550E-10

SWITCH 1 IS ON

```
   T =  0.1454E-03

V  1 = -0.1043E 01
V  2 = -0.1251E 01
V  3 =  0.1200E 02
V  4 =  0.8520E-01

I  1 =  0.1604E-04
I  2 =  0.2085E 01
I  3 =  0.2085E 01
I  4 =  0.2085E 01
I  5 =  0.8519E-01
I  6 = -0.8520E-01
I  7 = -0.2000E 01
I  8 =  0.3007E-06


   T =  0.1493E-03

V  1 = -0.9137E 00
V  2 = -0.1096E 01
V  3 =  0.1170E 02
V  4 = -0.1226E-00

I  1 =  0.1591E-04
I  2 =  0.1827E 01
I  3 =  0.1827E 01
I  4 =  0.1827E 01
I  5 = -0.1226E-00
I  6 =  0.1226E-00
I  7 = -0.1950E 01
I  8 =  0.1252E-10
```

SWITCH 1 IS OFF

```
   T =  0.1493E-03

V  1 =  0.1482E 02
V  2 =  0.1463E 02
V  3 =  0.1170E 02
V  4 = -0.1224E-00

I  1 =  0.1827E 01
I  2 = -0.1482E-04
I  3 =  0.1827E 01
I  4 =  0.1827E 01
I  5 = -0.1225E-00
I  6 =  0.1224E-00
I  7 = -0.1950E 01
I  8 = -0.3011E-06
```



```
   T =  0.1600E-03
```

```
V  1 =  0.1480E 02
V  2 =  0.1460E 02
V  3 =  0.1171E 02
V  4 =  0.3425E-01

I  1 =  0.1986E 01
I  2 = -0.1480E-04
I  3 =  0.1986E 01
I  4 =  0.1986E 01
I  5 =  0.3425E-01
I  6 = -0.3425E-01
I  7 = -0.1952E 01
I  8 = -0.2897E-06


   T =  0.1672E-03

V  1 =  0.1479E 02
V  2 =  0.1458E 02
V  3 =  0.1200E 02
V  4 =  0.8528E-01

I  1 =  0.2085E 01
I  2 = -0.1479E-04
I  3 =  0.2085E 01
I  4 =  0.2085E 01
I  5 =  0.8528E-01
I  6 = -0.8528E-01
I  7 = -0.2000E 01
I  8 =  0.1752E-10

SWITCH    1   IS ON


   T =  0.1672E-03

V  1 = -0.1043E 01
V  2 = -0.1251E 01
V  3 =  0.1200E 02
V  4 =  0.8513E-01

I  1 =  0.1604E-04
I  2 =  0.2085E 01
I  3 =  0.2085E 01
I  4 =  0.2085E 01
I  5 =  0.8512E-01
I  6 = -0.8513E-01
I  7 = -0.2000E 01
I  8 =  0.3007E-06


   T =  0.1712E-03

V  1 = -0.9140E 00
V  2 = -0.1097E 01
V  3 =  0.1170E 02
V  4 = -0.1219E-00
```

```
I  1 =  0.1591E-04
I  2 =  0.1828E 01
I  3 =  0.1828E 01
I  4 =  0.1828E 01
I  5 = -0.1219E-00
I  6 =  0.1219E-00
I  7 = -0.1950E 01
I  8 =  0.1371E-10


SWITCH   1   IS OFF


   T =  0.1712E-03

V  1 =  0.1482E 02
V  2 =  0.1463E 02
V  3 =  0.1170E 02
V  4 = -0.1217E-00


I  1 =  0.1828E 01
I  2 = -0.1482E-04
I  3 =  0.1828E 01
I  4 =  0.1828E 01
I  5 = -0.1217E-00
I  6 =  0.1217E-00
I  7 = -0.1950E 01
I  8 = -0.3010E-06


   T =  0.1800E-03

V  1 =  0.1480E 02
V  2 =  0.1461E 02
V  3 =  0.1167E 02
V  4 =  0.1547E-01


I  1 =  0.1960E 01
I  2 = -0.1480E-04
I  3 =  0.1960E 01
I  4 =  0.1960E 01
I  5 =  0.1547E-01
I  6 = -0.1547E-01
I  7 = -0.1944E 01
I  8 = -0.3344E-06


   T =  0.1890E-03

V  1 =  0.1479E 02
V  2 =  0.1458E 02
V  3 =  0.1200E 02
V  4 =  0.8502E-01


I  1 =  0.2085E 01
I  2 = -0.1479E-04
I  3 =  0.2085E 01
I  4 =  0.2085E 01
I  5 =  0.8503E-01
I  6 = -0.8502E-01
```

```
I  7 = -0.2000E 01
I  8 = -0.2265E-10

SWITCH    1    IS ON

   T =   0.1890E-03

V  1 = -0.1043E 01
V  2 = -0.1251E 01
V  3 =   0.1200E 02
V  4 =   0.8494E-01

I  1 =   0.1604E-04
I  2 =   0.2085E 01
I  3 =   0.2085E 01
I  4 =   0.2085E 01
I  5 =   0.8491E-01
I  6 = -0.8494E-01
I  7 = -0.2000E 01
I  8 =   0.3007E-06

   T =   0.1929E-03

V  1 = -0.9147E 00
V  2 = -0.1098E 01
V  3 =   0.1170E 02
V  4 = -0.1207E-00

I  1 =   0.1591E-04
I  2 =   0.1829E 01
I  3 =   0.1829E 01
I  4 =   0.1829E 01
I  5 = -0.1207E-00
I  6 =   0.1207E-00
I  7 = -0.1950E 01
I  8 =   0.3171E-10

SWITCH    1    IS OFF

   T =   0.1929E-03

V  1 =   0.1482E 02
V  2 =   0.1463E 02
V  3 =   0.1170E 02
V  4 = -0.1205E-00

I  1 =   0.1829E 01
I  2 = -0.1482E-04
I  3 =   0.1829E 01
I  4 =   0.1829E 01
I  5 = -0.1205E-00
I  6 =   0.1205E-00
I  7 = -0.1950E 01
I  8 = -0.3010E-06
```

T = 0.2000E-03

```
V  1 =  0.1481E 02
V  2 =  0.1461E 02
V  3 =  0.1164E 02
V  4 = -0.4858E-02

I  1 =  0.1935E 01
I  2 = -0.1481E-04
I  3 =  0.1935E 01
I  4 =  0.1935E 01
I  5 = -0.4858E-02
I  6 =  0.4858E-02
I  7 = -0.1940E 01
I  8 = -0.3623E-06
```

U. S. ARMY ENGINEER DISTRICT, WALLA WALLA

WALLA WALLA, WASHINGTON

AN INTEGRATED SPS EARTHWORK SYSTEM FOR THE IBM 1620

A Paper Presented at the IBM 1620 Users Group Western Region Conference at Tempe, Arizona

12 December 1963

Prepared by

Cecil L. Ashley
ADP Coordinator, NPW

November 1963

CONTENTS

ABSTRACT

The purpose of this paper is to review the group of roadway design programs being utilized and developed by the U. S. Army Engineer Division, North Pacific, in cooperation with State Highway Departments of Washington, Oregon, and California, U. S. Bureau of Public Roads, U. S. Forest Service and IBM.

Programs discussed include horizontal and vertical control, data check, template design, alignment offset, quantity computation and several related utility programs, all of which are coded in SPS and make up an integrated system which is convenient as well as fast. The specific use of these programs by the Corps of Engineers is in conjunction with relocation of highways, railroads, county roads and municipalities involved in the design and construction of large multi-purpose hydroelectric projects.

The "System" of programs has been developed through a combination of initial development, modification of others' programs and adoption of others' programs with no modification. Coordination of programming efforts is accomplished largely through facilities of HEEP.

The abstracts of programs at the end of this paper are brief and general in nature. Writeups are complete but not necessarily in compliance with USERS GROUP Standards. Interested parties should address inquiries to:

Division Engineer
U. S. Army Engineer Division, North Pacific
210 Custom House
Portland 9, Oregon

PART I

GENERAL PROBLEM

Terrain. - The nature of the earthwork problems being encountered by the Walla Walla District may not be entirely unique, but they are certainly extreme at times. The Columbia and Snake Rivers have both cut deeply into Miocene basalt flows that form the regional bedrock. For 45 miles above the John Day Lock and Dam project, the Columbia River flows in a relatively deep valley with precipitous walls rising to a height of 3,000 feet above the general land surface on the north side, and up to 1,000 feet on the south side. The canyon walls consist of a series of stepped basalt cliffs interposed with steep talus slopes and terminating at the top in relatively flat plateau areas. Because of the constricted nature of the canyon, there is very little choice for the alignments; however, where the alignment is benched into the basalt edges, a minor shift in alignment can mean a major difference in cost because of the rock excavation and high fills which may occur.

