U.S. Department
of Transportation
Federal Railroad Administration
Office of Research
and Development
Washington, D.C.
20590

Railroad Financial Evaluation Model: Description and Computer Program Users' Manual

Bolt Beranek and Newman Inc.
50 Moulton Street
Cambridge, MA 02238

FRA/ORD-81/25.II

August 1981

Final Report

J. A. Kane
C. E. Waldman

Document is available to the U.S. public through the National technical Information Service, Springfield, Virginia 22161

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
FRD/ORD-81/25.II		

4. Title and Subtitle	5. Report Date
Railroad Financial Evaluation Model: Description and Computer Program Users' Manual	August 1981
	6. Performing Organization Code
7. Author(s) J.A. Kane, C.E. Waldman	8. Performing Organization Report No. BBN Report No. 4632
9. Performing Organization Name and Address Bolt Beranek and Newman Inc. 10 Moulton Street Cambridge, MA 02238	10. Work Unit No. (TRAIS)
	11. Contract or Grant No. DOT-FR-8091
12. Sponsoring Agency Name and Address U.S. Department of Transportation Federal Railroad Administration Office of Research and Development Washington, DC 20590	13. Type of Report and Period Covered Final Report Sept. 1978-Feb. 1981
	14. Sponsoring Agency Code FRA/RRD-11

15. Supplementary Notes

16. Abstract

This report is part of a larger study to identify potentially cost-effective advanced braking and coupling systems and to prepare a plan for conducting the research and development needed to bring about implementation of these systems. This report describes a model for determining the economic feasibility of implementing advanced braking and coupling systems. First the financial analysis concept of a discounted cash flow is explained. That analysis technique is then adapted for a user-oriented computer model. Several example cases are then presented to demonstrate the actual operation of the model.

This study has also resulted in the following reports; "Methodology for Evaluating the Cost and Benefit of Advanced Braking and Coupling Systems" (FRA/ORD-79-57); "Evaluation of the Costs and Benefits of Advanced Braking and Coupling Systems" (FRA/ORD-80/49); "Recommendations for Research and Development on Advanced Braking and Coupling Systems" (FRA/ORD-81/24); "Railroad Yard Simulation Model: Description and Computer Program Users' Manual"(FRA/ORD-81/25.I).

17. Key Words	18. Distribution Statement
Discounted cash flow Braking and coupling systems Cost and benefit analysis Freight cars Computer Program Users' Manual Railroad financial analysis model	Document is available through the National Technical Information Service Springfield, VA 22161

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
UNCLASSIFIED	UNCLASSIFIED	97	

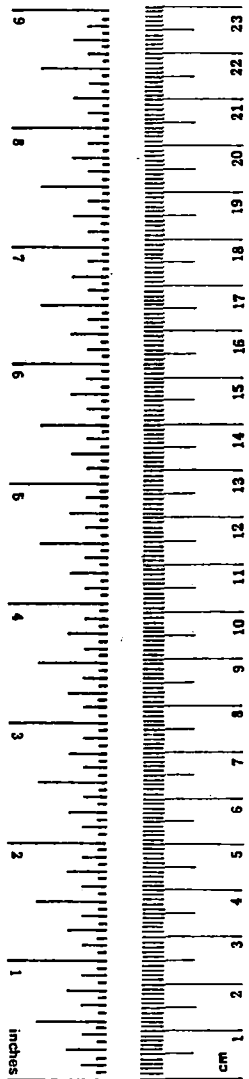
Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

METRIC CONVERSION FACTORS

Approximate Conversions to Metric Measures

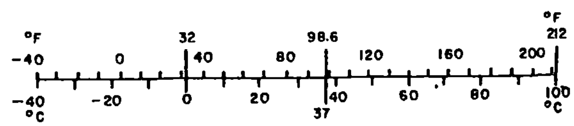
Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
in	inches	*2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
		AREA		
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
		MASS (weight)		
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2000 lb)	0.9	tonnes	t
		VOLUME		
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
		TEMPERATURE (exact)		
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C

*1 in = 2.54 (exactly). For other exact conversions and more detailed tables, see NBS Misc. Publ. 286, Units of Weights and Measures, Price $2.25, SD Catalog No. C13.10:286.



Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
		AREA		
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
		MASS (weight)		
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1000 kg)	1.1	short tons	
		VOLUME		
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	cubic meters	35	cubic feet	ft^3
m^3	cubic meters	1.3	cubic yards	yd^3
		TEMPERATURE (exact)		
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F



PREFACE

The authors express their appreciation to the people and organizations that have helped considerably throughout this project. The FRA COTRs, Ms. Marilynne Jacobs and subsequently Dr. N. Thomas Tsai, have provided invaluable guidance and direction. In addition, an industry committee composed of Messrs. Geoffrey Cope of Dresser Industries, John Punwani of the Association of American Railroads, Bruce Shute of the New York Air Brake Co., Donald Whitney of the Burlington Northern Railroad, and Carl Wright of Westinghouse Air Brake Co. have performed important review and consultation. The American railroad industry, in particular the Southern Railway, Boston and Maine, Conrail, and several other railroads, has graciously provided information and an opportunity to observe railroad operations.

The model described in this report was initially developed by Alan Berger. The original computer code was written by Jonathan Cohen and later revised by Carol Waldman. Deborah Melone edited the entire report.

TABLE OF CONTENTS

page

PREFACE..iii

LIST OF FIGURES AND TABLES.................................vi

SECTION 1. GENERAL OVERVIEW.................................1

2. DESCRIPTION OF THE FIMOD MODEL..........................5

2.1 Financial Analysis Concepts............................5

2.2 Elements of the Model..................................7

2.3 Processing Sequence and Procedures....................18

2.4 Program Structure.....................................27

3. PROGRAM OPERATION......................................34

3.1 Initiating the Program................................34

3.2 Example Cases...34

3.3 The DELETE Command....................................50

REFERENCES...53

APPENDIX A. LISTING OF FORTRAN PROGRAM....................A-1

LIST OF FIGURES AND TABLES

page

Figure 1. Method of analysis of braking and coupling systems....2

2. Example of financial analysis using method shown in Fig. 1....8

3. Diagram of input of adjustments and parameters to FIMOD model....10

4. Processing sequence flowchart....21

5. Flowchart of MAIN routine....28

6. Flowchart of LOOK subroutine....29

7. Flowchart of PRINT subroutine....33

8. Flowchart of GRAPH subroutine....33

Table 1. Summary of adjustments and structural parameters....11

2. Summary of program control....31

3. Summary of program input....32

4. Summary of OUTPUT subroutines....32

1. GENERAL OVERVIEW

This report describes FIMOD, an interactive computer model for assessing the financial feasibility of investments by U.S. railroads in advanced braking and coupling systems. The purpose of this report is to introduce prospective users of the model to both the conceptual basis and the actual operation of the FIMOD model. Before discussing the model itself, we first discuss the larger project of which this model was a part and the methodology with which it was integrated in that project. In subsequent sections we describe the FIMOD model in detail and present examples of its operation by a user.

The FIMOD model was developed as part of a project sponsored by the Federal Railroad Administration (FRA) on advanced braking and coupling systems. That project was intended to determine a future course for railroad braking and coupling systems research and development through analysis of the costs and benefits of 16 specific innovative systems. The results of the project are presented in a three-volume series of reports [1,2,3].

The method of analysis used in the project is presented in Fig. 1. Briefly, analysts identify a candidate system and conceptualize the hardware for that system. Once the conceptual design of the system is established, they make an equipment cost analysis to estimate both the equipment cost per car and any associated changes in maintenance costs. The benefits of the candidate system - reductions in operating costs and reductions in accidents and maintenance costs - are estimated using operational and dynamic evaluation models. The output of those models, together with changes in maintenance costs from the equipment analysis, are primary inputs to the financial analysis model, FIMOD. The output of FIMOD is the maximum allowable cost per freight car that is financially feasible under conditions specified by the user. If allowable costs are greater than the

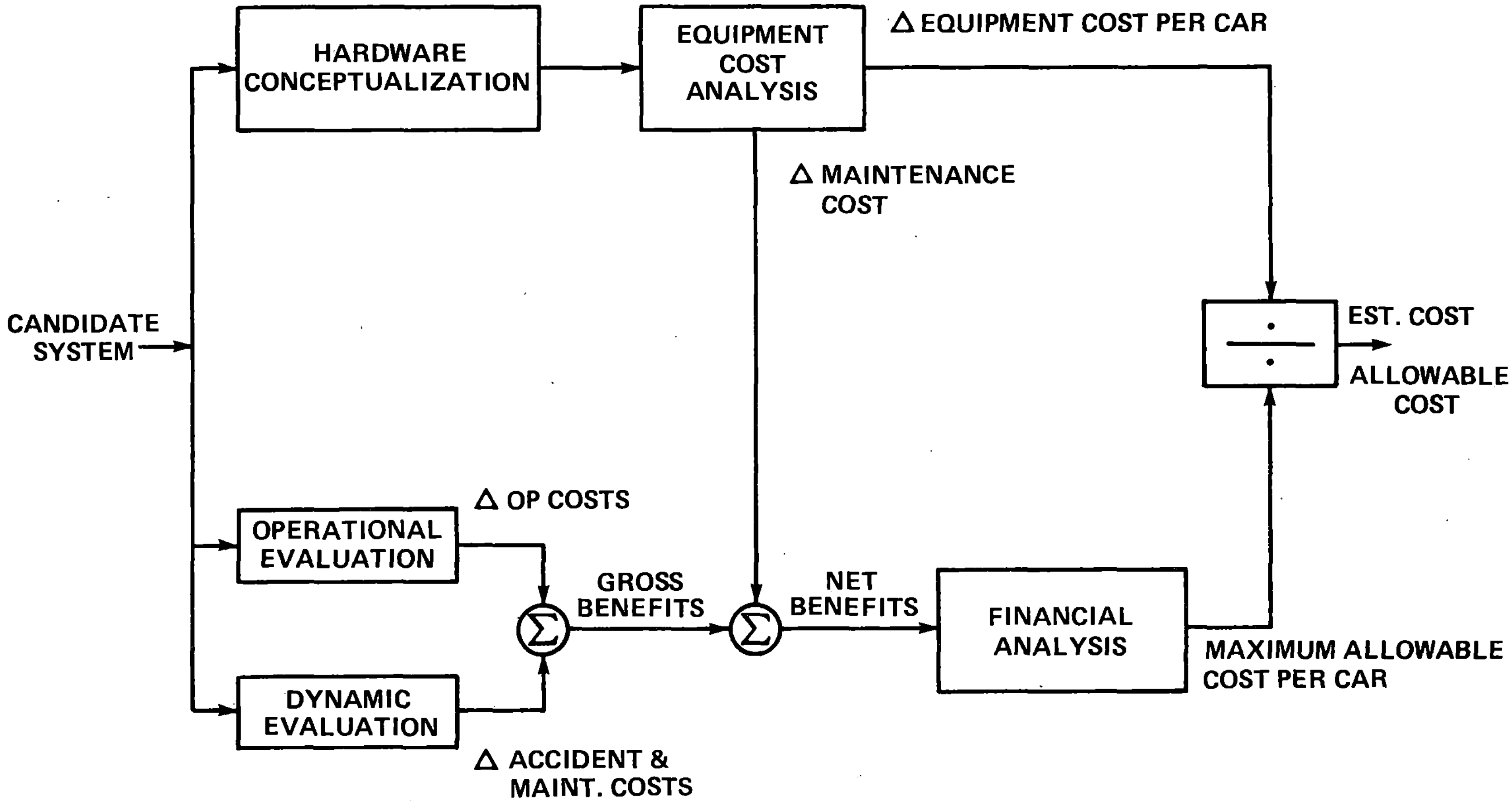


FIG. 1. METHOD OF ANALYSIS OF BRAKING AND COUPLING SYSTEMS.

estimated cost per car, the candidate system is financially attractive. If they are less than the estimated cost per car, the investment is not worthwhile.

This overview of a relatively complex project is necessarily brief and general. Each element in the overall method represents a significant amount of analysis. For example, the operational evaluation is performed in part with the computer model RAIL, which is described and documented in a companion users' manual [4]. The financial model, described in this report, takes account of a wide variety of tax considerations, labor relations, policies, implementation strategies, and discount rates, all of which affect financial feasibility.

In its present configuration, the FIMOD model computes the allowable costs for the entire U.S. railroad industry. The model is constructed this way chiefly because the types of advanced systems under consideration require compatibility among all cars in interchange service. The model could be modified, however, to reflect more accurately the operations of a specific service.

The user's manual in this report follows the recommended format for computer program users' manuals shown in Federal Information Processing Standards Publication No. 38, "Guidelines for Documentation of Computer Programs and Automated Data Systems." In addition to those requirements, Sec. 2 includes an expanded section that describes the model. Section 3, Program Operation, presents several example cases demonstrating how to initiate the program, what types of commands can be made, and what the output looks like.

User Profile/Operational Environment

The financial evaluation model, FIMOD, is an interactive Fortran computer program developed to run on the computer system maintained by the BBN Research Computer Center (RCC). The program is available through BBN on a time-shared system that uses Digital Equipment Corporation computers. Alternatively, tapes of the program can be obtained from BBN or NTIS and made operational on other computer systems.

After accessing the model through a keyboard computer terminal, users are prompted by instructions from the program. For example, typing the word HELP (followed by pressing the return key) will produce a brief explanation of the program, including a description of the commands that it will accept. The program is written in the Fortran computer language, but users do not need to know Fortran unless they want to make changes to the model. To run the program, users simply type brief commands on the terminal. Little typing skill is required to use the FIMOD program, because all the commands and other inputs to the program are kept short.

To use the model properly, however, users must become familiar with the basic concepts of financial evaluation, as described in Sec. 2. For example, they should understand the implication of specifying one or another depreciation method. The model prompts the user to choose the method he wants, and this choice affects the results obtained from the analysis.

2. DESCRIPTION OF THE FIMOD MODEL

This section is intended to provide users of the FIMOD model with sufficient information to understand the conceptual basis of financial analysis and how it is adapted to evaluating advanced braking and coupling systems.

2.1 Financial Analysis Concepts

The basic issue in financial analysis is whether an investment is worthwhile. Making that determination requires comparison of the costs of the investment with the benefits it is expected to generate. This simple comparison of costs and benefits is complicated by the fact that the costs and benefits occur over time. A dollar of benefits next year does not exactly offset a dollar of costs today. Three elements cause future dollars to be worth less than today's dollars: inflation, risk, and the time preference for money, which is the natural preference for having money and its benefits sooner rather than later. The value of future dollars, whether benefits or costs, is reduced by discounting them to their "present value." Discounting future costs and benefits to their present value enables one to compare costs and benefits on a consistent basis - i.e., their value today.

Net Present Value (NPV) is the conceptual basis of the FIMOD model. NPV analysis is the method by which costs and benefits occurring over time are discounted to their present value. If the discounted value of all the benefits is greater than the discounted value of all the costs, the net present value is positive. The formula for NPV is:

$$NPV = \sum_{t=0}^{N} \frac{B_t - C_t}{(1 + r)^t} , \qquad (1)$$

where

NPV = the net present value of investment project,

B_t = the benefits generated by the project in year "t",

C_t = the costs of the project in year "t",

r = the appropriate discount rate or "cost of capital."*

Consider a simple example. Assume that an investment today (t = 0) of $1000 with annual maintenance costs of $100 would generate annual benefits of $600. Further assume that the discount rate is 20% and the investment life is 3 years, at which time the investment would be worthless. The NPV would be:

$$-1000 + \frac{(600 - 100)}{1.2} + \frac{(600 - 100)}{1.2^2} + \frac{(600 - 100)}{1.2^3}$$

NPV = -1000 + 1053.24 = $53.24.

The positive cash flows (benefits) more than offset the negative cash flows, and hence the net present value is positive.

A crucial consideration in this type of analysis is the rate at which future cash flows are discounted; that is, the discount rate. This discount rate is not merely the anticipated rate of inflation. Rather, it also reflects the degree of risk and the return that could be realized by a firm from other investments. Companies often deal with this decision by having an internal rate of return (IRR) that is set as a standard for project acceptance.† The FIMOD model allows the user to vary the

---

*In the FIMOD model, the discount rate reflects the return a railroad must earn on a given project in order to generate funds from investors.

†The internal rate of return is formally defined as follows: when NPV is set equal to zero and the equation (1) is solved for r, r is called the internal rate of return. In the above example, IRR = 23.38%.

discount rate to reflect the appropriate internal rate of return. An IRR of 20% was determined to be appropriate for evaluating railroad industry investments in 1979 [2].

2.2 Elements of the Model

Given this brief overview of financial analysis, and referring to Fig. 1, the user can begin to see how the FIMOD model operates. The output of the Equipment Cost Analysis component is the initial investment cost per car of a new technology. In financial analysis terms, it is the cash flow representing initial outlay for the investment. All the subsequent costs and benefits resulting from that investment are evaluated in the Financial Analysis component. The output of the Financial Analysis component is an estimate of the maximum allowable cost per car that would be justified on the basis of the net benefits to be realized from the investment, discounted over the life of the investment. By comparing the increase in equipment costs per car to the maximum allowable cost per car, one can determine the feasibility of the investment. If maximum allowable cost is greater than the estimated equipment cost, the investment would be financially feasible, for the discounted net benefits would more than offset the initial cost of the equipment.

Figure 2 depicts the simple numerical example presented in Sec. 2.1 as it would be adapted to the overall methodology of Fig. 1. The $1000 cost per car is less than the $1053 net present value of the gross benefits less increased maintenance costs that would be realized over a three-year life for the investment. The investment is therefore justified.

This example is intended to convey the simple mechanics of evaluating investments using the FIMOD model. The actual process is somewhat more complicated, because additional considerations,

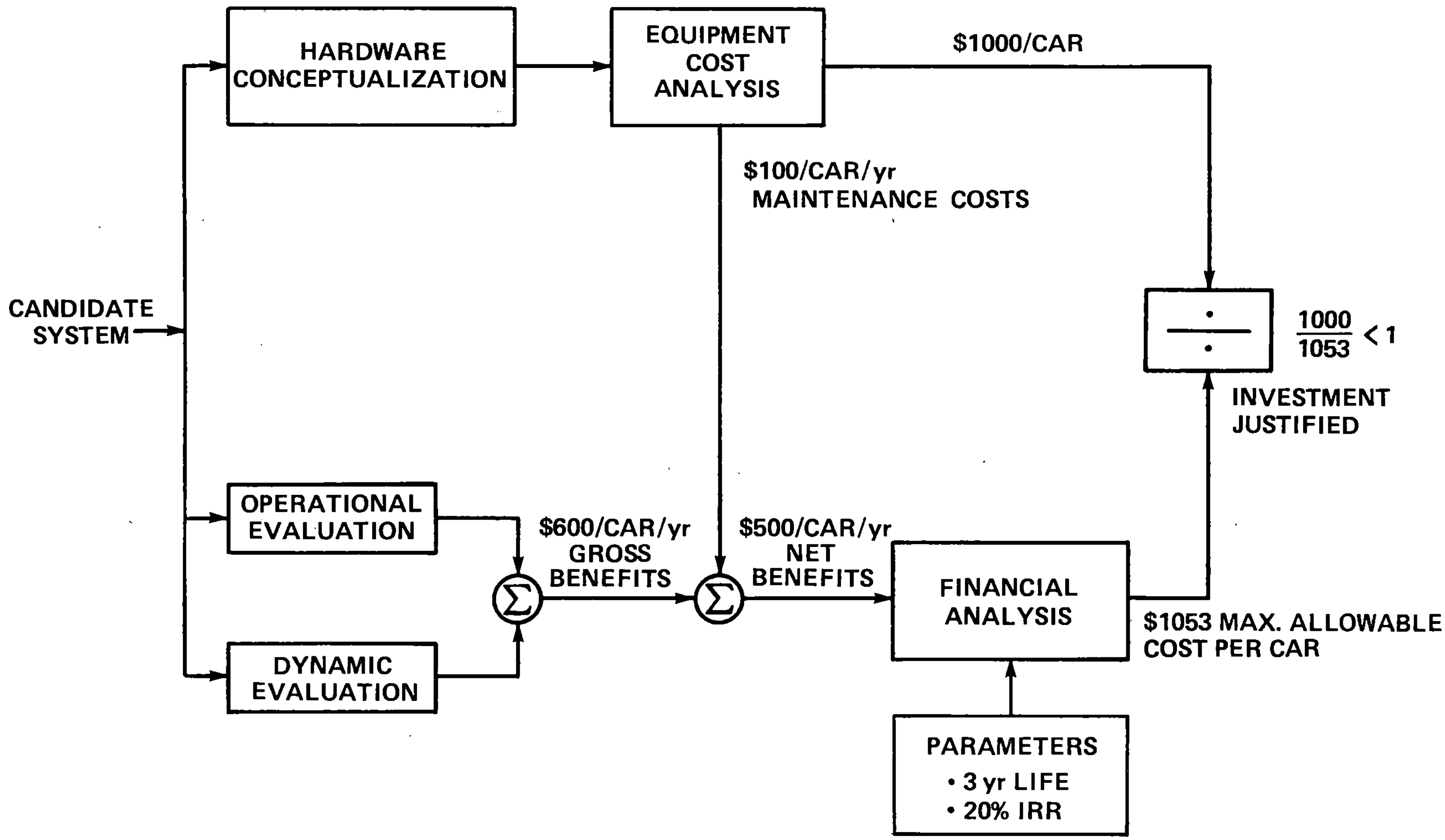


FIG. 2. EXAMPLE OF FINANCIAL ANALYSIS USING METHOD SHOWN IN FIG. 1.

such as tax implications, labor relations, and equipment depreciation, must enter the analysis. The remainder of this section discusses all these additional factors and how they enter the FIMOD model.

There are two additional sets of inputs that must enter the financial analysis:

- Adjustment to benefits
- Specification of structural parameters.

The manner in which inputs enter the financial analysis is depicted in Fig. 3.

The user specifies the adjustments to be made to the costs and benefits to be realized. By doing so, the user determines <u>how</u> benefits are to be realized. For example, by specifying an accelerated depreciation schedule rather than a simple straight-line method, the user can gain additional tax benefits in the early years of the equipment.

Structural parameters specified by the user determine <u>when</u> benefits are realized. These structural parameters largely determine the time dimensions of the analysis and how the investment is to be implemented over time. For example, the user must specify the number of years over which the next present value of costs and benefits are to be calculated, i.e., the maximum value of "t" in Eq. 1. The specific adjustments and structural parameters to be specified by the user are identified in Table 1. Each element and its significance are discussed in the text following the table.

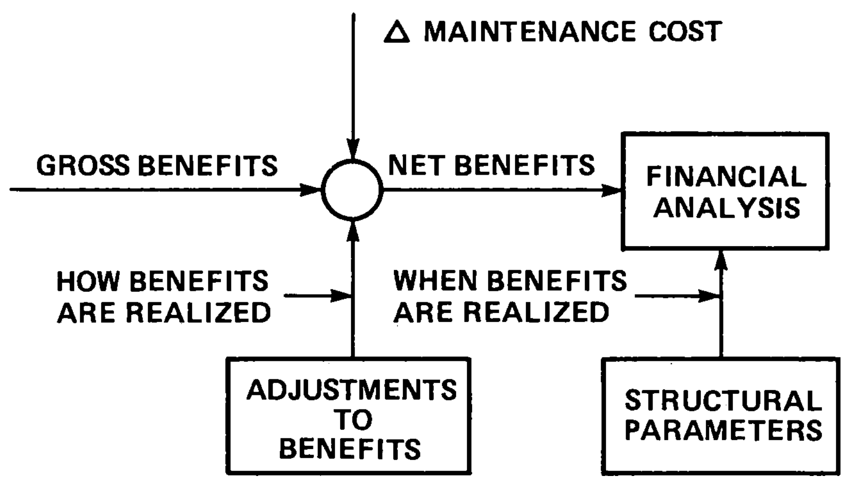


FIG. 3. DIAGRAM OF INPUT OF ADJUSTMENTS AND PARAMETERS TO FIMOD MODEL.

TABLE 1. SUMMARY OF ADJUSTMENTS AND STRUCTURAL PARAMETERS.

Adjustments to Benefits	Structural Parameters
• Inflation rates for materials, labor costs, and other savings	• Number of cars in system
• Fraction of labor savings paid to union	• Years to system compatibility
• Number of years of union payout	• Years cash flows to be calculated
• Depreciation method	• Depreciation lifetime of the technology
• Fraction of investment allowable for investment tax credit	• Physical lifetime of freight cars
• Federal tax rate	• Fraction of retrofit cost required per new car production
	• Internal rate of return

Inflation rates for materials, labor costs, and other savings. If the time frame of the financial analysis is 20 years, one simply cannot assume that the wages of yard employees and the prices of the component parts will remain constant over the 20-year period -- or even over a 2-year period. The user must therefore specify inflation rates for labor, materials, and other savings. "Other" savings represent benefits such as improved utilization of equipment or less frequent replacement of existing components.

Note that this specification does not offset the effect of discounting future costs and benefits, since the inflation rate specified by the user will be less than the discount rate used in the analysis. Recall that the discount rate accounts not only for inflation but also for risk and the attractiveness of other investments. Therefore, if an investment is expected to reduce labor costs by $100 per year, inflation is assumed to be 10% per year, and the discount rate is 25%, the net present value of that savings over 3 years would be

$$\frac{100 * 1.1}{1.25} + \frac{100 * 1.1^2}{1.25^2} + \frac{100 * 1.1^3}{1.25^3} = \$233.59$$

Fraction of labor savings paid to union. Introduction of labor-saving devices and practices in the railroad industry has often been accompanied by compensation to unions for the loss of income by members whose jobs were eliminated. The FIMOD model explicitly permits the user to specify the amount and duration of such compensation. The amount of compensation is specified as a percentage of total labor savings attributable to the innovation. The duration is specified as the number of years over which such compensation is to be paid.

If the innovation requires several years to take effect, and no labor savings are realized until the implementation period is completed, the model does not begin to make this payout until the savings begin. For example, if the user specifies that the payoff period is 10 years, but the innovation takes 5 years to implement before any benefits are realized, the payoff is made in years 6 to 16, not in years 1 to 10. This is referred to as a Type I payout. However, if the innovation begins to produce benefits as soon as it is installed, and systemwide benefits increase as more and more cars have the innovation installed, the model computes the union payout starting in year 1, even though only a portion of the fleet has the new technology. This is referred to as a Type II payout.

Depreciation Method. The user may choose one of three depreciation methods:

- straight line
- double declining balance
- sum-of-years digit.

Straight-line depreciation simply allocates equal depreciation charges over the depreciation life of assets. Hence, a $1000 asset depreciated on a straight-line basis over 10 years would be written off at $100/yr. The other two methods permit accelerated depreciation. These methods, which allow larger depreciation charges in the early years of the asset, yield tax advantages. The specific formulae for double-declining-balance and sum-of-digits methods can be found in most elementary accounting textbooks. Users interested in using these methods should consult an accounting textbook for the advantages and disadvantages of each method.

Note that once a user chooses a depreciation method, he cannot change the technique during that run. The user can choose only one method at a time and must use only that method throughout that run of the analysis. After the run is complete, when FIMOD prompts the user, he can use the CHANGE command to choose another method for another run of the analysis.

Investment tax credit. An investment tax credit (ITC) can be realized from investment in new equipment. The ITC is specified as a percent of the investment made. The IRS code specifies the allowable tax credit that can be taken for investments of different lifetimes. Ten percent of the initial investment outlay would be a typical rate for investments in advanced braking and coupling systems. Hence a $1000 investment would produce a $100 tax credit, which would be applied in the first year after the investment.

Federal tax rate. In adjusting gross benefits to net benefits, consideration of the after-tax advantages of an investment is probably the largest single factor. If an investment reduces costs while revenues stay constant, profits will increase. The tax rate applied to those profits will determine the after-tax benefit of the investment. The higher the tax rate, the lower the after-tax profits, and thus the lower the after-tax benefits. A tax rate of 46% is typical for the U.S. railroad industry [2].

Example of Adjustments. The following example shows in accounting fashion the types of adjustments that would be made in the FIMOD model. Note that these examples are shown in an accounting format. The FIMOD model actually uses a different computational procedure to facilitate computer calculations, but the underlying acounting logic is equivalent.

Base Case

Revenues (R)		1,000,000
Operating Costs (OC)		700,000
Taxes:		
Revenue Less Operating Costs (R-OC)	300,000	
Less Depreciation	100,000	
Taxable Income (TI)	200,000	
Tax (50% of TI)		100,000
Net Cash Flow (R-OC-tax)		200,000

Post-Investment Case

Given: An investment of $500,000 to be depreciated on a straight-line basis for 10 years. The investment will reduce labor costs by $100,000 per year with a zero inflation rate and have a 10% investment tax credit.

	Year 1		Year 2 - 10	
Revenues (R)		1,000,000		1,000,000
Operating Costs (OC)		600,000		600,000
Taxes:				
Revenue Less Operating Cost (R-OC)	400,000		400,000	
Less Depreciation	150,000		150,000	
Taxable Income (TI)	250,000		250,000	
Tax (46% of TI)	115,000		115,000	
Less Investment Tax Credit (ITC) (10% of Investment)	50,000		0	
Total Tax (TAX-ITC)		65,000		115,000
Net Cash Flow (R-OC-TOTAL TAX)		335,000		285,000

Annual increase in after-tax profit

year 1 $335,000 - 200,000 = $135,000

years 2-10 $285,000 - 200,000 = $85,000

Note that the FIMOD model would then discount these increases in after-tax profits to obtain maximum allowable investment. Further note that a zero inflation rate is assumed in this analysis and that no union payout is made.

Structural parameters control the time dimensions of the financial analysis. Each of the parameters identified in Table 1 is discussed below.

Number of cars in the system. Since the analysis is on a per-car basis, the user must specify the number of cars in the U.S. rail system. The analysis in Ref. 2 is based on a fleet of 1,444,000 freight cars.

Years to system compatibility. Some type of advanced braking and coupling systems are compatible with existing systems. For example, knuckle-open coupler devices would be compatible with existing couplers, and the benefits would accrue immediately. In contrast, other advanced systems, such as those requiring electrical power, would not be compatible until the entire fleet had the capacity for electrical connections. Hence 10 or more years might be required until the new advanced system was compatible and the benefits could be realized. Note that in actually using the model, the convention is that if it takes 10 years for compatibility, the system is compatible in year 10, and there were 9 years before the system was compatible.

Years cash flows to be calculated. The user must specify the number of years over which costs and benefits are to be discounted; that is, the maximum value of "t" in Eq. 1. Since some advanced systems might require 10 or even more years to implement, the user would probably want to specify at least 20 years for this parameter. Note that as the discount rate increases, the present value of net benefits in later years of analysis decreases.

Depreciation lifetime of the technology. This parameter sets the number of years over which the new system is to be depreciated for tax purposes. It is the depreciation lifetime of the technology as distinguished from its physical life. In most instances the depreciation lifetime of the technology will be the same as the physical lifetime. In those instances where there is a difference, depreciation lifetime will be shorter, in order to take advantage of the depreciation tax credits that would be realized from the rapid write-off of the asset.

Physical lifetime of the technology. This parameter specifies the lifetime of the technology. It is used to calculate the attrition rate of technology. The significance of this parameter is that it determines when new technology must be replaced during the life of a car. The technology lifetime will often be less than the lifetime of a freight car; therefore, there will be a cost associated with the replacement of the technology over the life of the car.

Physical lifetime of freight cars. This parameter is used to calculate the attrition rate of freight cars, which in turn controls the flow of new cars with the advanced systems into the fleet. For example, if the user specifies that the lifetime of cars is 30 years, the model will assume that 1/30 or 3.33% of the fleet will be retired each year and replaced with new cars.

Fraction of retrofit cost required for new production. It is generally less expensive to incorporate a new system into a freight car while it is being manufactured than to retrofit the car while it is in service. The user can account for this explicitly by specifying the savings that could be realized by incorporating the new system in car production. Hence, if the user specifies the value of this parameter as 50%, the model will assume that a $1000 retrofit cost would be only $500 for a new car.

Internal Rate of Return. The user specifies the IRR as the rate to discount each year of cash flows. The choice of the discount rate should reflect the target rate of return for the company (i.e., the company expects to make 25%, since it is borrowing at 15%). Projects with relatively greater risk should have higher discount rates. A range of 5 to 25 percent internal rate of return is incorporated in the model, so the user does not have to specify a particular IRR.

Example: The parameters specified by the user control the timing of the analysis and, in particular, the actual introduction of the new system into the fleet through retrofitting and new cars. The following example shows how this would be accomplished.