Because of the type and shape of the terrain, alignment is quite critical and many trial alignments may be required in order to arrive at what is considered an optimum one.

Variety of Criteria. - Another feature that requires considerable attention is the number of agencies to be dealt with. In one project area there may be up to three State Highway Commissions, four railroads, and a dozen counties involved; all of which have from minor to major differences in criteria. This condition is emphasized when two railroads and a highway are aligned adjacently and there is hardly room for one of them in the canyon. Something has to move up the hill.

The John Day Lock and Dam presents a good example of the magnitude of the work being done in the North Pacific Division. This will be a concrete gravity structure with an ultimate installation of 2,700,000 KW produced by twenty 135,000 KW units driven by Kaplan turbines measuring 305 inches throat diameter. The total excavation for the project will be approximately 80,000,000 cubic yards, half of which will be hard rock. This total excavation figure is exceeded only by Owyhee, Fort Randall, Fort Peck, and Garrison project; all of which are earth or rockfill dams. As far as we know, the 40,000,000 cubic yards of hard rock excavation will comprise a world record.

The relocation at this project involves 80 miles of SP&S Railroad, 57 miles of UP Railroad, 40 miles of Washington State Highway, 32 miles of Oregon Interstate Highway, and the town of Arlington with a population of 633, 1960 census.

When the alignments involved are required to be revised up to 8 or 10 times in order to reach a compromise with economics and the agencies concerned, the workload approaches staggering proportions.

PART II

APPROACH TO PROBLEM

Acquiring Data. - In order to plan and design such a project and its accompanying relocation work, surface data must be translated into punched cards representing ground cross sections along the alignment centerline.

First, of course, an approximate alignment must be determined on paper. The horizontal alignment and geometric computation programs are utilized at this stage for horizontal control. Survey parties then either "flag" that alignment in the field, or establish control points. Helicopters are utilized to transport the party members to inaccessible locations. Aerial photographers are then hired to "fly" the line and the resulting film is converted to glass plates. Sufficient area is covered to enable the designer to shift the alignment or position several roadway alignments such as railroad mainline and shoofly, and highway and detour.

Data Translation and Preparation. - With the pictures taken and plates made, the next problem is to translate these to punched cards which are input media to the computer. This is accomplished by one of two ways, with emphasis on minimum manual work. For planning and design stages, if contour maps are available, alignments may be laid out on them and cross section cards punched by means of the Benson-Lehner Digital Scale. If contour maps are unavailable, or if for other reasons, it is desirable to do so, the cross section cards are prepared on a WILD A-7 Autograph Stereo Plotter connected to an IBM 026 Key Punch through a WILD EK-5. A hard copy is prepared concurrently on an electric typewriter. An optional feature of this system permits printing out on typewriter the X, Y, and Z coordinates at points selected.

For initial phases, cross sections are taken covering sufficient area to allow for alignment shifts and multiple alignments.

Once the cross section cards are punched, they are checked for "detectable bulls"by means of a computer pass using the "Data Check" program which senses such errors as overhangs, double or no centerline, excessive horizontal or vertical difference between rod readings, etc. as may be logically determined.

At this stage, cross sections can be plotted by the Benson-Lehner Electroplotter. To many designers, cross sections are considered obsolete but the terrain and materials with which we have to contend make cross sections desirable and often necessary.

Another means by which segments of cross sections are sometimes prepared is the Benson-Lehner Oscar Trace Record Reader. This machine uses fathometer graphs as input and produces punched cards in the same format produced by the Digital Scale or WILD A-7. This method is useful where the toe of proposed fill slopes extend below present water surfaces and in navigation channel and powerhouse excavation applications. A program is currently being developed by the Walla Walla District which will update cross section data cards by extending on either side, by inserting additional readings or replacing corrected readings.

PART III

INTEGRATED PROCESSING

Alignment and Realignment. - With the cross section cards prepared and checked, the question arises, "Has the alignment been revised during the process of preparing the data cards?" If so, the new alignment computation data is combined with the original or previous alignment data and processed through the Earthwork Offset Alignment Reference program which computes the offset and skew angle between the two alignments and new stationing. If the skew angle is not too great and the original cross section cards cover sufficient distance, the output cards of this computer pass serve as header cards which are combined with the cross section cards and processed through the Earthwork Alignment Shift program to compute new cross section cards referenced to the new alignment centerline.

Template Generation. - Roadway templates to be used for computing cut and fill quantities can be prepared on the computer at this stage through what is at present a two-pass process. The first pass establishes profile grade and the coordinates of the basic roadway points reflecting superelevation for horizontal curves. This basic roadway consists of two to four planes defined by from three to five points.

The second pass completes the templates by adding ditches where required, and slope readings. This pass also serves as an additional data check by indicating slopes that will not catch, etc. The complexity of road design involving berms on fill slopes for embankment protection, and benches on cut slopes for rock fall protection or due to material classification change, has prompted a recent modification to this program which enables the designer to specify these features on the terminal slopes of the template. Templates can be produced with either slope readings or catch point coordinates at extremities.

Quantity Computation. - For computing quantities, the cross section and template cards are merged and processed through the Earthwork Quantity Computation program. The output of the pass provides cut and fill areas, accumulated cut and fill quantities and mass ordinate. This program also has optional features for dredging and levee applications, in which case, directed slopes and outside catch points are used. It will also make quantity adjustments for curvature correction.

Design for Contract. - Listings of this output are provided the designer who then determines if and where alignment needs to be revised vertically or horizontally. This is repeated usually for various segments of the line, then when the alignment is satisfactory, the complete line is recomputed for quantities.

Special Options. - If two alignments are adjacent and the embankments sometime overlap, the Stage Development program is used to combine the cross section and template cards to produce new cross section cards which can then be line-shifted and used to compute berms for riprap which is to be computed separately from basic quantities.

The answer cards from the quantity run can be processed through the Station Interval Quantity summary to produce listings of quantity summations at selected intervals.

Cross section and template cards are processed through the Earthwork Data Plot Reduction program to replace template slope readings with catch point coordinates and reference all vertical coordinates to a common elevation so the cross sections and templates can be plotted.

Pay Quantities Computation. - During and after construction when pay quantities are to be computed, the line is re-flown and "as-built" templates prepared. These cards differ from the previously mentioned templates in that they do not have slope readings but extend from catch point to catch point. These templates are merged with cross section cards of the final alignment and pay quantities are computed through the Earthwork Quantity Computation program.

The "as-built" template cards are point-plotted over the "as-designed" cross sections and used to check for over- or under-built conditions.

PART IV

RESULTS

Opinions of the users of these programs are varied, so in order to avoid giving an overly optimistic view, this example was taken from the most "conservative" user. Sixty miles of railroad alignment on the Little Goose Lock and Dam project was processed through eight alignments before arriving at the final choice. By thoroughly analyzing the first seven alignments, five million yards of rock excavation were saved between the seventh and eighth alignments. If hand methods had been used, there would not have been time for more than four "shotgun" alignments and the savings resulting from computer utilization would not have been possible.

The cost involved in design stages may be equal to or even greater than that incurred by "shotgun" manual methods, but this includes increased analyzation of alignments, making possible savings such as cited in the example. Even greater savings are being realized with a computer on-site in the Walla Walla District so that even more alignment trials can be made in certain areas where the alignment is most problematical.

One installation has combined template design and quantity computation programs enabling them to obtain quantities in one pass without producing intermediate templates. This version also incorporates line shifting as an integral part of the single pass. It requires a different format of data cards than that being used by our District and 60 K. The type of processing usually encountered in the Walla Walla District require some insertion of hand-prepared templates before the final pass, so we have not seriously considered their degree of combination. The two passes of template design might be combined to advantage, providing header card storage did not become too restricted.

PART V

ABSTRACTS OF PROGRAMS

1. HORIZONTAL ALIGNMENT - This program computes horizontal alignment from the following basic input alignment data:

   a. Beginning station and coordinates (RP)

   b. Coordinates of PI's

   c. Degree or radius of curves

   d. Spiral lengths (highways) or chord lengths and number (UPRR)

Output information includes:

   a. Bearing and length of course between PI's

   b. Stations and coordinates of all curve points

   c. Deflection angle (Δ) and semi-tangents

   d. Degree, length and radius of circular curves

   e. Deflection angle (DE), x and y, u and v of spirals

Curve definition can be by arc or chord. Distances are rounded to nearest .01 foot, coordinates are carried to nearest .001 foot.