Parameters:

- number of cars 1000
- years before compatibility 10
- lifetime of the technology 10
- lifetime of the cars 20

Implementation:

Year	New Replacements	Retrofits	Cars With	Cars Without
1	50	50	100	900
2	50	50	200	800
.				
.				
.				
10	50	50	1000	000

Note in this example that 50 "new technology" cars per year enter the fleet, based on the attrition rate of 1/20, or 5% of 1000. Since this would only change one half of the fleet in a 10-year period, the other 500 cars must be retrofitted. The number of cars retrofitted each year is prorated at the years to compliance, 500 cars/10 years = 50 cars/year. The system is compatible in year 11.

2.3 Processing Sequence and Procedures

This section presents the actual processing sequence and procedures of the FIMOD model. It describes how the concepts of financial analysis and specific adjustments to the analysis are incorporated in the principal subroutines of the model. The purpose of this section is to inform the user of exactly how the model processes the input information specified by the user.

The FIMOD model calculates benefits and costs (positive and negative cash flows) adjusted for taxes and inflation over the period of analysis. It then discounts those costs over a range of internal rates of return. The actual computations are not difficult, but they are complicated, because there is a great deal of information to keep track of. The advantages of using a computer to make the numerous calculations and keep track of all the information are obvious.

The actual computations made by the FIMOD model and their sequence are designed to take fullest advantage of the computer's processing and storage capabilities. The sequence, algorithms, and computer code might be awkward for hand calculations but are in fact very efficient for the computer. While using the FIMOD model, users never see the computations themselves but only the output they generate. However, it is important for users to know how the output is generated so that they can be sure that the model is actually computing what they want.

The FIMOD model organizes processing in an eight-step sequence shown in Fig. 4. The steps are:

1. Calculate the cost of retrofitting cars with advanced system
2. Calculate cost of installing advanced system on new cars
3. Calculate investment tax credit
4. Calculate depreciation tax credit
5. Calculate tax-adjusted labor savings
6. Calculate tax-adjusted other savings
7. Sum cash flows
8. Solve for maximum allowable cost.

The financial model consists of a set of equations organized in subroutines. For each year the start-up costs (if any), annual costs, investment tax credit, and depreciation tax credit

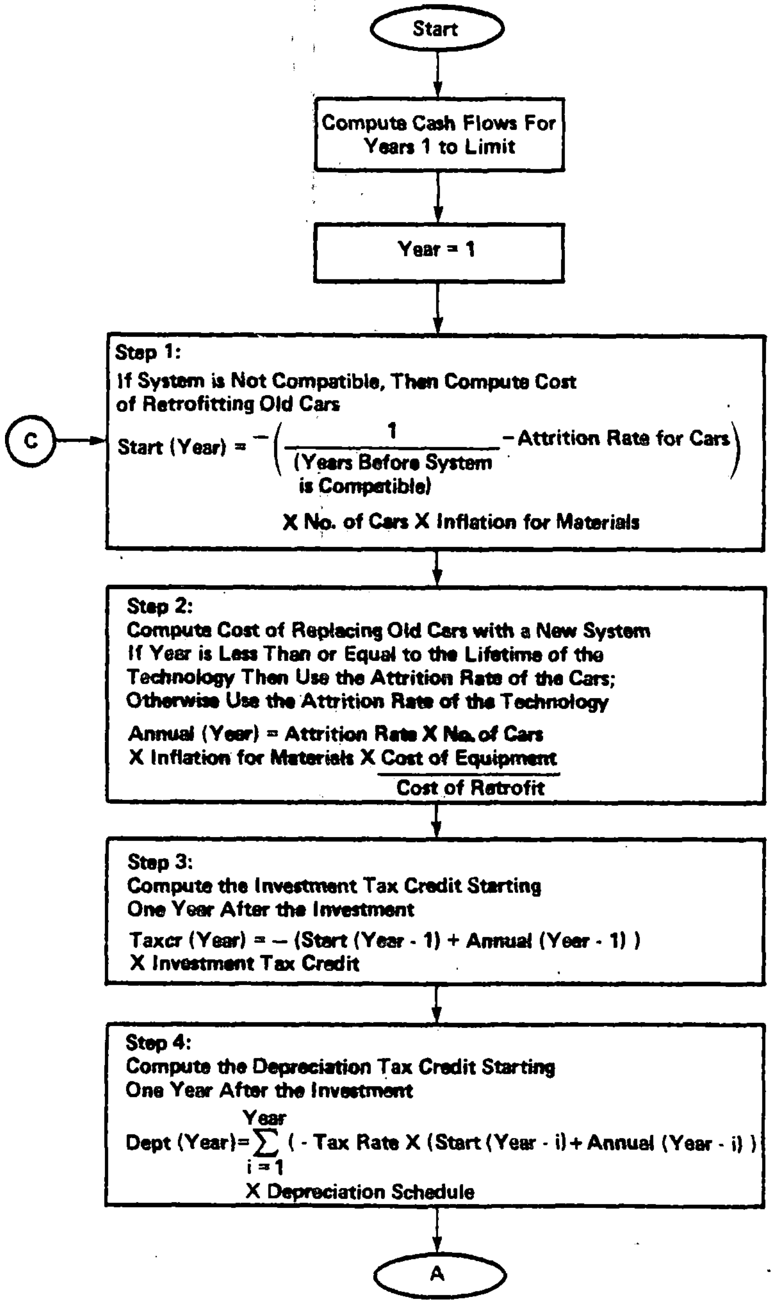


FIG. 4. PROCESSING SEQUENCE FLOWCHART.

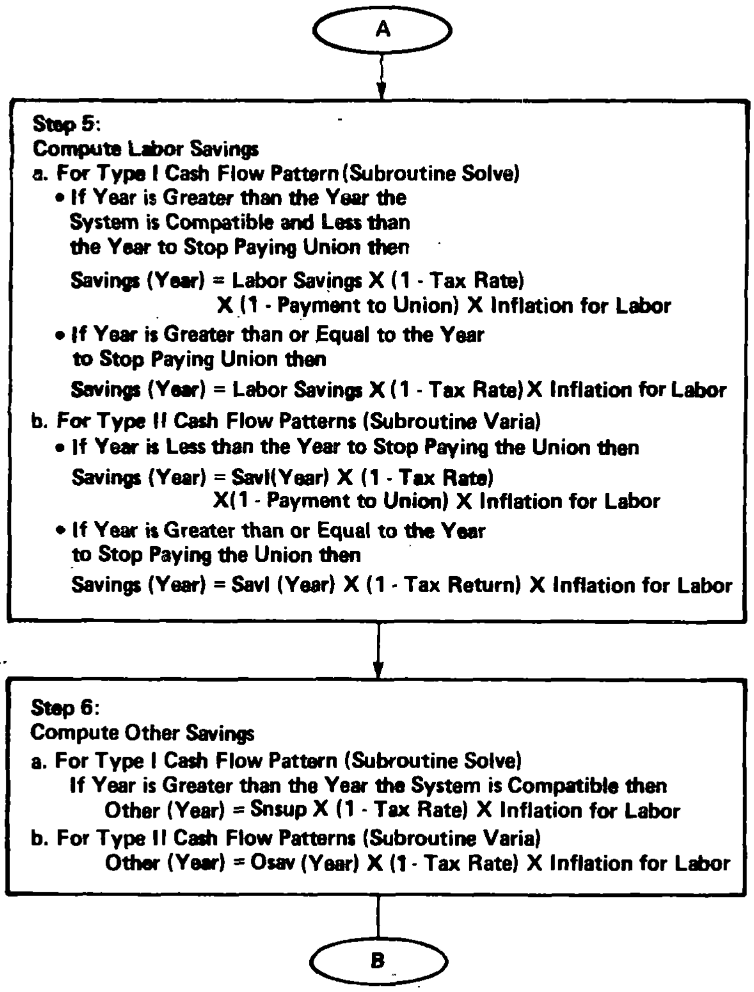


FIG. 4 (Cont.) PROCESSING SEQUENCE FLOWCHART.

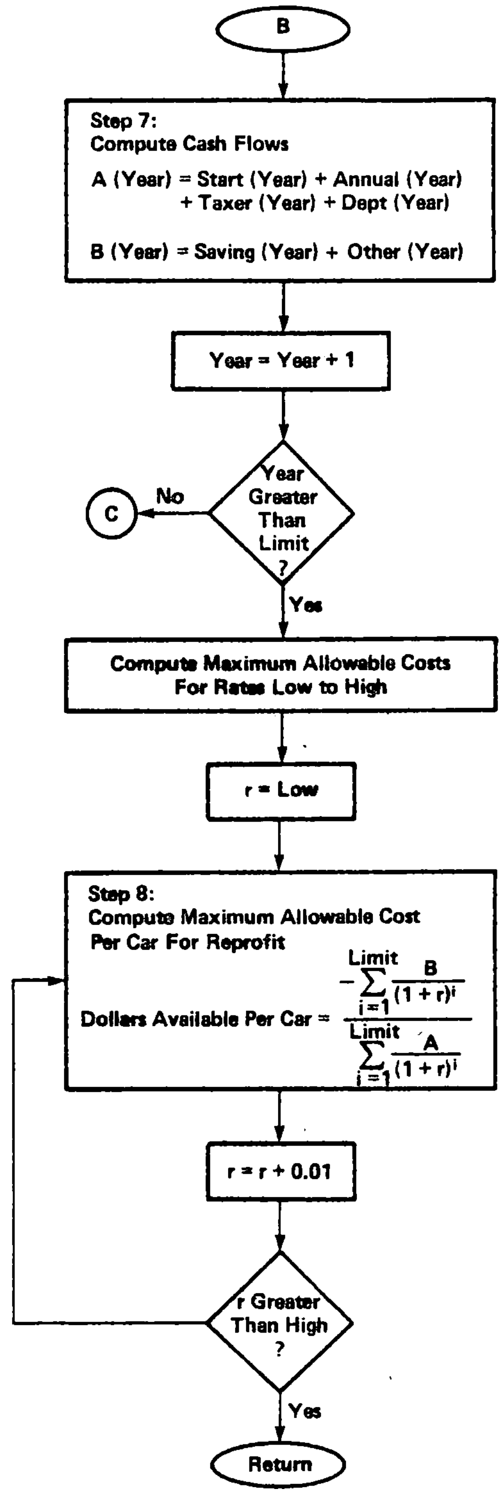


FIG. 4 (Cont.) PROCESSING SEQUENCE FLOWCHART.

are computed. Labor and other savings are also computed. All the costs are proportional to the cost of retrofitting a single car. The dollar amount available per car is obtained in the final step by dividing the sum of all the savings by the sum of all the costs. The discussion below highlights each step in this process. In reviewing these steps, the reader should keep in mind the following considerations:

- All costs are relative to the cost of retrofitting a freight car
- The attrition rate of cars is separate and distinct from the attrition rate of technology
- Many calculations embody a time lag because benefits in time "t" are the result of actions in time "t-1."

Step 1 - Calculate Costs of Retrofitting. The program calculates the cost of retrofitting cars for each year before the system is compatible. This cost is referred to as START. If it requires 10 years for the system to be compatible, the cost of retrofitting is calculated for years 1 to 10. The program automatically accounts for how many cars would be retrofitted each year on the basis of the attrition rate of cars, the number of cars in the fleet, and the number of years required for system compatibility. In this first step, the program calculates the cost of retrofitting and the number of cars to be retrofitted for a given year. It is assumed that retrofit costs are only incurred from the beginning of the implementation period to the year before system compatibility.

Step 2 - Calculate Costs for New Cars. The marginal cost of installing advanced braking and coupling systems on new cars is calculated in this second step. The marginal cost (ANNUAL) is determined by user input specifying new car costs as a percent of retrofit. In general, the marginal cost of installing the technology on a new car will be less than retrofitting. The number of new cars and the cost of the technology on those cars is a function of the attrition rates. Specifically, if the current year, t, is less than or equal to the lifetime of the technology, then the attrition rate of the technology is used.

In Steps 1 and 2, the program calculates the costs of the advanced system for a given year. Over the period of analysis, these are the gross negative cash flows associated with the system. In the next series of steps, the program focuses on the benefits - that is, positive cash flows generated by the system, first by reducing the effective cost through tax savings and then by accounting for other direct savings.

Step 3 - Calculate Investment Tax Credit. The program checks to determine if in the previous year expenditures were made on retrofitting existing cars or installing technology on new cars. The program sums these two costs (START and ANNUAL respectively) and then multiplies that sum by an investment tax credit rate specified by the user.

Step 4 - Calculate Depreciation Tax Credit. A firm investing in advanced systems would realize tax benefits attributable to the depreciation of the equipment. The benefits in the current year, DEPRT, are equal to the sum of investments in retrofitting old cars and installing the system on new cars times the proportion of that investment to be depreciated this year times the tax rate. Note that depreciation benefits in this year are related not solely to investments last year but also to investments in previous years and to the time schedule over which they are being written off. That schedule is determined by the depreciation method chosen and the lifetime of the asset for tax purposes, both of which are specified by the user.

At the end of this step, the program has calculated all the costs and benefits associated with the freight cars themselves. In the first two steps the program calculated the cost of implementing the system on cars, while in the third and fourth steps the program calculated offsetting tax benefits, specifically investment and depreciation tax credits, directly based on the costs calculated in Steps 1 and 2. In the following steps, the program focuses on labor savings and other benefits.

Step 5 - Calculate Labor Savings. To calculate labor savings there are three considerations:

- Benefits subject and not subject to union payout
- When benefits begin to be calculated
- How benefits are actually calculated.

Benefits subject to union payout are determined by the user, who specifies the proportion of labor savings paid to the union and the number of years the payments are to be made. If the user specifies a union payout, there are two ways in which the payout can be made.

The program checks to determine if the entire system must be compatible before benefits are realized - Type I payout schedule - or if benefits are realized as soon as implementation begins - Type II payout schedule. The program automatically assumes a Type I cash flow if the user does not specify either variable labor savings, or variable other savings. However if the user does enter variable savings, the program automatically assumes a Type II cash flow, and union payment will begin in year 1. In the Type I case, labor savings do not begin until the system is compatible. The union payout would begin in the first year of compatibility and continue for the number of years specified by the user. Hence, if it took 10 years for compatibility, and union payout was for 10 years, the program would start to calculate in year 11, the first year of compatibility, and would continue for 10 years. If, on the other hand, the benefits begin immediately with implementation, the program begins to calculate in year 1. Since benefits are proportional to the number of cars that have the technology, total benefits increase over the implementation period. Returning to the example of a 10-year union payout and a 10-year period until the system was installed on the entire fleet, the model would calculate one-tenth of labor savings in year 1, two tenths in year 2, etc. Thus, the actual labor saving and the amount paid to the union would increase over the implementation period. In this example, payments would cease at the end of year 10.

The program calculates the net benefits by first determining the labor savings for current year, given the considerations described above. Those savings are then multiplied by: (1 - tax rate) times (1 - proportion to union payout) times the inflation rate for labor. The product of that calculation is the after-tax, after-payout labor saving benefits for each year of the analysis.

Step 6 - Calculate Other Savings. The benefits of other savings, such as reduced maintenance and accident-related costs, are calculated in this step. In effect, all savings not subject to union payout are accounted for. An adjustment is made for both taxes and inflation.

Step 7 - Sum Cash Flows. At this point the program has calculated as many as six cash flows for every year of the analysis. In Step 6, those cash flows are reduced to two. The sum of Steps 1, 2, 3, and 4 is calculated as A. Steps 5 and 6 total B. These sums, A and B, become inputs to Step 8.

Step 8 - Calculate Maximum Cost Per Car. In this final step, the present values of A and B are calculated for each year of the analysis. The yearly present values are summed to SUMA and SUMB. The negative of the total of SUMB is divided by SUMA, and the result is maximum allowable investment per freight car. This process is repeated over the range of internal rates of return specified by the user. The overall output is a series of maximum allowable expenditures for alternative rates of return.

2.4 Program Structure

Previous sections have described how the FIMOD model calculates allowable investment. That calculation is part of a much larger computer program. This section describes the other major components of the computer model of which the financial evaluation component is a part.

The FIMOD program is organized into four principal components. Each component has one or more subroutines to execute the objectives of that component. The components are:

- Program control
- Input
- Computation
- Output.

The computation component calculates allowable investment. It incorporates two subroutines, SOLVE and VARI. Section 2.3 has presented a description of computations performed by the computer model. This section, 2.4, describes how the FIMOD program executes the control, input, and output components.