2. <u>GEOMETRIC COMPUTATION</u> - The present geometric program combines the features of several related programs which were used on the IBM 650. The following types of computations can be performed singly or in combination in one machine pass.

a. Survey traverse using bearings with azimuth or deflection angles.

b. Compass or transit rule traverse adjustment.

c. Areas bounded by traverses which may contain circular segments.

d. In a traverse between known points, a combination of two unknowns (bearings, 2 distances or a distance and a bearing) may be solved for and stored for recall in subsequent problems in same pass with interdependency of geometric figures.

e. Location of points on tangents, circular curves, spirals or offset spirals and intersections of any two of these.

f. The intersection of tangents or circular curves with spirals or offset spirals.

g. Convert selected coordinates from any one plane system to another.

The ability to store data pertaining to any course for use in problems which follow makes it possible for the engineer to begin with a minimum of data and to solve complex office and field layout problems in one pass on the computer.

Problems are arranged in logical sequence to develop intermediate answers as would be done by "people-computers" following traditional methods.

A closed traverse can be adjusted in one problem, for instance, and an adjusted loop can be run from it to close on any point for which coordinates are known or have been computed. Two alignments can be intersected and the bearing and distance to land corners or other references readily determined. The area of gusset plates can be determined and a host of other geometric problems can be solved. As a matter of fact, some users say "it can do anything!"

3. <u>EARTHWORK DATA CHECK</u>

This program checks earthwork input data in format used by U. S. Army Engineer Division, North Pacific, and its various Districts. Errors which would result in erroneous results or machine halts are detected and error messages typed or punched identifying the station and specifying the type of error detected.

4. <u>PROFILE GRADE AND TEMPLATE GENERATION</u>

This program converted from Washington State Highway Department's Five-Point Profile Grade Program, produces the following results:

a. Generates basic roadbed templates for input to Slope Selection - Template Generation Program.

b. Produces pavement elevation cards supplying elevation of five points at each station to .01 foot accuracy.

c. Computes profile grade at selected or incremental stations.

d. Reproduces templates inserting new profile grade elevation in the reference elevation field.

e. Punches a summary of vertical control data at end of each run.

5. <u>SLOPE SELECTION - TEMPLATE GENERATION</u>

This program requires ground line cross section cards plus the template cards produced by the Profile Grade Program as basic data. Other input consists of slope specification cards and bench-berm data cards. Three fill slopes and four cut slopes may be specified for appropriate depths of fill or cut. Ditches can be specified with V or flat bottom with depth and slope optional. Terminal slopes may include bench (cut) or berm (fill).

The templates produced may include catch points, or the terminal rod readings may be replaced with slope readings in the form xxx.xx horizontal to 1 foot vertical.

These templates when collated with ground line cross section cards are the input to the Earthwork Quantity Computation Program.

6. EARTHWORK QUANTITY COMPUTATIONS

a. The purpose of this program is to compute quantities of earthwork to be excavated and placed on a given job. The program is patterned after the IBM 650 Cut and Fill Program (H-841) as modified by Bureau of Public Roads, Vancouver, Washington. The computations are performed in the conventional manner by taking the original ground topography and the design of the completed section, as specified by the engineer, and computing the cut and fill end areas. These are then used to compute the volumes between sections by the Average-End-Area method. This program will apply a volume correction due to curvature if desired. Swelling or compaction can be applied to either the cut or the fill and these quantities are accumulated throughout a project to produce a mass-ordinate. The total accumulated quantities of unadjusted cut and fill are also available for each section.

b. As a by-product of these computations, some design information is available at each section to aid the engineer in his appraisal of the design that has just been computed. These include the cut or fill at the pivot point, at the toe of both the right and left slope, and at centerline. Also, the rod and distance to the catch point of the template slope and the original ground line, if these were not part of the input data.

7. EARTHWORK LINE SHIFT

The purpose of this program is to shift topog card data (type "0") to a new or offset centerline. This enables the engineer the choice of running multiple trial alignments while using the original topog data but shifting the topog data to coincide with the alignment. The program can also be used for station adjustment and for the reverse line shifting of topog data. It is desirable in some phases of earthwork to shift the topog data from the original line to an offset line, generate templates at the offset line, stage the ground line, and then shift the staged ground back to the original line. By proper program control in the Job Control Card, the original line shift headers (used to shift original topog to an offset line) and the staged ground at the offset line can be used to shift the staged ground back to the original line.

8. EARTHWORK TEMPLATE SHIFT

This program is designed to take the Type "1" templates that were prepared or generated at an offset centerline and punch new templates having original centerline stationing and shot distances referenced to the original centerline. The new Type "1" templates can then be collated with the original terrain cards (Type "0") and a plotter deck is obtained which has the original centerline as a base for plotting of both terrain data and Type "1" templates. Use of this program will facilitate plotting operations when more than one roadway section is planned and is to be plotted on the same cross section of terrain.

9. <u>EARTHWORK ALIGNMENT OFFSET</u>

This program is designed to compute offset distances, skew angles, and centerline station coordinates of a base line and an offset line. Output also includes line shift headers which enables the Earthwork Line Shift Program to produce a new set of ground line cards referenced to the offset line.

10. <u>EARTHWORK DATA PLOT REDUCTION</u>

This program reproduces ground line and template cards adding plotter signs where necessary and references both type of cards to a common reference elevation. It will also provide demand origin offset codes and insert blank cards where necessary.

11. <u>CENTERLINE DATA PLOT REDUCTION</u>

This program compiles station numbers and centerline elevations of ground line cross sections or template cards in a format for plotting of centerline profile.

12. <u>STATION INTERVAL QUANTITY SUMMARY</u>

The purpose of this program is to produce a summary of quantities of earthwork to be excavated and/or placed on a given job. Two types of summary options are available: Station or Interval. The listings are used by the engineer for design analyses and as tabulated quantity listings for plan and profile drawings for design memorandums and/or contract drawings.

13. <u>PLANE COORDINATE CONVERSION - PHOTOGRAMMETRY LEAST SQUARES METHOD</u>

The purpose of this program is primarily to convert the machine coordinates of a universal stereo-plotter to a local coordinate system, applying the least squares method of conversion (Helmert).

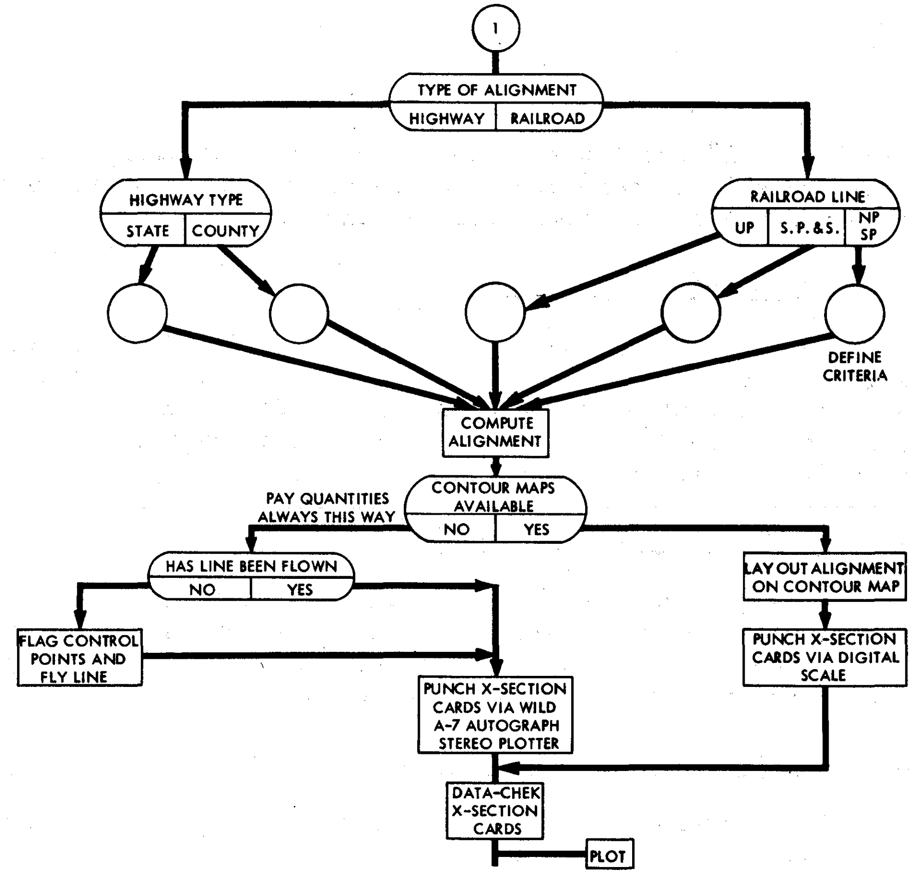
This conversion is mathematically expressed as follows:

$$x = p + k X \cos a - k Y \sin a$$

$$y = q + k Y \cos a + k X \sin a$$

The quantities p and q are ground coordinates of the origin of the machine coordinate system, a is the angle of rotation between the two systems, and k is the distance scale factor.