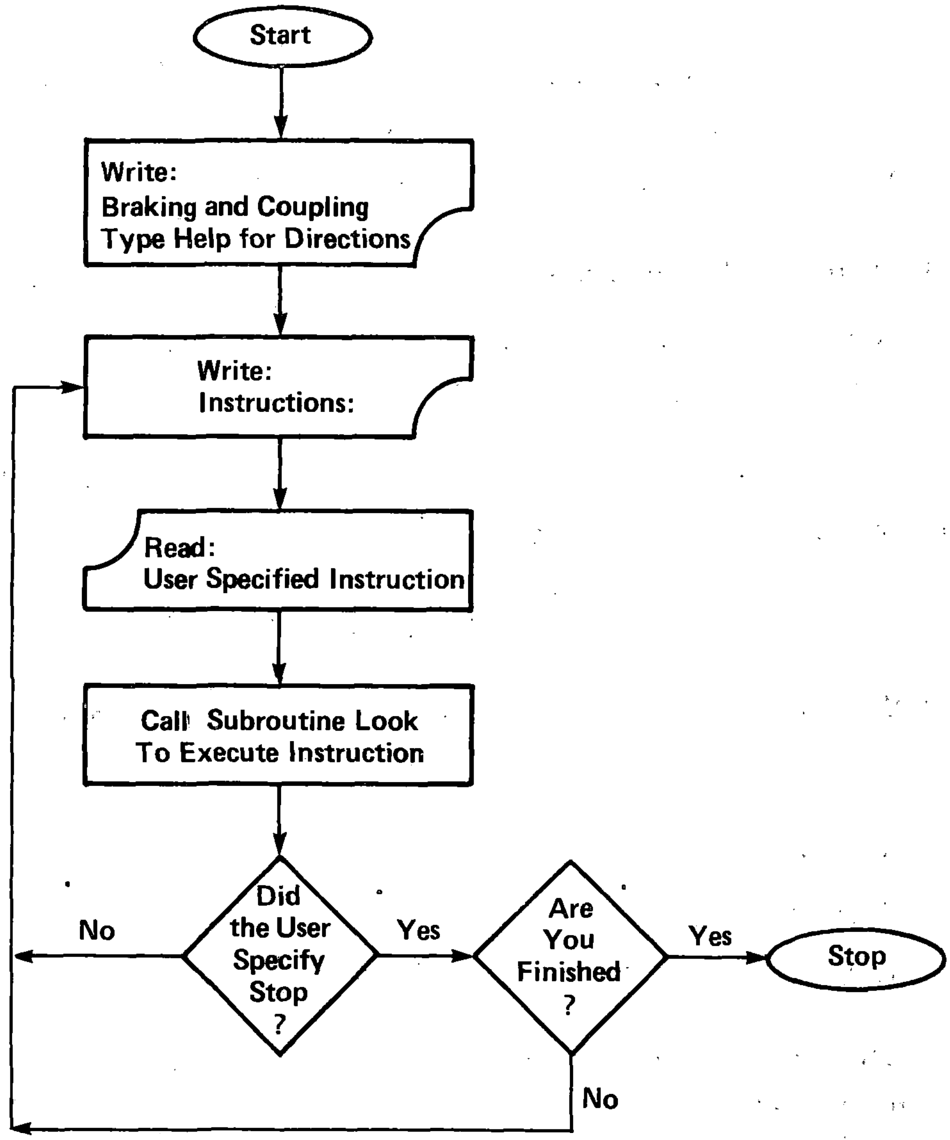


FIG. 5. FLOW CHART OF MAIN ROUTINE.

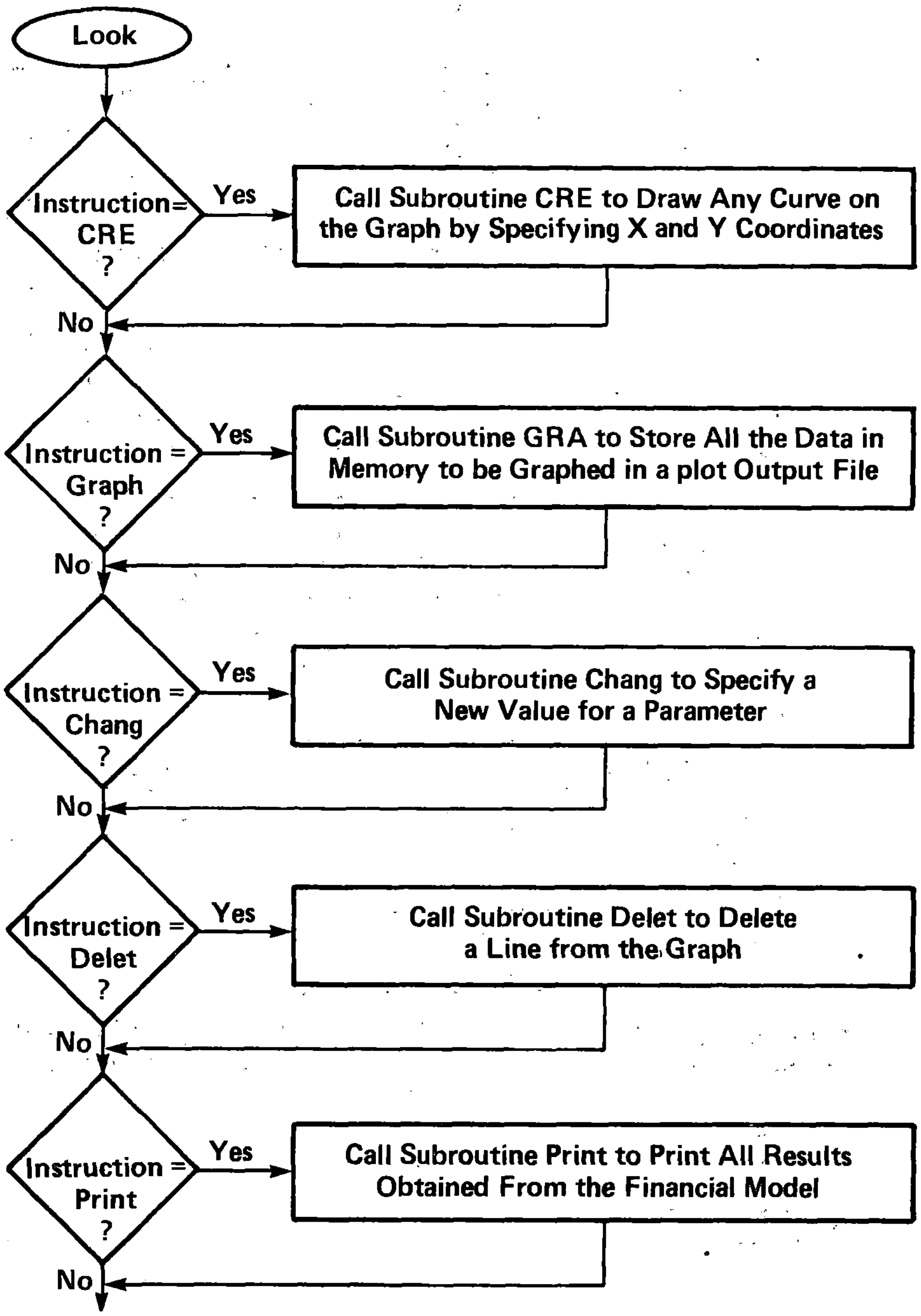


FIG. 6. FLOWCHART OF LOOK SUBROUTINE.

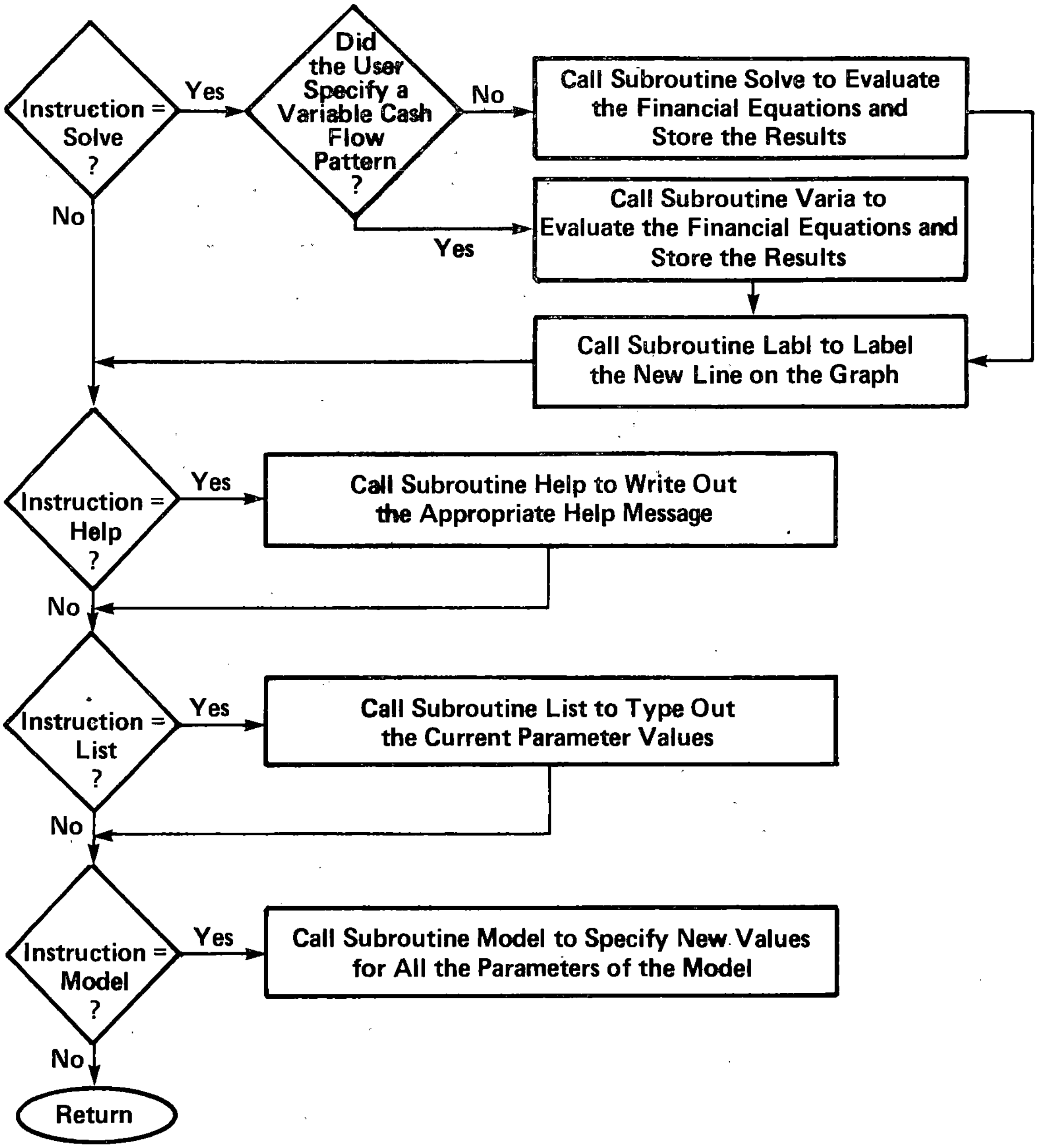


FIG. 6 (Cont.) FLOWCHART OF LOOK SUBROUTINE.

2.4.1 Program control

The overall control of the program is accomplished by the main program, MAIN, and by the subroutine LOOK. The main program is the primary user interface. It introduces the model, prompts for an instruction, and responds to the user's selections by calling the LOOK subroutine. The LOOK subroutine in turn calls the subroutine that performs the function specified by the user - such as PRINT, SOLVE, or HELP. Flowcharts for MAIN and LOOK are presented in Figs. 5 and 6. The control component is summarized in Table 2.

TABLE 2. SUMMARY OF PROGRAM CONTROL.

Component	Routines	Functions
Control	MAIN	Introduces the model. Prompts for an instruction. Receives user instructions. Calls LOOK subroutine.
	LOOK	Calls the specific subroutine that performs the function specified by the user.

2.4.2 Input

The input component allows user to enter information and data. The subroutines in this component let users specify values for adjustments and parameters that enter and control program computations. Table 3 presents a summary of the input subroutines and their function.

TABLE 3. SUMMARY OF PROGRAM INPUT.

Component	Subroutine	Functions
INPUT	CHANG	Changes a parameter value.
	MODEL	Specifies new values for all the parameters of the model.
	NEW	Enters values for type II cash flow patterns.

2.4.3 Output

The various output subroutines enable users to specify the results they want and the way the results are displayed. In addition, the users can obtain information on parameter values and access a variety of help messages. Table 4 summarizes the output subroutines and their functions. Flowcharts for the PRINT and GRAPH subroutines are presented in Figs. 7 and 8.

TABLE 4. SUMMARY OF OUTPUT SUBROUTINES.

Component	Subroutine	Functions
OUTPUT	CRE	Draws any curve on the graph by specifying X and Y coordinates.
	PRINT	Prints all the results obtained from the model at any point in time.
	DELET	Deletes a line from the graph.
	GRA	Takes all of the data in memory to be graphed and stores it in a plot output file.
	HELP	Prints the appropriate help message.
	LIST	Types all the parameter values.
	LABL	Labels the last line generated on the graph.

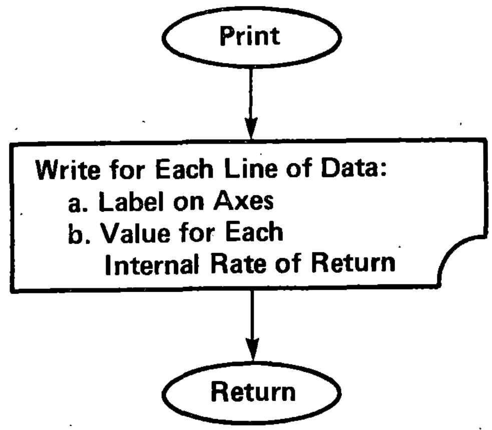


FIG. 7. FLOWCHART OF PRINT SUBROUTINE.

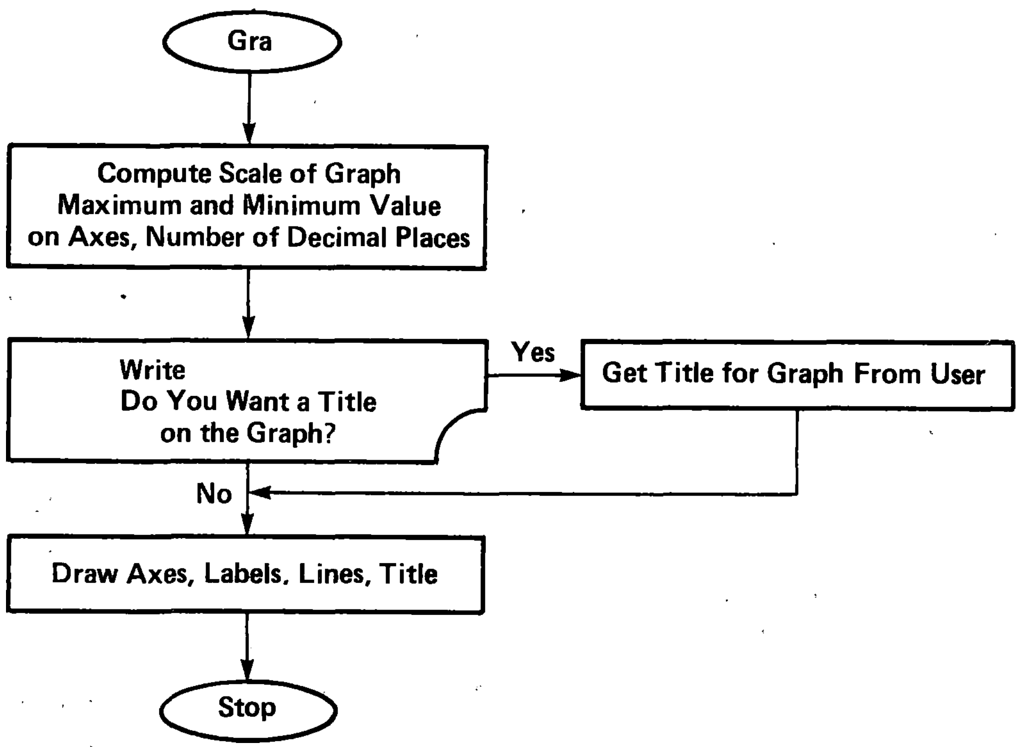


FIG. 8. FLOWCHART OF GRAPH SUBROUTINE.