PREPARATION OF X-SECTION CARDS



COMPUTATION OF CUT AND FILL QUANTITIES

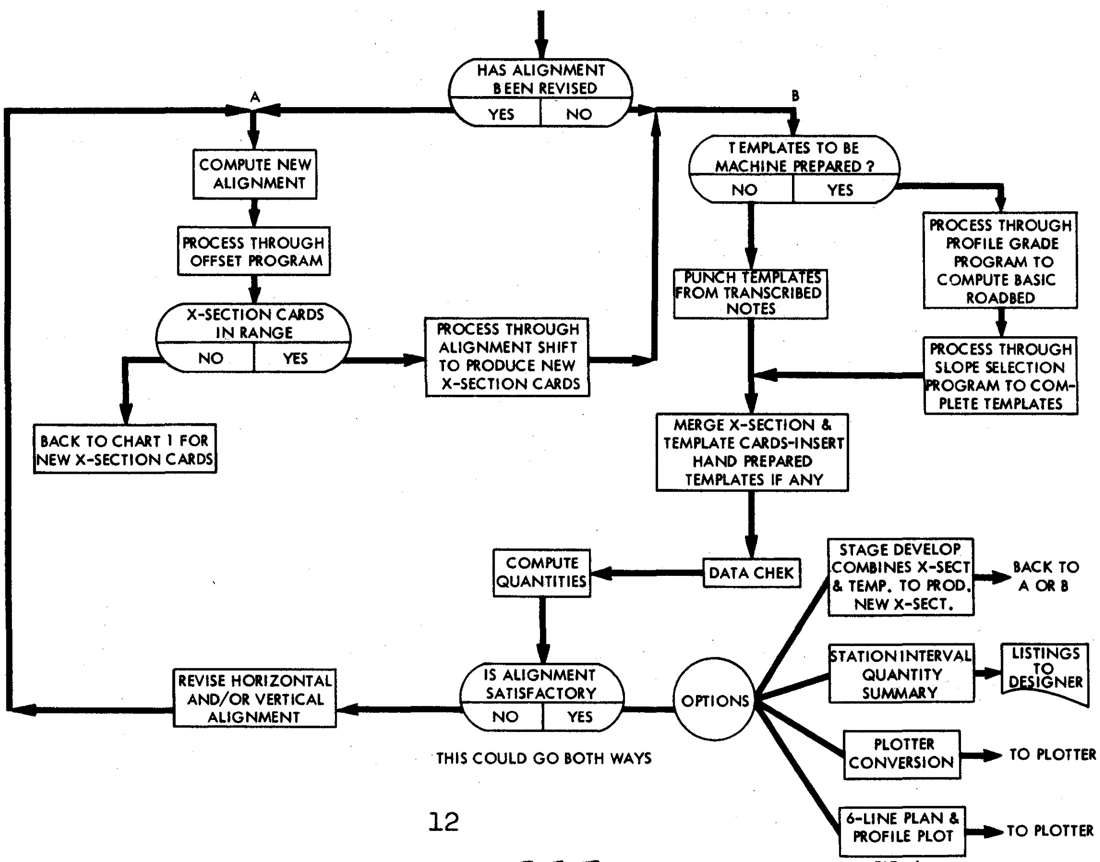


FIG. 1

HYDRO SYSTEM DAILY OPERATION ANALYSIS PROGRAM

CHARLES R. HEBBLE
U.S. ARMY ENGINEER DISTRICT, WALLA WALLA
WALLA WALLA, WASHINGTON

HYDRO SYSTEM DAILY OPERATION ANALYSIS PROGRAM

by

Charles R. Hebble 1/

INTRODUCTION

The computer program described in this paper, along with certain fixed input data, constitutes a detailed mathematical model of a system of hydroelectric projects. It is a general program applicable to any river system and scheme of development. Natural lake and channel storage may be synthesized in addition to reservoir storage and backwater effects. The program is capable of accurately simulating the hour-by-hour power loading and water regulation of a hydro system.

Through use of the program it is possible to determine the operating characteristics of planned future projects for various-sized power installations. Backwater encroachment on upstream reservoirs, pondage requirements, and the effects of peaking discharges on downstream river stages and reservoirs may be evaluated.

The program may also be used to compare alternative distributions of system load among an existing group of hydraulically and electrically integrated projects. It is thus possible, through a trial and error approach, to determine optimum load distribution.

The program has been used in investigating existing system development, along with future planned developments of the lower Columbia and lower Snake Rivers. Here a series of run-of-river projects, below Grand Coulee Dam, will develop almost all of the available head (see Figures 1 and 2). It is anticipated the program will be used to analyse future operating problems in regards to power loading and water regulation.

DESCRIPTION OF EQUIPMENT

Two versions of the program exist: one for an IBM 1620 having a 40,000 digit memory with an IBM 1622 Card READ-PUNCH for input-output; the other for an IBM 1920 System (combined IBM 1620/1401) having a 60,000 digit memory in the IBM 1620 and 4,000 digits in the IBM 1401. This latter system has both an IBM 1402 Card READ-PUNCH and an IBM 1403 on-line printer for input-output. Both systems have the following optional features required for program execution: Transfer Numeric Strip, Transfer Numeric Fill, Move Flag, and Indirect Addressing.

1/ Civil Engineer, Automatic Data Processing Section
U. S. Army Engineer District, Walla Walla

Peripheral equipment required is as follows: card keypunch, card verifier, and card sorter. In addition, for the system with the IBM 1622, a card tabulating machine with a special wired panel is required.

ANALYSIS AND METHOD OF SOLUTION

General. - The program is intended for the detailed hour-by-hour analysis of hydro system operations; basic input to the program and output from the program is hourly. However, periods greater or less than one hour may be used. There is no limit to the number of real-time operating intervals which can be simulated by the computer. The program continues to run as long as there are input data in the READ hopper. Ordinarily, however, real-time periods of more than a week are not analyzed because of the excessive length of the computer runs for all but the smallest systems. As a rough estimate, the program requires two minutes to compute one day's output for each hydro project in the system. The number of reaches and projects that can be analyzed in a given system is dependent on the computer storage capacity. The program instructions require approximately 25,000 digits of memory. A rule-of-thumb estimate for a given hydro system, including open river reaches, is 1,000 digits of memory for each hydro station. Variation of memory requirement is due to optional lengths of tables used to define system parameters.

Program Operating Modes. - Discharges from the various hydro projects are specified, either directly or indirectly as input to the program for each real-time period. Four operating modes exist for this purpose: (1) station power loading given; (2) system power load plus generation allocation (breakpoint) settings given; (3) total project discharge given; and (4) project forebay elevations specified. Any of the foregoing, in combination with other fixed and variable data, determines the individual project discharges. Different projects may be operated under different operating modes at the same time, and the operating mode of any project may be changed at any time.

Variable Input. - Each hydro station obtains operating data for each real-time period from coded variable input cards. Variable input data include project power loads (or discharges), system loads with project breakpoint settings for load-frequency control, local inflow, optional spill, and number of generating units synchronized on the line and available. These data may be varied as desired during the course of a run.

Operation Sequence. - The program begins with the upstream project of the series and proceeds downstream, routing flows through each open channel reach or reservoir. Local inflow between projects is added to the routed flows. The outflow from the projects is computed as the sum of power discharge, spill, and average fixed release. (The average fixed release, which must remain constant throughout the run, includes losses due to lockage, leakage, and useage.) The entire process is repeated for successive 15-minute computation (routing) periods. Project and desired reach answers are obtained upon completion of the fourth routing period. The variable data for the next hour is then read in and the entire process repeated. Routing and output intervals may be varied by a special job definition routine. The routing interval may be different for different projects or river reaches but must be a multiple of the basic routing interval. The computer program has two fundamental parts. One is the method of simulating a hydro-power station, the other is the flow routing procedure. Each part will be considered separately and in some detail.

Hydro-Power Station Simulation. - Each generating station synthesized in the program is represented by individual unit characteristics as illustrated in Figure 3. This is in contrast to the method used in other power programs for coarser increments of time which consider only generating station characteristics as a whole. Such programs assume some operating efficiency between best efficiency and, say, full-gate efficiency. This assumption, while adequate for studies having a basic time period of day, week, or month, during which there are many swings in station power loadings, are inadequate for the detailed hourly computations required in the study of peaking operations. The number of generating units synchronized on the line may be specified as variable input or may be automatically selected by the program. The program has the ability to automatically add units as required to meet the load up to the maximum number available. When a power load in excess of the maximum capability of the installation is specified, generation will automatically be adjusted to equal plant capability as limited by head. The output is coded to indicate this alteration by the computer program.

Under automatic selection of units the program computes the number of units for best efficiency operation. The total station load (or discharge) is divided by the unit best-efficiency loading (or discharge) for the particular head existing on the station to determine the desired number of units. Fractional numbers of units are truncated; thus unit loadings are at or higher than the best efficiency point. The number of units selected is limited to the maximum available.

The program automatically causes all excess water above maximum pool elevation to be spilled. This is termed "mandatory" spill. Power discharge, fixed release, optional spill and mandatory spill are summed to arrive at the total project discharge. The alarm section of the output is coded calling attention to such mandatory spill.

Should a project attempt to draft below its minimum pool elevation as a result of releases exceeding inflow; the power discharge and hence, generation is automatically reduced to prevent the overdraft, irrespective of the power demand or discharge called for by the input data. Here again the alarm section of the output is coded.

The computation of power, discharge, head, etc. are accomplished by means of table look-up and interpolation. For existing projects, unit performance tables may be prepared from observed data. In the case of planned projects, the basic data must first be computed. A separate computer program has been developed which can quickly compute performance characteristics. The program is based on the turbine performance characteristics of a unit machine, i.e., a machine runner diameter of one foot operating under a net head of one foot. These data are summarized in a unit performance hill in which power output is plotted against the peripheral speed coefficient. It is thus possible to quickly evaluate the performance of different sized units and units having different characteristics.

Centralized Load-Frequency Control. - A feature of the program is a provision for simulating centralized control of the station power loadings in a manner similar to that of centralized load-frequency control equipment. Such equipment is presently installed at the major Federal hydro stations on the Columbia River. The power loadings of these projects are centrally controlled from the system load dispatcher's office in Portland, Oregon.

When the load-frequency control feature of the program is used, participating settings for each of the hydro stations are then included as input data and a single system load rather than individual station loadings are specified. The system load is apportioned among the hydro stations in accordance with station participation values. Other than this, the program operates in the same manner as when individual loadings are prescribed for the several hydro stations.

This centralized, automatic dispatching of power loadings affords the opportunity for efficient coordinated system operation of the Federal projects. Use of the load-frequency control feature of the computer program will allow planning studies and scheduling of power operations in conjunction with the equipment, in addition to its other uses. It is envisioned that ultimately the load dispatching of these hydro stations will be directly and automatically controlled by computer. The present program could well serve as a basis for such a future program.

Flow Routing. - The total discharge from a project (power discharge, spill, and fixed release) is routed downstream by a method of flow routing known as incremental storage routing. This approach consists of subdividing reservoirs or river channels into incremental reaches. Each incremental reach is represented in the program by two tables: one giving the relationship between storage and elevation for the particular reach; the other giving the relationship between discharge and elevation. By proper selection of routing interval and choice of the number of reaches, actual hydraulic conditions may be accurately approximated. The discharge from any reach is computed as a function of the reach elevation, or where backwater effects must be considered, as a function of the next downstream reach as well. The change in storage content of the incremental reservoirs is related to the difference between inflow and outflow.

The combined effect of the storage-elevation and discharge-elevation relationships mentioned in the preceding paragraph causes a time constant to be introduced into the routing. This is known as "time of storage" and is defined as the change in storage per unit change in discharge.

$$Ts = \frac{\Delta S / \Delta Z}{\Delta Q / \Delta Z} = \frac{\Delta S}{\Delta Q}$$

where,

Ts = time of storage
S = storage volume
Q = discharge rate
Z = elevation

Dimensionally, using the units adopted for program use,

$$Ts\ (\text{min}) = \frac{\Delta S(\text{kcfs-min}) / \Delta Z(\text{feet})}{\Delta Q(\text{kcfs}) / \Delta Z(\text{feet})}$$

The routing procedure used in the program does not explicitly consider time of storage; however, the routing interval used in the program must be shorter than the minimum time of storage for any given reach. Failure to observe this criterion will result in oscillations of the reach elevation and discharge. These oscillations tend to increase in amplitude until the range of tables is exceeded. A reservoir which is encroached upon by backwater from a downstream reservoir may have an extremely short time of storage.

The time-of-storage concept may be visualized by assuming a constant rate of increase in inflow to an incremental reservoir. Outflow from the reservoir will eventually reach an equilibrium condition where it is increasing at the same rate as the inflow; at this time, the outflow hydrograph will be displaced from the inflow hydrograph by the time of storage. Figure 4 illustrates a discharge hydrograph routed through six identical incremental reservoirs. Storage-elevation and discharge-elevation curves are shown on this same sheet.

Overlapping reservoirs are normally subdivided into two reaches: a forebay reach and a tailwater reach for the next upstream project. The sum of the storages of the individual reaches must equal the actual reservoir storage.

In subdividing open channel reaches into incremental reaches, no definite rules can be given for the number of increments. In general, the greater the number of reaches, the greater the translation of the discharge hydrograph with minimum attenuation. Discharge hydrographs can be modified, to a lesser degree, by choice of routing interval. Here again, selection of routing interval beyond that required to prevent oscillation, is subject to rules established for particular problem definition.

For river systems where all tailwater reaches have extremely short times of storages, an alternate routing method is available. This method permits use of relatively long routing intervals, up to the answer output interval if desired. In effect it considers the time of storage to be zero. Tailwater elevation is first determined as a function of inflow (and next downstream reach elevation if necessary); the storage corresponding to the new tailwater elevation is then computed as is the change in storage during the routing interval; this change in storage is converted to an average rate of flow during the routing interval and added (or subtracted) to tailwater inflow to arrive at the outflow. The algebraic sum added to the inflow is dependent on change of tailwater elevation during the routing period, i.e., a rising tailwater will cause outflow to be less than inflow with the opposite occurring on a declining tailwater condition.

The flow and storage-routing procedure used in the program has a sound theoretical basis which permits analysis of future reservoirs as well as existing reservoirs and river reaches. The computation of discharge, elevation, storage content, etc. are accomplished by means of table look-up and interpolation. Volume-elevation relationships for open river reaches and reservoirs are generally obtained from survey data. Water surface profiles are computed for various forebay elevations and rates of flow by independent means. (Here again separate computer programs are available for determining water surface profiles, and volume relationship, in reservoirs or open channel reaches.) From these profiles, the discharge-elevation-volume relationships are derived.

PROGRAM OUTPUT

Output Data. - Program output consists of the following data for each project:

Date and time
Abbreviated project name

End-of-period forebay elevation
End-of-period tailwater elevation
Average head
Forebay storage content
Storage change during period
Local inflow
Routed inflow
Power discharge
Spill
Total discharge
Generation
Number of units operating
Alarm codes
Operating mode number
Project number

Output may be obtained for selected reaches, in addition to project data, at users option. Reach output consists of the following data:

Date and time
Abbreviated reach name
End-of-period elevation
End-of-period total discharge
Reach number