3. PROGRAM OPERATION

This section introduces users to the actual operation of the model. It is meant to provide users with specific information required to operate the model. It also presents sample runs of the program.

3.1 Initiating the Program

This section presents several sample runs of the braking and coupling system financial evaluation model. The runs correspond to the type E coupler cases examined in the FRA cost/benefit study of advanced braking and coupling systems [1,2]. The sample runs evaluate Type E couplers with a shelf. A different implementation scenario is applied in each of the three sample runs presented below.

The first example includes a more detailed listing than would normally be necessary, in order to illustrate some of the information that the program provides to users. The later examples show how the program can be run more efficiently, with fewer commands, when the user requires less prompting information.

3.2 Example Cases

Since this is the first example of the FIMOD program in this manual, we will provide a fairly complete listing here; Examples 2 and 3 present a shorter version of the model for the same braking and coupling device. By comparing these examples, you can see how to run the program more efficiently once you are more familiar with it. Examples 2 and 3 also show how to use other features of the model not shown in this first example.

3.2.1 Example 1: Type E coupler with shelf - 25-year implementation

To use the program, for the first time, on the same line as the INSTRUCTION: prompt, type HELP followed by a carriage return. You will get the following response:

```
INSTRUCTION:HELP
THIS PROGRAM CONTAINS A NUMBER OF SUBPROGRAMS TO PERFORM SPECIFIC TASKS.
TO USE ONE TYPE THE KEYWORD FOR THAT UNIT.

KEYWORD          FUNCTION
 HELP      INFORMATION ABOUT A PARTICULAR QUESTION
 MODEL     SET PARAMETERS FOR BRAKING AND COUPLING MODEL
 CHANGE    CHANGE SPECIFIC PARAMETERS IN THE MODEL
 SOLVE     SOLVE FOR AVAILABLE DOLLARS PER CAR AND STORE THE RESULTS
 GRAPH     PLOT THE DATA IN THE FILE
 PRINT     PRINT THE DATA IN THE FILE
 LIST      THE PARAMETERS AND THEIR VALUES IN THE MODEL
 DELETE    REMOVE ONE OR MORE LINES FROM THE DATA FILE
 CREATE    ENTER A LINE INTO THE FILE


THIS PROGRAM WILL NOW AUTOMATICALLY ENTER MODEL AND THEN LIST.
YOU CAN THEN USE CHANGE TO CORRECT ANY ERRORS. THEN TYPE SOLVE FOLLOWED
BY PRINT OR GRAPH.
NOTE THAT PRINT AND GRAPH WILL OUTPUT ALL THE SOLUTIONS
MADE UP TO THAT TIME. PARTICULAR SOLUTIONS CAN BE REMOVED WITH DELETE.
```

The program has responded to your request for help by listing the options available to you and telling you that at this time it is "automatically" going to MODEL, the subprogram you would want when first using the program. As is shown on the next page, the MODEL subprogram will ask you for information on the parameters you want to specify for this first run of the model. After entering all the input parameters, you would probably want to use the LIST command so that you could verify your input and check for any mistakes. As the message the computer has printed indicates, this is exactly what the program is now going to do automatically; you do not have to type in the MODEL or LIST commands. The PRINT and GRAPH subprograms are described later.

Note that this is the only time the program will automatically move you to the MODEL and LIST functions. From here on you must type these commands to access these functions.

The program is now at the beginning of the MODEL subprogram. You will always know when you enter the MODEL subprogram because the computer will print the following message:

```
THIS IS A MODEL TO ESTIMATE THE AMOUNT THAT CAN
BE SPENT PER FREIGHT CAR FOR ADVANCED BRAKING
AND COUPLING.
```

The computer will immediately begin to type out, one by one, a list of questions to which you must respond. Each question asks you to specify the value you want for a particular adjustment or parameter discussed in Sec. 2.3. You must respond to each question by entering the correct word or numbers. Then press the carriage return key. The cariage return key tells the program that you are ready for the next question.

If you make a mistake answering a question, there are two ways you can correct your mistake. If you realize you have made a mistake but have not yet pressed the carriage return, press the DEL key as many times as required to "erase" your mistake. Each time you press the key you will delete one space at a time. For example, 24 <DEL> 5 changes 24 to 25; 19 <DEL><DEL> 20 changes 19 to 20. If you have already pressed the carriage return, proceed through the remaining questions and then use the CHANGE command to make your correction. We discuss how to use the CHANGE command in Sec. 3.2.2.

Proceeding with our specific example of a Type E coupler with a shelf, the values entered in response to the questions correspond to the scenario in Sec. 2.3.3 of Ref. 2. The example is to phase in shelves on new cars and cars that need replacement couplers. Implementation would take 25 years. Remember to press carriage return after typing each reply; otherwise the program will not proceed to the next question.

The answers to the questions are fairly straightforward. However, following are several peculiarities of the program that you should be aware of:

- How many years before the system becomes compatible? Since you want the system fully implemented in the 25th year, there are 24 years before the system is compatible. Hence the answer to this question will be 1 less than the year when total compatibility is achieved. The key word is before.
- What fraction of retrofit cost is required for new production? The answer in this example is 1, because there is no retrofit in this example. Rather, the new technology is introduced through new production. See example 3 to see how your answer would change in that implementation scenario.
- Do you want to enter variable savings for this range? This is an example of a Type II cash flow (c.f. Section 2.3). The savings increase over the 25-year period in proportion to the number of cars that have the new couplers. The computer will print out the schedule of savings by year after you answer the question. As the next questions indicate, this example assumes that the savings remain constant (Type I) from years 25 to 50 at $12 million.
- Method: SUM. After you select the depreciation method, in this case sum of digits, the computer will print the schedule of the amount of depreciation taken each year.

```
FOR HOW MANY YEARS SHOULD THE CASH FLOWS BE CALCULATED?50
HOW MANY YEARS BEFORE THE SYSTEM BECOMES COMPATIBLE?24
HOW MANY CARS ARE IN THE SYSTEM?1444000
WHAT IS THE EXPECTED LIFETIME OF THE CARS? 30
WHAT IS THE EXPECTED LIFETIME OF THE TECHNOLGY
BEING IMPLEMENTED ON THE RAILROAD SYSTEM? 30
WHAT FRACTION OF RETROFIT COST IS REQUIRED FOR
NEW PRODUCTION (PER CAR)?1
FRACTION= 100.0%
IS THIS CORRECT?YES
TYPE YES IF YOU WANT TO ENTER LABOR SAVINGS
THAT IS SUBJECT TO UNION PAYOFF ON A YEAR BY
YEAR BASIS.  TYPE NO OTHERWISE:  NO
WHAT IS THE LABOR SAVINGS PER YEAR THAT IS
SUBJECT TO UNION PAYOUT?0
FOR HOW MANY YEARS WILL SAVINGS BE PAID TO THE UNION?0
WHAT FRACTION OF LABOR SAVINGS ARE PAID TO THE UNION?0
TYPE YES IF YOU WANT TO ENTER SAVINGS
THAT IS NOT SUBJECT TO UNION PAYOFF ON A YEAR BY
YEAR BASIS.
TYPE NO OTHERWISE:  YES
TYPE THE END YEAR FOR THE RANGE TO ENTER SAVINGS: 24
DO YOU WANT TO ENTER VARIABLE SAVINGS FOR THIS RANGE? YES
 YEAR             SAVINGS
  1               .48E06
  2               .96E06
  3               1.44E06
  4               1.92E06
  5               2.4E06
  6               2.88E06
  7               3.36E06
  8               3.84E06
  9               4.32E06
 10               4.8E06
 11               5.28E06
 12               5.76E06
 13               6.24E06
 14               6.72E06
 15               7.2E06
 16               7.68E06
 17               8.16E06
 18               8.64E06
 19               9.12E06
 20               9.6E06
 21               10.08E06
 22               10.56E06
 23               11.04E06
 24               11.52E06
```

```
TYPE THE END YEAR FOR THE RANGE TO ENTER SAVINGS: 50
DO YOU WANT TO ENTER VARIABLE SAVINGS FOR THIS RANGE? NO
WHAT IS THE SAVINGS FROM YEAR 25 TO YEAR 50? 12E06
WHAT IS THE TAX RATE FOR THE RAILROAD INDUSTRY?.46
WHAT FRACTION OF INVESTMENTS ARE DEDUCTIBLE FOR
INVESTMENT TAX CREDIT?.1
WHAT IS THE INFLATION RATE FOR:
         MATERIALS (IN PERCENT)?10
         LABOR (IN PERCENT)?8.7
         SAVINGS NOT SUBJECT TO UNION PAYOUT (IN PERCENT)?10
WHAT IS THE DEPRECIATION LIFETIME OF THE ASSET?30
WHICH METHOD OF DEPRECIATION DO YOU WANT TO USE?
         STRAIGHT      -STRAIGHT LINE
         DOUBLE        -DOUBLE DECLINING BALANCE
         SUM           -SUM OF YEARS DIGITS
METHOD:SUM
YEAR  FRACTION WRITTEN OFF IN THAT YEAR
  1      .065
  2      .062
  3      .060
  4      .058
  5      .056
  6      .054
  7      .052
  8      .049
  9      .047
 10      .045
 11      .043
 12      .041
 13      .039
 14      .037
 15      .034
 16      .032
 17      .030
 18      .028
 19      .026
 20      .024
 21      .022
 22      .019
 23      .017
 24      .015
 25      .013
 26      .011
 27      .009
 28      .006
 29      .004
 30      .002
 31      .000
```

32	.000
33	.000
34	.000
35	.000
36	.000
37	.000
38	.000
39	.000
40	.000
41	.000
42	.000
43	.000
44	.000
45	.000
46	.000
47	.000
48	.000
49	.000

TOTAL=1.000

Now that you have entered all the input required for the model, the program goes to the LIST subprogram and prints all the data that are now stored in the program. The output of the LIST routine is shown below. The LIST output provides a chance for you to check the data and ensure that you haven't made any mistakes. Note that this is the only time the program automatically goes to LIST. Further note that while in this example there are no savings subject to union payout, the program will still print out zero savings for those years. The output also shows you that it is going to use a range of 5-25 percent as the internal rate of return. In addition, it gives you information that you would need to use the GRAPH function.*

*The GRAPH function is specific to the BBN system and its use requires the user to be physically present at BBN's computer center. Therefore, we merely mention it in this manual. Interested users can contact C. Waldman at BBN for instructions on how to use GRAPH.

VARIABLE	KEYWORD	CURRENT VALUE
NUMBER OF YEARS IN ANALYSIS	LIMIT	50
YEARS BEFORE SYSTEM IS COMPATIBLE	COMPATIBLE	24
NUMBER OF CARS	NUMBER	1444000.000
ATTRITION RATE	ATTRITION	0.033
NEW COST OF EQUIPMENT	FRACTION	1.000
INVESTMENT TAX CREDIT	INVESTMENT	0.100
TAX RATE	TAX	0.460
LOSS TO UNION	UNION	0.000
EXPECTED LIFETIME OF THE TECHNOLOGY	TECH	30.000
LABOR SAVINGS	SAVINGS	
YEAR 1		0.000
YEAR 2		0.000
YEAR 3		0.000
YEAR 4		0.000
YEAR 5		0.000
YEAR 6		0.000
YEAR 7		0.000
YEAR 8		0.000
YEAR 9		0.000
YEAR 10		0.000
YEAR 11		0.000
YEAR 12		0.000
YEAR 13		0.000
YEAR 14		0.000
YEAR 15		0.000
YEAR 16		0.000
YEAR 17		0.000
YEAR 18		0.000
YEAR 19		0.000
YEAR 20		0.000
YEAR 21		0.000
YEAR 22		0.000
YEAR 23		0.000
YEAR 24		0.000
YEAR 25		0.000
YEAR 26		0.000
YEAR 27		0.000
YEAR 28		0.000
YEAR 29		0.000
YEAR 30		0.000
YEAR 31		0.000
YEAR 32		0.000
YEAR 33		0.000
YEAR 34		0.000
YEAR 35		0.000
YEAR 36		0.000
YEAR 37		0.000
YEAR 38		0.000
YEAR 39		0.000

YEAR 40		0.000
YEAR 41		0.000
YEAR 42		0.000
YEAR 43		0.000
YEAR 44		0.000
YEAR 45		0.000
YEAR 46		0.000
YEAR 47		0.000
YEAR 48		0.000
YEAR 49		0.000
YEAR 50		0.000
YEARS SAVINGS ARE LOST TO UNION	LOSE	0
OTHER SAVINGS	OTHER	
YEAR 1		480000.000
YEAR 2		960000.000
YEAR 3		1440000.000
YEAR 4		1920000.000
YEAR 5		2400000.000
YEAR 6		2880000.000
YEAR 7		3360000.000
YEAR 8		3840000.000
YEAR 9		4320000.000
YEAR 10		4800000.000
YEAR 11		5280000.000
YEAR 12		5760000.000
YEAR 13		6240000.000
YEAR 14		6720000.000
YEAR 15		7200000.000
YEAR 16		7680000.000
YEAR 17		8160000.000
YEAR 18		8640000.000
YEAR 19		9120000.000
YEAR 20		9600000.000
YEAR 21		10080000.000
YEAR 22		10560000.000
YEAR 23		11040000.000
YEAR 24		11520000.000
YEAR 25		12000000.000
YEAR 26		12000000.000
YEAR 27		12000000.000
YEAR 28		12000000.000
YEAR 29		12000000.000
YEAR 30		12000000.000
YEAR 31		12000000.000
YEAR 32		12000000.000
YEAR 33		12000000.000
YEAR 34		12000000.000
YEAR 35		12000000.000
YEAR 36		12000000.000
YEAR 37		12000000.000
YEAR 38		12000000.000
YEAR 39		12000000.000
YEAR 40		12000000.000
YEAR 41		12000000.000
YEAR 42		12000000.000
YEAR 43		12000000.000
YEAR 44		12000000.000

```
        YEAR 45                                            12000000.000
        YEAR 46                                            12000000.000
        YEAR 47                                            12000000.000
        YEAR 48                                            12000000.000
        YEAR 49                                            12000000.000
        YEAR 50                                            12000000.000
INFLATION:                          INFLATION
      MATERIALS                                                   1.100
      LABOR                                                       1.087
      OTHER                                                       1.100
MINIMUM INTERNAL RATE OF RETURN     RATES                         5
MAXIMUM INTERNAL RATE OF RETURN     RATES                         25
DEPRECIATION                        DEPRECIATION
      LIFETIME OF ASSETS                                          30
      SUM OF YEARS DIGITS DEPRECIATION USED.
THE AXES ARE LABELED AS FOLLOWS:
X-AXIS: IRR %
Y-AXIS: DOLLARS;AVAILABLE PER;FREIGHT CAR;FOR RETROFIT
                   KEYWORD IS AXES
```

In this example, your input data, which has been printed by LIST, are correct, and you are ready to proceed. Type the command SOLVE in response to the INSTRUCTION: prompt. The program will process the input data in the manner specified in Sec. 2.3. After the computations are completed, the program will ask you for a "label." The label will identify the results for this run. In this particular example, you label the run "Case 1 ECS." This label is merely a name you use to identify the run. The sequence is shown below.

```
INSTRUCTION:SOLVE
LABEL (MAXIMUM 10 CHARACTERS):CASE 1 ECS
CASE 1 ECS     IS THE LABEL CORRECT?YES
THERE IS NOW 1 LINE ON THE GRAPH.
```

The program does not automatically print the results. You have to ask to see them. To do so, type PRINT in response to the INSTRUCTION: prompt. The PRINT routine shows the output generated by the financial evaluation model. You see that with an internal rate of return of 20 percent (line 16) the maximum allowable cost you could spend for an E coupler with a shelf would be $58.01.

```
INSTRUCTION:PRINT
LINE NUMBER:  1   LABEL: CASE 1 ECS
       IRR %       DOLLARS;AVAILABLE PER;FREIGHT CAR;FOR RETROFIT
  1         5.000     163.124
  2         6.000     156.689
  3         7.000     149.493
  4         8.000     141.660
  5         9.000     133.366
  6        10.000     124.817
  7        11.000     116.228
  8        12.000     107.798
  9        13.000      99.698
 10        14.000      92.054
 11        15.000      84.950
 12        16.000      78.427
 13        17.000      72.496
 14        18.000      67.139
 15        19.000      62.325
 16        20.000      58.012
 17        21.000      54.153
 18        22.000      50.702
 19        23.000      47.614
 20        24.000      44.846
 21        25.000      42.360
```

You have now completed your first run. If you wanted to stop at this point, you could enter the STOP instruction and then log out. However, this example assumes that you want to continue immediately with the second example, and so we proceed to Case 2. Note that to run the model, you used four principal routines: MODEL, LIST, SOLVE, and PRINT.

3.2.2 Type E coupler with shelf - 10-year implementation

In this example, we will evaluate the same technology but change the implementation period. The scenario for this example is for the total changeover to be accomplished in 10 years. During this period, new cars being introduced into service and cars being refitted with new couplers would be supplied with Type

E couplers with shelves. Cars that are expected to be operational after the conversion period would be retrofitted with them. Cars with less than 10 years of life remaining would not be retrofitted.

Since you haven't stopped the program or logged out, you are still in the program, and it is asking you for the next instruction. Use the CHANGE command to implement this example. The CHANGE command lets you change the value of one or more input parameters. In this example, you want to change parameters that control the implementation period to shorten it from 25 to 10 years.

When you give the program the CHANGE command, the program responds by telling you that you can type LIST to see which parameters could be modified. Since you already know that you want to change years before the system becomes compatible, and "other" benefits from 25 to 10 years, you respond to the CHANGE prompt by typing OTHER. The sequence would appear as follows:

```
 INSTRUCTION:CHANGE
TYPE LIST FOR A LIST OF PARAMETERS THAT CAN BE
MODIFIED. TO CHANGE A PARAMETER TYPE ITS KEYWORD
AFTER THE PROMPT. WHEN FINISHED TYPE STOP.
CHANGE:OTHER
```

The program now asks you specific questions on "other" savings. Note that in this example you specify that the end year for savings is year 9, i.e., 1 less than year 10, when you want complete implementation. The same analysis period of 50 years is specified. Thus, the changes you have made shorten the implementation period. The CHANGE routine would appear as follows:

```
TYPE YES IF YOU WANT TO ENTER SAVINGS
THAT IS NOT SUBJECT TO UNION PAYOFF ON A YEAR BY
YEAR BASIS.  TYPE ONE IF YOU WANT TO CHANGE
SAVINGS FOR ONLY ONE YEAR.
TYPE NO OTHERWISE:  YES
TYPE THE END YEAR FOR THE RANGE TO ENTER SAVINGS: 9
DO YOU WANT TO ENTER VARIABLE SAVINGS FOR THIS RANGE? YES
 YEAR              SAVINGS
  1                1.2E06
  2                2.4E06
  3                3.6E06
  4                4.8E06
  5                6E06
  6                7.2E06
  7                8.4E06
  8                9.6E06
  9                10.8E06
TYPE THE END YEAR FOR THE RANGE TO ENTER SAVINGS: 50
DO YOU WANT TO ENTER VARIABLE SAVINGS FOR THIS RANGE? NO
WHAT IS THE SAVINGS FROM YEAR 10 TO YEAR 50? 12E06
CHANGE:COMPATIBLE
HOW MANY YEARS DOES THE SYSTEM TAKE TO BECOME COMPATIBLE?9
CHANGE:STOP
```

By typing STOP at this point, you leave the CHANGE routine and return to the INSTRUCTION prompt. You could type LIST at this point, but in this example you elect to go to SOLVE. You label this example Case 2 ECS.

```
 INSTRUCTION:SOLVE
LABEL (MAXIMUM 10 CHARACTERS):CASE 2 ECS
CASE 2 ECS     IS THE LABEL CORRECT?YES
THERE ARE NOW  2 LINES ON THE GRAPH.
```

Instead of going to PRINT, you elect instead to go to the third and final case.

3.2.3 Example 3 - Type E coupler, welded shelf

This third example involves implementation of Type E coupler with shelves over a 10-year period by independently welding shelves onto existing couplers, as was done recently as a safety precaution for all tank cars that carry dangerous liquids or gases. We will proceed through this example using four steps:

- CHANGE
- LIST
- SOLVE
- PRINT.

The run for this example is shown below. Note that the procedure is very similar to the previous example - revising the input with CHANGE, running the model with the SOLVE command, and then giving the run a label.

```
 INSTRUCTION:CHANGE
TYPE LIST FOR A LIST OF PARAMETERS THAT CAN BE
MODIFIED. TO CHANGE A PARAMETER TYPE ITS KEYWORD
AFTER THE PROMPT. WHEN FINISHED TYPE STOP.
CHANGE:FRACTION
WHAT FRACTION OF RETROFIT COST IS REQUIRED FOR
NEW PRODUCTION (PER CAR)?.8
FRACTION=  80.0%
IS THIS CORRECT?YES
CHANGE:STOP
 INSTRUCTION:SOLVE
LABEL (MAXIMUM 10 CHARACTERS):CASE 3 ECS
CASE 3 ECS     IS THE LABEL CORRECT?YES
THERE ARE NOW  3 LINES ON THE GRAPH.
```

This scenario assumes that the cost of retrofitting is $140, of which $112 is for a pair of coupler shelves and $28 is for labor to weld the shelves to a set of couplers in a car. Since there is now in this example a difference between retrofit and new production costs, you change the value for FRACTION from 1 to 0.8, since $112 (new production) is 80% of retrofit cost ($140).

The final step in this example is to PRINT the output. When you enter the PRINT command, the program will print the results for as many cases or runs as it currently has in storage. Since you have worked three examples, the program will print the three complete sets of results. These are presented below.

INSTRUCTION:PRINT

LINE NUMBER: 1 LABEL: CASE 1 ECS

	IRR %	DOLLARS;AVAILABLE PER;FREIGHT CAR;FOR RETROFIT
1	5.000	163.124
2	6.000	156.689
3	7.000	149.493
4	8.000	141.660
5	9.000	133.366
6	10.000	124.817
7	11.000	116.228
8	12.000	107.798
9	13.000	99.698
10	14.000	92.054
11	15.000	84.950
12	16.000	78.427
13	17.000	72.496
14	18.000	67.139
15	19.000	62.325
16	20.000	58.012
17	21.000	54.153
18	22.000	50.702
19	23.000	47.614
20	24.000	44.846
21	25.000	42.360

LINE NUMBER: 2 LABEL: CASE 2 ECS

	IRR %	DOLLARS;AVAILABLE PER;FREIGHT CAR;FOR RETROFIT
1	5.000	164.629
2	6.000	156.943
3	7.000	148.226
4	8.000	138.734
5	9.000	128.801
6	10.000	118.781
7	11.000	109.003
8	12.000	99.728
9	13.000	91.130
10	14.000	83.304
11	15.000	76.275
12	16.000	70.022
13	17.000	64.489
14	18.000	59.610
15	19.000	55.311
16	20.000	51.520
17	21.000	48.169
18	22.000	45.200
19	23.000	42.560
20	24.000	40.203
21	25.000	38.091

```
LINE NUMBER:  3   LABEL: CASE 3 ECS
        IRR %        DOLLARS;AVAILABLE PER;FREIGHT CAR;FOR RETROFIT
  1          5.000     200.692
  2          6.000     189.892
  3          7.000     177.830
  4          8.000     164.920
  5          9.000     151.655
  6         10.000     138.526
  7         11.000     125.950
  8         12.000     114.232
  9         13.000     103.549
 10         14.000      93.973
 11         15.000      85.490
 12         16.000      78.034
 13         17.000      71.509
 14         18.000      65.809
 15         19.000      60.829
 16         20.000      56.469
 17         21.000      52.642
 18         22.000      49.269
 19         23.000      46.286
 20         24.000      43.635
 21         25.000      41.269
```

You have now finished using the FIMOD model. To terminate the session, first type STOP to leave the program.

```
INSTRUCTION:STOP
ARE YOU FINISHED WITH THE PROGRAM?YES
```

Next, log out of the system using the standard LOGO command. The system will tell you the resources you used.

```
STOP

END OF EXECUTION
CPU TIME: 0.71  ELAPSED TIME: 24.56
EXIT.
^C
@LOGO
```

3.3 The DELETE Command

As the examples in Sec. 3.2 show, the FIMOD model generates solutions for each run you make. The output of each run is identified by a label that you give each run. The program assigns a "line number" to each run as well. Thus the output of the third example was labeled "Case 3 ECS" and identified as line number 3.

The DELETE command allows you to erase (i.e., delete) the output of a given run. Let's assume that you decide a 25-year implementation period is unacceptable and you therefore no longer need the results for Case 1 ECS. You use the DELETE command to get rid of results for this case. When you type DELETE, the program tells you how many "lines" of output it has and asks you which line you want to delete. Note that you can only delete one line at a time. You can't specify more than one line in response to each LINE prompt.

In this example, you delete Case 1 ECS, which is line 1. The model tells you that you have two lines left and asks you if you:

- want to delete another line?
- see the remaining lines?
- stop the delete routine?

```
 INSTRUCTION:DELETE
THERE ARE NOW  3 LINES. WHICH ONE DO YOU
WANT TO DELETE? TYPE PRINT TO SEE THE REMAINING DATA.
TYPE STOP WHEN FINISHED.
LINE:HELP
ENTER THE LINE NUMBER (AS LISTED AFTER TYPING PRINT) CORRESPONDING TO
THE LINE YOU WISH TO DELETE. TYPE STOP WHEN FINISHED.
THERE ARE NOW  3 LINES. WHICH ONE DO YOU
WANT TO DELETE? TYPE PRINT TO SEE THE REMAINING DATA.
TYPE STOP WHEN FINISHED.
LINE:1
THERE ARE NOW  2 LINES. WHICH ONE DO YOU
WANT TO DELETE? TYPE PRINT TO SEE THE REMAINING DATA.
TYPE STOP WHEN FINISHED.
LINE:STOP
```

In this example, you ask to see the remaining lines by typing PRINT. The sequence is shown below.

```
INSTRUCTION:PRINT


LINE NUMBER:  1   LABEL: CASE 2 ECS
       IRR %         DOLLARS;AVAILABLE PER;FREIGHT CAR;FOR RETROFIT
  1          5.000     164.629
  2          6.000     156.943
  3          7.000     148.226
  4          8.000     138.734
  5          9.000     128.801
  6         10.000     118.781
  7         11.000     109.003
  8         12.000      99.728
  9         13.000      91.130
 10         14.000      83.304
 11         15.000      76.275
 12         16.000      70.022
 13         17.000      64.489
 14         18.000      59.610
 15         19.000      55.311
 16         20.000      51.520
 17         21.000      48.169
 18         22.000      45.200
 19         23.000      42.560
 20         24.000      40.203
 21         25.000      38.091

LINE NUMBER:  2   LABEL: CASE 3 ECS
       IRR %         DOLLARS;AVAILABLE PER;FREIGHT CAR;FOR RETROFIT
  1          5.000     200.692
  2          6.000     189.892
  3          7.000     177.830
  4          8.000     164.920
  5          9.000     151.655
  6         10.000     138.526
  7         11.000     125.950
  8         12.000     114.232
  9         13.000     103.549
 10         14.000      93.973
 11         15.000      85.490
 12         16.000      78.034
 13         17.000      71.509
 14         18.000      65.809
 15         19.000      60.829
 16         20.000      56.469
 17         21.000      52.642
 18         22.000      49.269
 19         23.000      46.286
 20         24.000      43.635
 21         25.000      41.269
```

```
STOP

END OF EXECUTION
CPU TIME: 44.75 ELAPSED TIME: 27:3.50
EXIT.
```

Note that the line number assigned to each case has changed. Line 2 has become line 1, and line 3 has become line 2. Each line is still identified by its label as well. It is best for you to make sure you know the label for each run, because this is how you can identify the output. The line number is the mechanism by which the computer keeps track of the output.

REFERENCES

1. E.K. Bender, A.J. Berger, J.W. Ernest, and L.E. Wittig, "Methodology for Evaluating the Cost and Benefit of Advanced Braking and Coupling Systems," U.S. Department of Transportation, Federal Railroad Administration, Report No. FRA/ORD-79-57, November 1979.

2. E.K. Bender, L.E. Wittig, and H.A. Wright, "Evaluation of the Costs and Benefits of Advanced Braking and Coupling Systems," U.S. Department of Transportation, Federal Railroad Administration, Report No. FRA/ORD-80/49, October 1980.

3. E.K. Bender, L.E. Wittig, and H.A. Wright, "Recommendations for Research and Development on Advanced Braking and Coupling Systems," U.S. Department of Transportation, Federal Railroad Administration, FRA/ORD-81/24, January 1981.

4. L.E. Wittig, C.E. Waldman, and E.K. Bender, "Railroad Yard Simulation Model: Description and Computer Program Users' Manual," U.S. Department of Transportation, Federal Railroad Administration, FRA/ORD-81/25.I, February 1981.