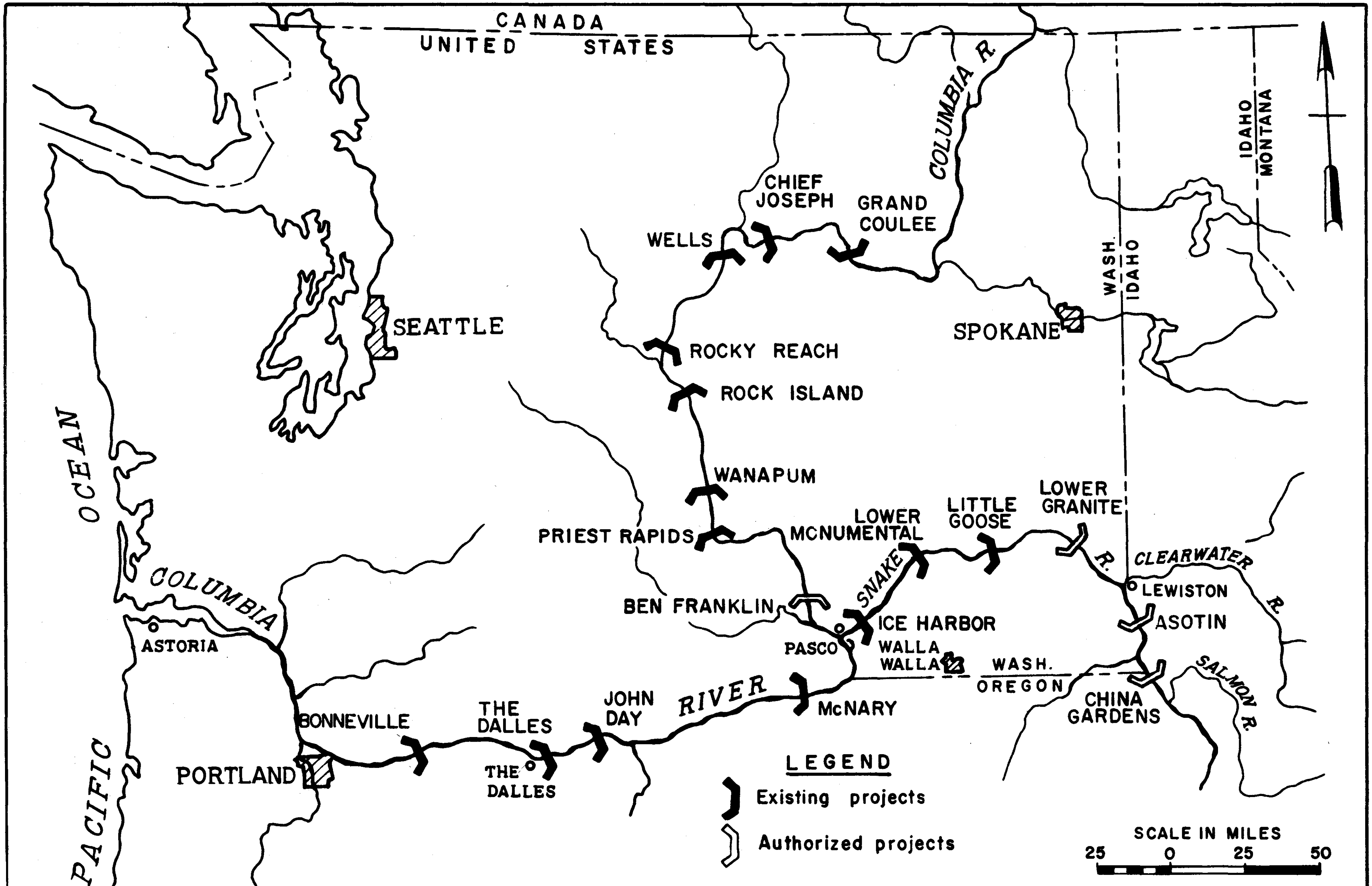
The 1920 System program output consists of both on-line printing and punched cards. The same data are given by both forms of output. Choice of output mode is optional with user. The 1620 System output is punched cards. The 1920 System uses an auxiliary listing program to list the punched cards for each project or optional reach and system Summary Cards.

Output data consists of Project Cards, selected Reach Cards and Summary Cards containing end-of-period system generation totals and system potential energy remaining in storage. The sequence of the punched card output and on-line printout is as follows: An initial Summary Card gives potential energy in the system based on starting profile data. This is followed by data for each hydro project and for optional selected reaches for the initial time period. A Summary Card concludes the data for the initial time period. Project, Reach and Summary data are then punched out for the next subsequent output interval. Output intervals may be varied by special control. In the absence of special control, the output interval is one hour. Sample output formats of a typical study are shown on Figure 5 for the IBM 1620 System and on Figure 6 for the IBM 1920 System.

ACKNOWLEDGEMENTS

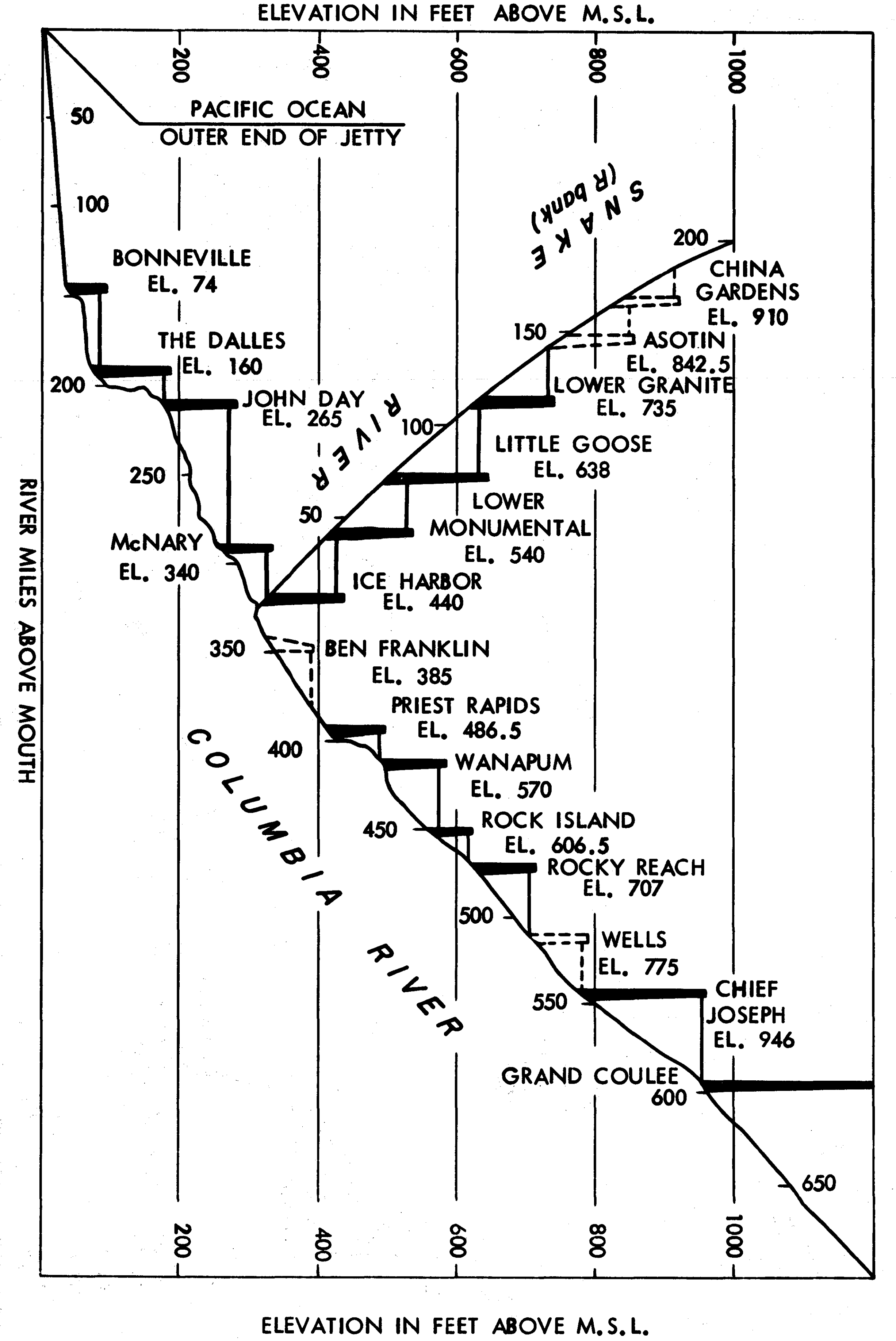
The original program was developed under the direction of Mr. C. E. Hildebrand, Water Control Branch, U. S. Army Engineer Division, North Pacific, Portland, Oregon. Michael A. C. Mann, Consulting Engineer, assisted in the analysis and programmed the basic program and supplementary routines for the IBM 650. The author wishes to acknowledge the work of Mr. Lyle A. Dunstan of H. Zinder and Associates, who programmed the present version. This work was financed by the U. S. Bureau of Reclamation and the program used by them in their planning studies. Additional logic, and program refinement was accomplished as a joint effort by U. S. Army Engineer Division, North Pacific and Walla Walla District, Corps of Engineers. The assistance of C. E. Hildebrand and Robert D. Moffitt of the U. S. Army Engineer Division, North Pacific in accomplishing the latter, and in the preparation of this paper is greatfully acknowledged.

Requests for program details should be directed to: Division Engineer, U. S. Army Engineer Division, North Pacific, 210 Custom House, Portland, Oregon, 97209.



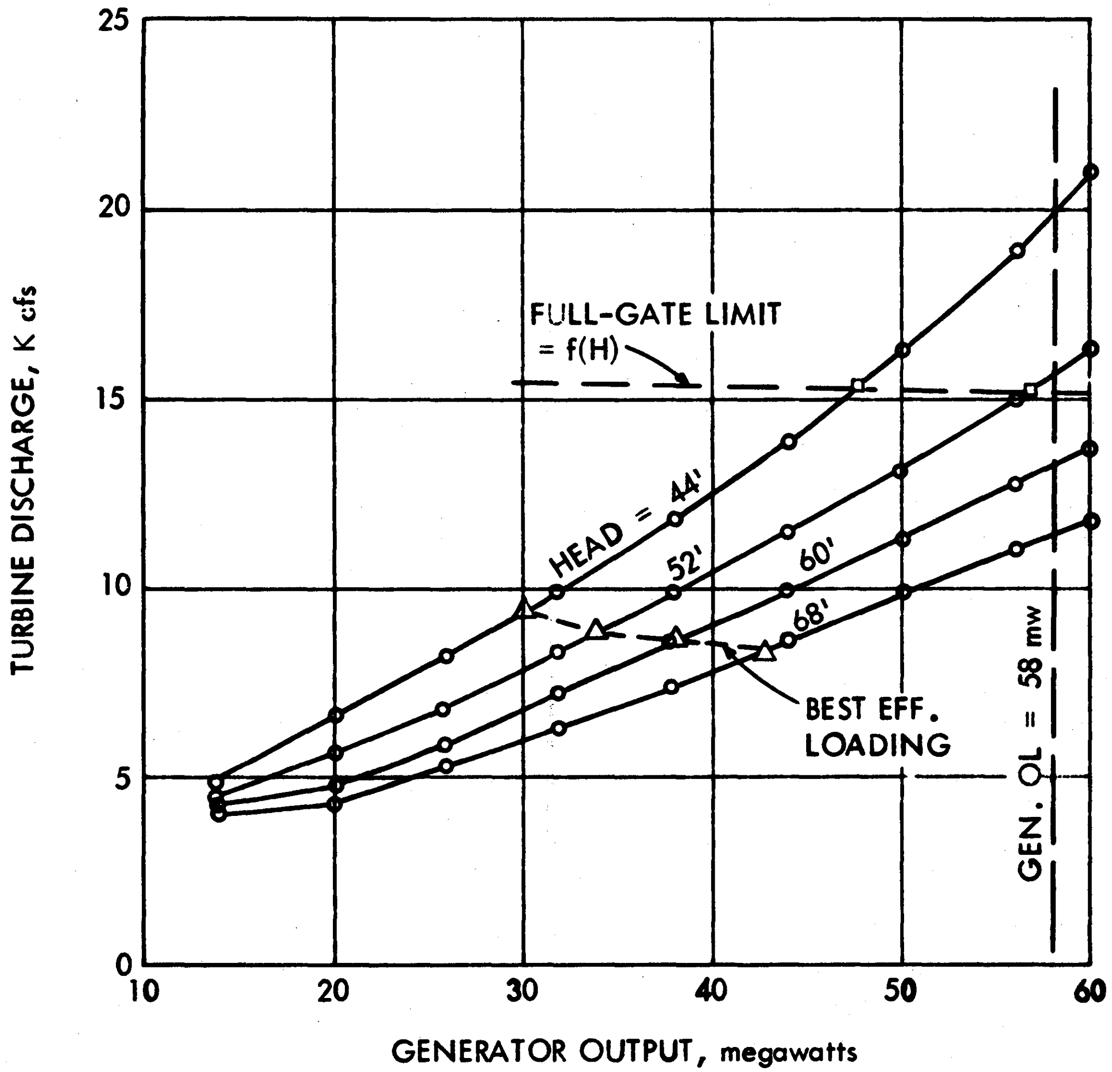
EXISTING AND FUTURE PROJECTS OF THE LOWER COLUMBIA AND SNAKE RIVERS

Fig. 1



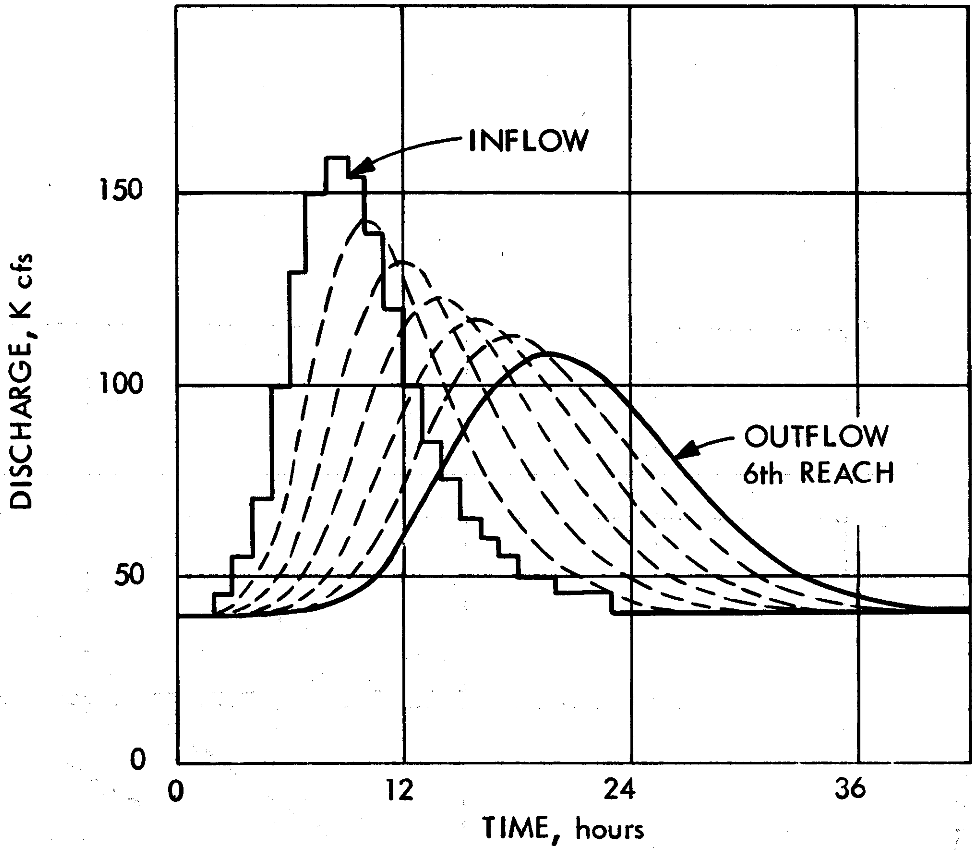
RIVER PROFILES, LOWER COLUMBIA AND SNAKE RIVERS

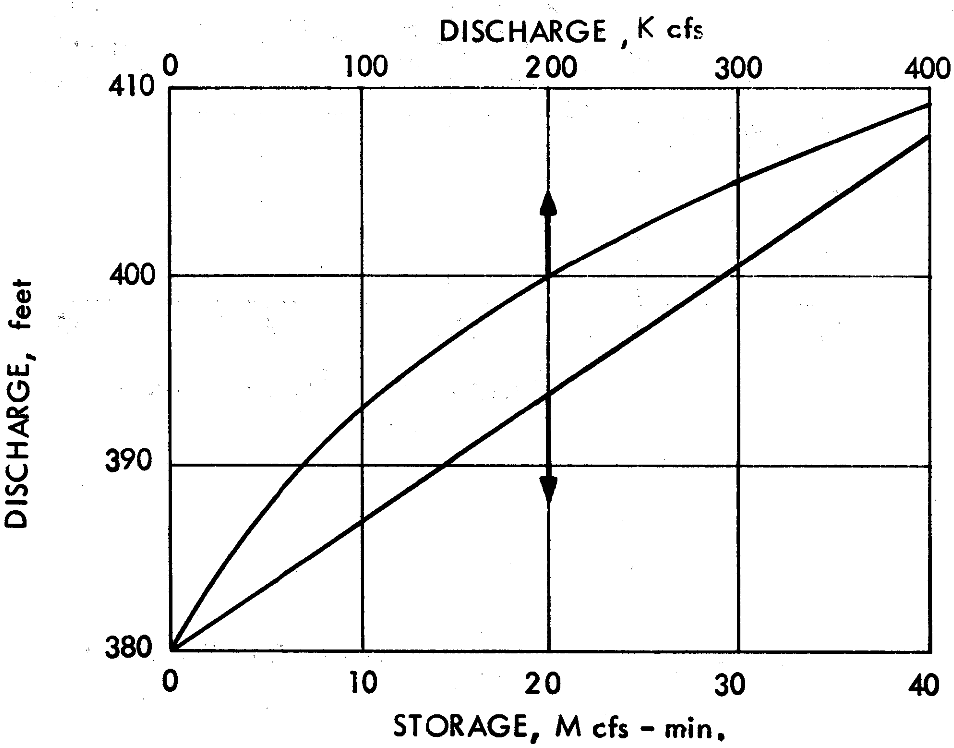
Fig. 2



GENERATING UNIT CHARACTERISTICS

Fig. 3





INCREMENTAL RESERVOIR ROUTING EXAMPLE WITH STORAGE - ELEVATION AND DISCHARGE - ELEVATION CURVES

Fig. 4

DATE-TIME	PROJ	FOREBAY FEET	TAILWATER FEET	AVG HEAD	FOREBAY STOR	STOR CHANGE	LOCAL INFLOW	ROUTED INFLOW	POWER DISCH	SPILL DISCH	TOTAL DISCH	GEN MW	NO. UNITS	C
15 1 85 100	GCL	1262.08	945.41	315.30	204,062	1,313		37,750	5,740		6,240	133	2.0	2
15 1 85 200	GCL	1262.16	944.47	317.19	205,485	1,423		36,700	2,050		2,550	45	1.0	2
15 1 85 300	GCL	1262.24	944.05	317.94	206,991	1,506		36,650			500			2
15 1 85 400	GCL	1262.33	943.89	318.27	208,497	1,506		36,650			500			2
15 1 85 500	GCL	1262.40	944.48	318.02	209,765	1,268		36,620	5,690		6,190	133	2.0	2
15 1 85 600	GCL	1262.44	945.99	317.05	210,496	731		36,650	18,600		19,100	439	4.0	2
15 1 85 700	GCL	1262.41	950.20	314.43	209,903	594-		36,620	50,370		50,870	1,197	13.0	2
15 1 85 800	GCL	1262.32	953.79	310.61	208,301	1,602-		36,650	74,590		75,090	1,749	20.0	2
15 1 85 900	GCL	1262.20	955.99	307.45	206,310	1,991-		36,650	83,930		84,430	1,944	22.7	2
15 1 85 1000	GCL	1262.09	957.01	305.73	204,295	2,015-		36,620	84,490		84,990	1,944	23.0	2
15 1 85 1100	GCL	1261.97	957.57	304.79	202,268	2,026-		36,650	84,780		85,280	1,944	23.0	2
15 1 85 1200	GCL	1261.86	957.89	304.21	200,233	2,035-		36,620	84,970		85,470	1,944	23.0	2
15 1 85 1300	GCL	1261.76	957.38	304.13	198,556	1,677-		36,650	76,380		76,880	1,749	21.0	2
15 1 85 1400	GCL	1261.68	956.50	304.71	197,192	1,364-		36,650	68,890		69,390	1,580	18.5	2
15 1 85 1500	GCL	1261.61	955.83	305.43	195,930	1,262-		36,620	66,420		66,920	1,527	18.0	2
15 1 85 1600	GCL	1261.53	955.84	305.74	194,470	1,460-		36,650	71,190		71,690	1,638	19.0	2
15 1 85 1700	GCL	1261.41	956.91	305.18	192,448	2,022-		36,620	84,650		85,150	1,944	23.0	2
15 1 85 1800	GCL	1261.29	957.52	304.19	190,413	2,034-		36,650	84,970		85,470	1,944	23.0	2
15 1 85 1900	GCL	1261.18	957.85	303.58	188,371	2,042-		36,650	85,160		85,660	1,944	23.0	2
15 1 85 2000	GCL	1261.07	957.72	303.33	186,491	1,880-		36,620	81,240		81,740	1,853	22.0	2
15 1 85 2100	GCL	1260.99	956.69	303.74	185,117	1,374-		36,650	69,130		69,630	1,580	19.0	2
15 1 85 2200	GCL	1260.96	954.43	305.17	184,555	562-		36,620	49,610		50,110	1,139	13.0	2
15 1 85 2300	GCL	1260.97	951.15	307.96	184,797	242		36,650	30,340		30,840	704	8.0	2
15 1 85 2400	GCL	1261.02	947.86	311.19	185,524	728		36,650	18,690		19,190	439	5.0	2
		30281.97		7415.34		17,223-		880,510				29,513		
			22866.42		760,474				1,281,880		1,293,880		346.2	

Figure 5