APPENDIX A: LISTING OF FORTRAN PROGRAM

```
      PROGRAM FIMOD
      COMMON LINES,DATA(26,2,10),POINT(10),LABEL(10,2),WIDTH(2)
     1,YLABEL(46),XLABEL(2),LIMIT,ICOMP,X,NCAR,ATTRAT,MATINF
     2,LABINF,LABORS,LOW,HIGH,PAYSTP,FRAC,FRIDT,TAXRAT,UPAYRT
     3,SCHED(49),SNSUP,OINF,METHOD,LIFE,TECH,SAVL(50),OSAV(50)
     4,IDEX1,IDEX2
      REAL NCAR,MATINF,LABINF,LABORS
      INTEGER POINT,YEAR,HIGH,LOW,PAYSTP
      LEVEL=1
      CALL PLOTS('GRF')
      TYPE 10
10    FORMAT(' BRAKING AND COUPLING',/,' TYPE HELP FOR DIRECTIONS.')
20    TYPE 30
30    FORMAT('  INSTRUCTION:',$)
      ACCEPT 40,ANS
40    FORMAT(A5)
      IMATCH=0
      IF(LEVEL.GT.2) LEVEL=2
      CALL LOOK(IMATCH,ANS,LEVEL)
      IF(ANS.EQ.'STOP') GO TO 50
      IF(IMATCH.EQ.0) GO TO 70
      GO TO 20
50    TYPE 51
51    FORMAT(' ARE YOU FINISHED WITH THE PROGRAM?',$)
      ACCEPT 40,ANS
      IF(ANS.NE.'YES') GO TO 20
      TYPE 60
60    FORMAT(' FINISHED. TURN ON PLOTTER AND TYPE THE FOLLOWING.'/
     2,' ASS PLT: (RETURN)',/
     2,' COP GRF (ESCAPE) ... (TO) PLT: (RETURN) (RETURN)',/
     3,' DEA PLT: (RETURN)')
      CALL PLTEND
      STOP
70    TYPE 80
80    FORMAT(' PLEASE CHECK THE COMMAND YOU USED. TO CHANGE A',/
     1,' PARAMETER YOU MUST FIRST TYPE CHANGE. TYPE HELP FOR',/
     2,' MORE INSTRUCTIONS.')
      GO TO 20
      STOP
      END
C
C
C
C     MODIFY PARAMETERS
      SUBROUTINE CHANG($,MATCH)
      COMMON LINES,DATA(26,2,10),POINT(10),LABEL(10,2),WIDTH(2)
     1,YLABEL(46),XLABEL(2),LIMIT,ICOMP,X,NCAR,ATTRAT,MATINF
     2,LABINF,LABORS,LOW,HIGH,PAYSTP,FRAC,FRIDT,TAXRAT,UPAYRT,
     3SCHED(49),SNSUP,OINF,METHOD,LIFE,TECH,SAVL(50),OSAV(50)
     4,IDEX1,IDEX2
      REAL NCAR,MATINF,LABINF,LABORS
      INTEGER POINT,YEAR,HIGH,LOW,PAYSTP
      MATCH=1
```

```
C
C       TYPE HEADING
10      TYPE 20
20      FORMAT(' TYPE LIST FOR A LIST OF PARAMETERS THAT CAN BE ',/
        2,' MODIFIED. TO CHANGE A PARAMETER TYPE ITS KEYWORD ',/
        3,' AFTER THE PROMPT. WHEN FINISHED TYPE STOP.')
30      TYPE 40
        LEVEL=17
40      FORMAT(' CHANGE:',$)
        ACCEPT 50,ANS
50      FORMAT(A5)
        IF(ANS.EQ.'DEPRE') GO TO 100
        IF(ANS.EQ.'LIMIT') GO TO 400
        IF(ANS.EQ.'COMPA') GO TO 430
        IF(ANS.EQ.'NUMBE') GO TO 480
        IF(ANS.EQ.'ATTRI') GO TO 520
        IF(ANS.EQ.'FRACT') GO TO 550
        IF(ANS.EQ.'INVES') GO TO 580
        IF(ANS.EQ.'TAX') GO TO 610
        IF(ANS.EQ.'UNION') GO TO 630
        IF(ANS.EQ.'SAVIN') GO TO 650
        IF(ANS.EQ.'OTHER') GO TO 670
        IF(ANS.EQ.'LOSE') GO TO 690
        IF(ANS.EQ.'INFLA') GO TO 720
        IF(ANS.EQ.'RATES') GO TO 760
        IF(ANS.EQ.'AXES') GO TO 840
        IF(ANS.EQ.'TECH') GO TO 900
        IF(ANS.EQ.'STOP') RETURN 1
55      IMATCH=0
        CALL LOOK(IMATCH,ANS,LEVEL)
        IF(IMATCH.EQ.0) TYPE 60
60      FORMAT(' PLEASE CHECK THE NAME YOU ENTERED. IT IS NOT',/
        1,' ON THE LIST. YOU MUST REENTER OR TYPE STOP.')
        GO TO 30
70      REREAD 80,ANS
        IMATCH=0
        CALL LOOK(IMATCH,ANS,LEVEL)
80      FORMAT(A5)
        IF(IMATCH.EQ.0) TYPE 90
90      FORMAT('  TYPE HELP FOR MORE INFORMATION')
        GO TO (400,430,480,520,550,650,690,630,670
        1,610,580,720,100,760) (LEVEL-2)
        TYPE 95,LEVEL
95      FORMAT(' THERE HAS BEEN AN ERROR. LEVEL=',I2)
        RETURN 1
C
C       COMPUTE DEPRECIATION SCHEDULE
100     TYPE 110
110     FORMAT(' WHAT IS THE DEPRECIATION LIFETIME OF THE ASSET?',$)
        LEVEL=15
        READ(5,120,ERR=70) LIFE
```

```
120     FORMAT(I3)
        IF(LIFE.GE.1) GO TO 140
        TYPE 130
130     FORMAT(' THE LIFETIME MUST BE ONE OR MORE. PLEASE REENTER.')
        GO TO 100
140     TYPE 150
150     FORMAT(' WHICH METHOD OF DEPRECIATION DO YOU WANT TO USE?',/
        1,T10,' STRAIGHT',T25,'-STRAIGHT LINE'
        2,/,T10,' DOUBLE',T25,'-DOUBLE DECLINING BALANCE'
        3,/,T10,' SUM',T25,'-SUM OF YEARS DIGITS',/,' METHOD:',$)
        ACCEPT 50,ANS
        IF(ANS.EQ.'STRAI') GO TO 170
        IF(ANS.EQ.'DOUBL') GO TO 190
        IF(ANS.EQ.'SUM') GO TO 240
        IF(ANS.EQ.'STOP') GO TO 165
        IMATCH=0
        CALL LOOK(IMATCH,ANS,LEVEL)
        IF(IMATCH.EQ.1) GO TO 140
C
C       PRINT ERROR MESSAGE SINCE ENTRY COULD NOT BE IDENTIFIED
        TYPE 160
160     FORMAT(' PLEASE TYPE STRAIGHT,DOUBLE, OR SUM.')
        GO TO 140
165     TYPE 166
166     FORMAT(' THE DEPRECIATION SCHEDULE HAS NOT BEEN CHANGED.')
C
C       STRAIGHT LINE METHOD
170     METHOD=1
        DO 180 I=1,49
        SCHED(I)=1.0/FLOAT(LIFE)
        IF(I.GT.LIFE) SCHED(I)=0.0
180     CONTINUE
        GO TO 300
C
C       DOUBLE DECLINING BALANCE METHOD
C
C       RATE OF DEPRECIATION IS TWICE THAT OF STRAIGHT LINE METHOD
190     METHOD=2
        PERC=2.0/FLOAT(LIFE)
        BALANC=1.0
        DO 220 I=1,48
        SCHED(I)=BALANC*PERC
220     BALANC=BALANC-SCHED(I)
        SCHED(49)=BALANC
        GO TO 300
C
C       SUM OF YEARS DIGITS METHOD
240     METHOD=3
        SUM=(LIFE**2+LIFE)/2
        DO 250 I=1,49
        SCHED(I)=FLOAT(LIFE-I+1)/SUM
        IF(I.GT.LIFE) SCHED(I)=0.0
```

```
250     CONTINUE
        GO TO 300
C
C       SHOW SCHEDULE
300     SUM=0.0
        TYPE 310
310     FORMAT(' YEAR  FRACTION WRITTEN OFF IN THAT YEAR')
        DO 330 I=1,49
        TYPE 320,I,SCHED(I)
320     FORMAT(2X,I2,5X,F5.3)
330     SUM=SUM+SCHED(I)
        TYPE 340,SUM
340     FORMAT(/,' TOTAL=',F5.3,//)
        GO TO 30
C
C       CHANGE TIME HORIZON
400     TYPE 410
410     FORMAT(' FOR HOW MANY YEARS SHOULD THE CASH FLOWS',
        1' BE CALCULATED?',$)
        LEVEL=3
        READ (5,420,ERR=70) LIMIT
420     FORMAT(I3)
        IF(LIMIT.LT.1) GO TO 450
        IF(LIMIT.GT.50) GO TO 450
        GO TO 30
C
C       CHANGE THE YEAR FLEET BECOMES COMPATIBLE
430     TYPE 440
440     FORMAT(' HOW MANY YEARS DOES THE SYSTEM TAKE TO BECOME'
        1,' COMPATIBLE?',$)
        LEVEL=4
        READ (5,420,ERR=70) ICOMP
        IF(ICOMP.LE.0) GO TO 470
        IF(ICOMP.GE.50) GO TO 470
        GO TO 30
450     TYPE 460
460     FORMAT(' YEAR MUST BE BETWEEN 1 AND 50, PLEASE REENTER.')
        GO TO 400
470     TYPE 460
        GO TO 430
C
C       NUMBER OF CARS IN THE SYSTEM
480     TYPE 490
490     FORMAT(' HOW MANY CARS ARE IN THE SYSTEM?',$)
        LEVEL=5
        READ (5,500,ERR=70) NCAR
500     FORMAT(E10.0)
        IF(NCAR.GT.0.0) GO TO 30
        TYPE 510
510     FORMAT(' THERE HAS TO BE MORE THAN ZERO CARS.')
        GO TO 480
```

```
C
C       ATTRITION RATE
520     TYPE 530
530     FORMAT(' WHAT IS THE EXPECTED LIFETIME OF THE CARS? ',$)
        LEVEL=6
        READ (5,500,ERR=70) ATTRAT
        IF(ATTRAT .NE. 0.0) ATTRAT = 1.0/ATTRAT
        IF(ATTRAT.GE.0.0.AND.ATTRAT.LE.1.0) GO TO 30
        TYPE 540
540     FORMAT(' THE EXPECTED LIFETIME OF THE CARS MUST BE',/,
        1 ' GREATER THAN OR EQUAL TO ZERO.')
        GO TO 520
C
C       ORIGINAL COST AS FRACTION OF NEW COST
550     TYPE 560
560     FORMAT(' WHAT FRACTION OF RETROFIT COST IS REQUIRED FOR',/
        1,' NEW PRODUCTION (PER CAR)?',$)
        LEVEL=7
        READ (5,500,ERR=70) FRAC
        XFRAC=FRAC*100.0
        TYPE 570,XFRAC
570     FORMAT(' FRACTION=',F6.1,'%')
        TYPE 571
571     FORMAT(' IS THIS CORRECT?',$)
        ACCEPT 50,XFRAC
        IF(XFRAC.NE.'YES') GO TO 550
        GO TO 30
C
C       FRACTION DEDUCTIBLE FOR INVESTMENT TAX CREDIT
580     TYPE 590
590     FORMAT(' WHAT FRACTION OF INVESTMENTS ARE DEDUCTIBLE FOR',/
        1,' INVESTMENT TAX CREDIT?',$)
        LEVEL=13
        READ (5,500,ERR=70) FRIDT
        IF(FRIDT.GE.0.0) GO TO 30
        TYPE 600
600     FORMAT(' FRACTION CANNOT BE LESS THAN ZERO.')
        GO TO 580
C
C       TAX RATE
610     TYPE 620
620     FORMAT(' WHAT IS THE TAX RATE FOR THE RAILROAD INDUSTRY?',$)
        LEVEL=12
        READ (5,500,ERR=70) TAXRAT
        IF(TAXRAT.GT.1) TAXRAT=TAXRAT/100.
        GO TO 30
C
C       UNION PAYOFF RATE
630     TYPE 640
```

```
640     FORMAT(' WHAT FRACTION OF LABOR SAVINGS ARE PAID'
        1,' TO THE UNION?',$)
        LEVEL=10
        READ (5,500,ERR=70) UPAYRT
        IF(UPAYRT.GT.1.) UPAYRT=UPAYRT/100.
        GO TO 30
C
C       SAVINGS SUBJECT TO UNION
650     TYPE 655
655     FORMAT(' TYPE YES IF YOU WANT TO ENTER LABOR SAVINGS',/
        1, ' THAT IS SUBJECT TO UNION PAYOFF ON A YEAR BY ',/
        2,' YEAR BASIS.  TYPE ONE IF YOU WANT TO CHANGE',/
        1,' SAVINGS FOR ONLY ONE YEAR.',/
        1,' TYPE NO OTHERWISE: ',$)
        LEVEL=8
        ACCEPT 656,ANS
656     FORMAT(A5)
        IF (ANS .EQ. 'NO')  GO TO 657
        IF(ANS.EQ.'ONE') GOTO 664
        IF (ANS .NE. 'YES') GO TO 650
        CALL NEW(SAVL,LIMIT,IDEX1)
        IF(IDEX2.EQ.1) GO TO 30
        IDEX2=1
        DO 651 I=1,ICOMP
651     OSAV(I)=0.0
        J=1+ICOMP
        DO 652 I=J,LIMIT
652     OSAV(I)=SNSUP
        GO TO 30
657     TYPE 660
660     FORMAT(' WHAT IS THE LABOR SAVINGS PER YEAR THAT IS',/
        1,' SUBJECT TO UNION PAYOUT?',$)
        READ (5,500,ERR=10) LABORS
        IF (IDEX2.EQ.0)GO TO 30
        IDEX1=1
        DO 662 I=1,ICOMP
662     SAVL(I)=0.0
        J=ICOMP+1
        DO 663 I=J,LIMIT
663     SAVL(I)=LABORS
        GO TO 30
664     IF(IDEX2.EQ.1) GOTO 668
        IDEX1=1
        IDEX2=2
        DO 665 I=1,ICOMP
        SAVL(I)=0.0
665     OSAV(I)=0.0
        J=ICOMP+1
        DO 666 I=J,LIMIT
        SAVL(I)=LABORS
666     OSAV(I)=SNSUP
```

```
668     TYPE 687
        READ(5,420,ERR=10) K
        IF(K.LT.1 .OR. K .GT. LIMIT) GO TO 667
        TYPE 660
        READ(5,500,ERR=10) SAVL(K)
        GOTO 30
667     TYPE 669,LIMIT
669     FORMAT(' THE YEAR MUST BE BETWEEN 1 AND ',I2)
        GOTO 668
C
C       SAVINGS NOT SUBJECT TO UNION PAYOFF
670     TYPE 680
680     FORMAT(' TYPE YES IF YOU WANT TO ENTER SAVINGS',/
        1,' THAT IS NOT SUBJECT TO UNION PAYOFF ON A YEAR BY ',/
        2,' YEAR BASIS.  TYPE ONE IF YOU WANT TO CHANGE',/
        1,' SAVINGS FOR ONLY ONE YEAR.',/
        1,' TYPE NO OTHERWISE:  ',$)
        LEVEL=11
        ACCEPT 656,ANS
        IF(ANS .EQ. 'NO') GO TO 673
        IF(ANS .EQ. 'ONE') GO TO 684
        IF(ANS .NE. 'YES') GO TO 670
        CALL NEW(OSAV,LIMIT,IDEX2)
        IF(IDEX1.EQ.1)GOTO 30
        IDEX1=1
        DO 671 I=1,ICOMP
671     SAVL(I)=0.0
        J=1+ICOMP
        DO 672 I=J,LIMIT
672     SAVL(I)=LABORS
        GOTO 30
673     TYPE 681
681     FORMAT(' WHAT IS THE ANNUAL SAVINGS NOT SUBJECT TO UNION',/
        1, ' PAYOUT? ',$)
        READ (5,500,ERR=10) SNSUP
        IF(IDEX1.EQ.0)GO TO 30
        IDEX2=1
        DO 682 I=1,ICOMP
682     OSAV(I)=0.0
        J=ICOMP+1
        DO 683 I=J,LIMIT
683     OSAV(I)=SNSUP
        GOTO 30
684     IF(IDEX1.EQ.1)GOTO 688
        IDEX1=1
        IDEX2=1
        DO 685 I=1,ICOMP
        SAVL(I)=0.0
685     OSAV(I)=0.0
        J=ICOMP+1
        DO 686 I=J,LIMIT
        SAVL(I)=LABORS
```

```
686     OSAV(I)=SNSUP
688     TYPE 687
687     FORMAT(' FOR WHAT YEAR SHOULD ANNUAL SAVINGS BE CHANGED? ',$)
        READ(5,420,ERR=10) K
        IF(K.LT.1 .OR. K .GT. LIMIT) GO TO 674
        TYPE 681
        READ(5,500,ERR=10) OSAV(K)
        GOTO 30
674     TYPE 669,LIMIT
        GOTO 688
C
C       STOP PAYING OFF UNION
690     TYPE 700
700     FORMAT(' FOR HOW MANY YEARS WILL SAVINGS BE PAID TO THE UNION?'
        1,$)
        LEVEL=9
        READ (5,420,ERR=70) PAYSTP
        I=LIMIT-ICOMP
        IF(PAYSTP.GE.0.0.AND.PAYSTP.LE.I) GO TO 30
        TYPE 710,LIMIT
710     FORMAT(' THERE MUST BE BETWEEN ZERO AND ',I2,' YEARS.')
        GO TO 690
C
C       INFLATION RATES
720     TYPE 730
730     FORMAT(' WHAT IS THE INFLATION RATE FOR:',/
        1,T10,' MATERIALS (IN PERCENT)?',$)
        LEVEL=14
        READ (5,500,ERR=70) MATINF
        MATINF=1+(MATINF/100.)
        TYPE 740