C-1 UNITS (F)

HYDRO SYSTEM DAILY OPERATION ANALYSIS
U. S. ARMY ENGINEER DIVISION, NORTH PACIFIC

GRAND COULEE

DATE AND TIME	ELEV. END PERIOD FOREBAY FEET	ELEV. END PERIOD TAILWTR FEET	AVG. HEAD FEET	FOREBAY STORAGE CFS-DYS	DELTA STORAGE CFS-DYS	INFLOW LOCAL CFS	INFLOW ROUTED CFS	INFLOW TOTAL CFS	DISCHARGE POWER CFS	DISCHARGE SPILL CFS	DISCHARGE TOTAL CFS	AVG. GEN. MW	AVERAGE NUMBER UNITS	OP. CD *
15 01 85 0100	1262.08	945.41	315.30	204,062	1,313	0	37,750	37,750	5,740	0	6,240	133	2.0	2
15 01 85 0200	1262.16	944.47	317.19	205,485	1,423	0	36,700	36,700	2,050	0	2,550	45	1.0	2
15 01 85 0300	1262.24	944.05	317.94	206,991	1,506	0	36,650	36,650	0	0	500	0	0.0	2
15 01 85 0400	1262.33	943.89	318.27	208,497	1,506	0	36,650	36,650	0	0	500	0	0.0	2
15 01 85 0500	1262.40	944.48	318.02	209,765	1,268	0	36,620	36,620	5,690	0	6,190	133	2.0	2
15 01 85 0600	1262.44	945.99	317.05	210,496	731	0	36,650	36,650	18,600	0	19,100	439	4.0	2
15 01 85 0700	1262.41	950.20	314.43	209,903	594-	0	36,620	36,620	50,370	0	50,870	1,197	13.0	2
15 01 85 0800	1262.32	953.79	310.61	208,301	1,602-	0	36,650	36,650	74,590	0	75,090	1,749	20.0	2
15 01 85 0900	1262.20	955.99	307.45	206,310	1,991-	0	36,650	36,650	83,930	0	84,430	1,944	22.7	2
15 01 85 1000	1262.09	957.01	305.73	204,295	2,015-	0	36,620	36,620	84,490	0	84,990	1,944	23.0	2
15 01 85 1100	1261.97	957.57	304.79	202,268	2,026-	0	36,650	36,650	84,780	0	85,280	1,944	23.0	2
15 01 85 1200	1261.86	957.89	304.21	200,233	2,035-	0	36,620	36,620	84,970	0	85,470	1,944	23.0	2
15 01 85 1300	1261.76	957.38	304.13	198,556	1,677-	0	36,650	36,650	76,380	0	76,880	1,749	21.0	2
15 01 85 1400	1261.68	956.50	304.71	197,192	1,364-	0	36,650	36,650	68,890	0	69,390	1,580	18.5	2
15 01 85 1500	1261.61	955.83	305.43	195,930	1,262-	0	36,620	36,620	66,420	0	66,920	1,527	18.0	2
15 01 85 1600	1261.53	955.84	305.74	194,470	1,460-	0	36,650	36,650	71,190	0	71,690	1,638	19.0	2
15 01 85 1700	1261.41	956.91	305.18	192,448	2,022-	0	36,620	36,620	84,650	0	85,150	1,944	23.0	2
15 01 85 1800	1261.29	957.52	304.19	190,413	2,034-	0	36,650	36,650	84,970	0	85,470	1,944	23.0	2
15 01 85 1900	1261.18	957.85	303.58	188,371	2,042-	0	36,650	36,650	85,160	0	85,660	1,944	23.0	2
15 01 85 2000	1261.07	957.72	303.33	186,491	1,880-	0	36,620	36,620	81,240	0	81,740	1,853	22.0	2
15 01 85 2100	1260.99	956.69	303.74	185,117	1,374-	0	36,650	36,650	69,130	0	69,630	1,580	19.0	2
15 01 85 2200	1260.96	954.43	305.17	184,555	562-	0	36,620	36,620	49,610	0	50,110	1,139	13.0	2
15 01 85 2300	1260.97	951.15	307.96	184,797	242	0	36,650	36,650	30,340	0	30,840	704	8.0	2
15 01 85 2400	1261.02	947.86	311.19	185,524	728	0	36,650	36,650	18,690	0	19,190	439	5.0	2
TOTALS					17,223-							29,513		
AVERAGES	1261.75	952.77	308.97	198,353		0	36,690	36,690	53,410	0	53,910	1,230	14.4	

Figure 6