740     FORMAT('+',T10,' LABOR (IN PERCENT)?',$)
        READ (5,500,ERR=70) LABINF
        LABINF=1.+(LABINF/100.)
        TYPE 750
750     FORMAT('+',T10,' SAVINGS NOT SUBJECT TO UNION PAYOUT'
        1,' (IN PERCENT)?',$)
        READ (5,500,ERR=70) OINF
        OINF=1.+(OINF/100.)
        GO TO 30
C
C       RANGE OF INTERNAL RATES OF RETURN TO BE USED
760     TYPE 770
770     FORMAT(' WHAT IS THE MINIMUM INTERNAL RATE OF RETURN ,/
        1, (IN PERCENT)?',$)
        LEVEL=16
        READ (5,420,ERR=70) LOW
        TYPE 780
780     FORMAT(' WHAT IS THE MAXIMUM INTERNAL RATE OF RETURN ,/
        1,  (IN PERCENT)?',$)
        READ (5,420,ERR=70) HIGH
```

```
790     I=HIGH-LOW
        IF(I.EQ.0) GO TO 800
        IF(I.GT.0.AND.I.LT.25) GO TO 30
        IF(I.GE.25) GO TO 820
        I=HIGH
        HIGH=LOW
        LOW=I
        GO TO 790
800     TYPE 810
810     FORMAT(' PLEASE SPECIFY A WIDER RANGE.')
        GO TO 760
820     TYPE 830
830     FORMAT(' PLEASE SPECIFY A NARROWER RANGE (LESS THAN'
        1,' 25 PERCENTAGE POINTS).')
        GO TO 760
C
C       LABELS ON AXES
840     TYPE 850
850     FORMAT(' WHAT IS THE NEW LABEL FOR THE X-AXIS?',/
        1,' MAXIMUM 10 CHARACTERS:',$)
        ACCEPT 860,XLABEL(1),XLABEL(2)
860     FORMAT(2A5)
        TYPE 870
870     FORMAT(' WHAT IS THE NEW LABEL FOR THE Y-AXIS?',/
        1,' USE ; INSTEAD OF A CARRAIGE RETURN.'
        1,' MAXIMUM 46 CHARACTERS:',$)
        ACCEPT 861,(YLABEL(I),I=1,46)
861     FORMAT(46A1)
        TYPE 880,XLABEL(1),XLABEL(2),(YLABEL(I),I=1,46)
880     FORMAT(' THE NEW LABELS ARE',/,' X-AXIS: ',2A5
        1,/,' Y-AXIS: ',40A1,/,' ARE THESE CORRECT?',$)
        ACCEPT 50,ANS
        IF(ANS.NE.'YES') GO TO 840
        GO TO 30
C
C       THE EXPECTED LIFETIME OF THE TECHNOLOGY
900     TYPE 910
910     FORMAT(' WHAT IS THE EXPECTED LIFETIME OF THE TECHNOLGY',/
        1, ' BEING IMPLEMENTED ON THE RAILROAD SYSTEM? ',$)
        LEVEL=6
        READ (5,500,ERR=70) TECH
        IF(TECH.GT.0.0) GO TO 30
        TYPE 930
930     FORMAT(' THE EXPECTED LIFETIME OF THE TECHNOLOGY MUST BE',/,
        1 ' GREATER THAN ZERO.')
        GO TO 900
        END
```

```
C       GENERATE A LINE OF DATA
        DIMENSION X(26,2)
        COMMON LINES,DATA(26,2,10),POINT(10),LABEL(10,2),WIDTH(2)
        1,YLABEL(46),XLABEL(2)
        INTEGER POINT
        MATCH=1
        TYPE 10
10      FORMAT(' TYPE STOP TO TERMINATE ENTRY',/
        1,' TYPE ERROR TO REENTER A NUMBER',/)
        I=0
20      I=I+1
25      IF(I.LT.10) TYPE 30,I
        IF(I.GE.10) TYPE 40,I
30      FORMAT('+X',I1,':'$)
40      FORMAT('+X',I2,':',$)
        READ (5,50,ERR=100),X(I,1)
50      FORMAT(F10.0)
        IF(I.LT.10) TYPE 60,I
        IF(I.GE.10) TYPE 70,I
60      FORMAT('+Y',I1,':',$)
70      FORMAT('+Y',I2,':',$)
        READ (5,50,ERR=100),X(I,2)
        TYPE 80,X(I,1),X(I,2)
80      FORMAT('+X=',F10.3,2X,'Y=',F10.3,//)
        IF(I.LT.26) GO TO 20
        TYPE 90
90      FORMAT(' DATA VECTOR IS FULL',/
        1,' NO MORE POINTS CAN BE PLOTTED ON THIS LINE')
        GO TO 190
100     REREAD 110,ANS
110     FORMAT(A5)
        I=I-1
        IF(ANS.EQ.'STOP') GO TO 190
        CALL LOOK(IMATCH,ANS,LEVEL)
C       ASSUME ERROR NEEDS TO BE CORRECTED
115     TYPE 120
120     FORMAT(' TYPE STOP TO TERMINATE ENTRY',/
        1,' X TO CORRECT AN X VALUE',/
        1,' Y TO CORRECT A Y VALUE',/
        3,' R TO RESUME NORMAL ENTRY')
125     TYPE 130
130     FORMAT(' X,Y,R OR STOP:',$)
        ACCEPT 110,ANS
        J=0
        IF(ANS.EQ.'R') GO TO 20
        IF(ANS.EQ.'STOP') GO TO 190
        IF(ANS.EQ.'X') J=1
        IF(ANS.EQ.'Y') J=2
        IF(J.EQ.0) GO TO 115
135     TYPE 140,ANS
140     FORMAT(' WHICH ',A1,' DO YOU WANT TO CORRECT?',$)
        READ (5,150,ERR=115) K
```

```
150     FORMAT(I2)
        IF(K.LE.0) GO TO 170
        IF(K.GT.I+1) GO TO 170
        IF(K.EQ.I+1) GO TO 25
        TYPE 160,ANS,K
160     FORMAT(1X,A1,I2,'=',$)
        READ (5,50,ERR=115) X(K,J)
        GO TO 125
170     TYPE 180,I
180     FORMAT(' MUST BE BETWEEN 1 AND ',I2,', PLEASE REENTER')
        GO TO 135
C       CLOSE ENTRY
190     TYPE 200
200     FORMAT(10X,'X',10X,'Y')
        DO 210 K=1,I
210     TYPE 220,K,X(K,1),X(K,2)
220     FORMAT(1X,I2,F10.3,5X,F10.3)
230     TYPE 240
240     FORMAT(' IS THIS CORRECT (YES OR NO)?',$)
        ACCEPT 110,ANS
        IF(ANS.EQ.'NO') GO TO 115
        IF(ANS.NE.'YES') GO TO 230
C       FILE DATA
        LINES=LINES+1
        POINT(LINES)=I
        DO 250 J=1,2
        DO 250 M=1,I
250     DATA(M,J,LINES)=X(M,J)
        CALL LABL($260,IMATCH)
260     RETURN 1
        END
C
C
C
C       PROVIDE INITIAL VALUES FOR PARAMETERS
        BLOCK DATA
        COMMON LINES,DATA(26,2,10),POINT(10),LABEL(10,2),WIDTH(2)
        1,YLABEL(46),XLABEL(2),LIMIT,ICOMP,X,NCAR,ATTRAT,MATINF
        2,LABINF,LABORS,LOW,HIGH,PAYSTP,FRAC,FRIDT,TAXRAT,UPAYRT
        3,SCHED(49),SNSUP,OINF,METHOD,LIFE,TECH,SAVL(50),OSAV(50)
        4,IDEX1,IDEX2
        REAL NCAR,MATINF,LABINF,LABORS
        INTEGER POINT,YEAR,HIGH,LOW,PAYSTP
        DATA LINES/0/
        DATA DATA/520*0.0/
        DATA POINT/10*0/
        DATA LABEL/20*'     '/
        DATA WIDTH/5.5,9./
        DATA YLABEL/'D','O','L','L','A','R','S',';','A','V','A','I','L'
        1,'A','B','L','E',' ','P','E','R',';','F','R','E','I','G','H',
        2'T',' ','C','A','R',';','F','O','R',' ','R','E','T','R',
```

```
      3'O','F','I','T'/
      DATA XLABEL/'IRR %','      '/
      DATA LIMIT/21/
      DATA ICOMP/5/
      DATA NCAR/1.7E06/
      DATA ATTRAT/0.037/
      DATA MATINF/1.1/
      DATA LABINF/1.087/
      DATA LABORS/220E06/
      DATA LOW/5/
      DATA HIGH/25/
      DATA PAYSTP/10/
      DATA FRAC/0.5/
      DATA FRIDT/0.1/
      DATA TAXRAT/0.46/
      DATA UPAYRT/.25/
      DATA SNSUP/0.0/
      DATA OINF/1.1/
      DATA METHOD/3/
      DATA LIFE/16/
      DATA SCHED/.118,.110,.103,.096,.088,.081,.074,.066
      1,.059,.051,.044,.037,.029,.022,.015,.007,33*0.0/
      DATA SAVL/30E06,60E06,90E06,120E06,150E06,5*158E06
      1,15*537E06,25*0./
      DATA OSAV/1.,2.2,3.4,4.6,5.8,20*59.2,25*0./
      DATA IDEX1/0/
      DATA IDEX2/0/
      DATA TECH/0.037/
      END
C
C
C
C     DELETE A LINE FROM DATA AND COMPRESS
      SUBROUTINE DELET($,MATCH)
      COMMON LINES,DATA(26,2,10),POINT(10),LABEL(10,2)
      INTEGER POINT
      MATCH=1
1     IF(LINES.GT.0) GO TO 5
      TYPE 2
2     FORMAT(' THERE ARE NO MORE LINES TO DELETE.')
      RETURN 1
5     TYPE 10,LINES
10    FORMAT(' THERE ARE NOW ',I2,' LINES. WHICH ONE DO YOU',/
      1,' WANT TO DELETE? TYPE PRINT TO SEE THE REMAINING DATA.',/
      2,' TYPE STOP WHEN FINISHED.',/
      3,' LINE:',$)
      LEVEL=19
      READ (5,20,ERR=45) LINE
20    FORMAT(I3)
      IF(LINE.LT.1) GO TO 30
      IF(LINE.GT.LINES) GO TO 30
      IF(LINE.NE.LINES) GO TO 60
      LINES=LINES-1
      GO TO 1
```

```
C
C       ERROR MESSAGE
30      TYPE 40,LINES
40      FORMAT(' LINE NUMBER MUST BE BETWEEN 1 AND ',I2,/
        2,' PLEASE REENTER OR TYPE STOP.')
        GO TO 1
45      REREAD 50,ANS
50      FORMAT(A5)
        IF(ANS.EQ.'STOP') RETURN 1
        CALL LOOK(IMATCH,ANS,LEVEL)
        GO TO 1
C
C       COMPRESS DATA
60      DO 90 I=LINE,LINES-1
        J=I+1
        ENCODE(5,70,LABEL(I,1)) LABEL(J,1)
        ENCODE(5,70,LABEL(I,2)) LABEL(J,2)
70      FORMAT(A5)
        DO 80 K=1,POINT(J)
        DATA(K,1,I)=DATA(K,1,J)
        DATA(K,2,I)=DATA(K,2,J)
80      CONTINUE
90      POINT(I)=POINT(J)
        LINES=LINES-1
        GO TO 1
        END
C
C
C
C
        SUBROUTINE GRA($,MATCH)
C
C       PLOT AXES, PLOT EACH LINE IN THE MATRIX
C       DATA, LABEL EACH LINE, PLACE TITLE UNDER GRAPH
        DIMENSION MIN(2),MAX(2),DIFF(2),TITLE(50),ISORT(10),
        1MAG(2),ITICK(2),IDEL(2),XINC(2),IDEC(2),SCALE(2),ISIG(2)
        COMMON LINES,DATA(26,2,10),POINT(10),LABEL(10,2),WIDTH(2)
        1,YLABEL(46),XLABEL(2)
        INTEGER POINT,IEND(10)
        REAL MIN,MAX,MAG
        DATA TITLEX/4.0/
        DATA TITLEY/9.00/
        DATA TITLE/50*' '/
        MATCH=1
C
C       DETERMINE IF THERE IS PLOTTING TO BE DONE
        IF(LINES.LT.1) GO TO 170
C
```

```
C       FIND MINIMUM AND MAXIMUM VALUES FOR EACH AXIS
        DO 50 J=1,2
        MAX(J)=DATA(1,J,1)
        MIN(J)=DATA(1,J,1)
        DO 10 K=1,LINES
        DO 10 I=1,POINT(K)
        IF(MIN(J).GT.DATA(I,J,K)) MIN(J)=DATA(I,J,K)
10      IF(MAX(J).LT.DATA(I,J,K)) MAX(J)=DATA(I,J,K)
C
C       FIND RANGE FOR EACH AXIS AND CHOOSE UPPER AND LOWER
C       BOUNDS SO THAT BOUNDARIES WILL BE ROUND NUMBERS
        DIFF(J)=MAX(J)-MIN(J)
        A=DIFF(J)
        IF(A.EQ.0.0) GO TO 170
        IEXP=0
        PTEN=1.0
11      IF(A.GE.1.0) GO TO 12
        A=A*10.0
        IEXP=IEXP-1
        PTEN=PTEN/10.0
        GO TO 11
12      IF(A.LT.10.0) GO TO 13
        A=A/10.0
        IEXP=IEXP+1
        PTEN=PTEN*10
        GO TO 12
13      MAG(J)=AINT(A)
C
C       XINC IS THE INTERVAL BETWEEN SLASHES ON THE AXES
        XINC(J)=.25
        IF(MAG(J).GE.2.0) XINC(J)=.5
        IF(MAG(J).GE.4.0) XINC(J)=1.0
        IF(MAG(J).GE.8.0) XINC(J)=2.0
        XINC(J)=XINC(J)*PTEN
        MIX=IFIX(MIN(J)/XINC(J)+.01)
        MAN=IFIX(MAX(J)/XINC(J)-.01)
        IF(MIN(J).LE.0.0) MIX=MIX-1
        IF(MAX(J).GT.0.0) MAN=MAN+1
C
C       CHOOSE UPPER BOUND AS THE LOWEST ROUND NUMBER ABOVE
C       THE MAXIMUM VALUE TO BE PLOTTED
        MAX(J)=XINC(J)*MAN
        MIN(J)=XINC(J)*MIX
C
C       ITICK IS THE NUMBER OF SLASHES TO BE DRAWN ON THE AXIS
        ITICK(J)=1+MAN-MIX
        DIFF(J)=MAX(J)-MIN(J)
C
C       IDEC IS THE NUMBER OF FIGURES TO THE RIGHT OF THE DECIMAL
C       POINT TO BE WRITTEN NEXT TO THE TICKS ON THE AXES
        IDEC(J)=0-IEXP
```

```
C
C       ISIG IS THE NUMBER OF DIGITS PLOTTED NEXT TO THE TICKS
        X=AMAX1(ABS(MIN(J)),ABS(MAX(J)))
        ISIG(J)=2+MAXO(0,INT(ALOG10(X)))+MAXO(0,IDEC(J))
C
C       COMPUTE SCALE FACTOR BASED ON THE GRAPH DIMENSIONS
        IF(WIDTH(J).LT.1.0) GO TO 190
45      SCALE(J)=DIFF(J)/WIDTH(J)
50      CONTINUE
C
C       FIND TITLE FOR GRAPH
51      TYPE 52
52      FORMAT(' DO YOU WANT A TITLE ON THIS GRAPH?',$)
        LEVEL=20
        ACCEPT 55,ANS
55      FORMAT(A5)
        IF(ANS.EQ.'NO') GO TO 54
        IF(ANS.EQ.'YES') GO TO 345
        CALL LOOK(IMATCH,ANS,LEVEL)
        GO TO 51
300     TYPE 310
310     FORMAT(' HOW FAR ABOVE THE X-AXIS DO YOU WANT THE'/
        1,' TOP OF THE FIRST LINE TO BE (IN INCHES,BETWEEN -1 AND 10)?'
        2,$)
        READ (5,330,ERR=400) TITLEY
330     FORMAT(E10.0)
        TYPE 340
340     FORMAT(' HOW FAR TO THE RIGHT OF THE Y-AXIS DO YOU WANT'
        1,/,' THE LEFT HAND EDGE OF THE TITLE (BETWEEN 0 AND 6)?',$)
        READ (5,330,ERR=400) TITLEX
345     TYPE 350,TITLEY,TITLEX
350     FORMAT(' THE TITLE WILL BE ',F4.1,' INCHES ABOVE THE X-AXIS',/
        1,' AND ',F4.1,' INCHES TO THE RIGHT OF THE Y-AXIS.',/
        2,' IS THIS CORRECT?',$)
        ACCEPT 55,ANS
        IF(ANS.NE.'YES') GO TO 300
355     TYPE 360
360     FORMAT(' WHAT IS THE TITLE? USE ; INSTEAD OF CARRIAGE RETURN',/
        1,' MAXIMUM 50 CHARACTERS:',$)
        ACCEPT 370,(TITLE(I),I=1,50)
370     FORMAT(50A1)
        TYPE 380,(TITLE(I),I=1,50)
380     FORMAT(' IS THIS CORRECT: ',50A1,/,' (YES OR NO):',$)
        ACCEPT 55,ANS
        IF(ANS.NE.'YES') GO TO 355
        GO TO 54
400     REREAD 55,ANS
        CALL LOOK(IMATCH,ANS,LEVEL)
        GO TO 300
C
C       MOVE PAPER AND CHOOSE ORIGIN
```

```
54      CALL PLOT(0.0,0.0,-3)
        CALL PLOT(3.0,0.0,-2)
        CALL PLOT(10.0,-12.0,-3)
        CALL PLOT(0.0,1.0,-1)
C
C       DRAW X-AXIS
        X=WIDTH(1)
        CALL PLOT(X,0.0,2)
C
C       LABEL THE X-AXIS
        X=WIDTH(1)+.3
        DO 56 J=1,2
        IF(XLABEL(J).EQ.'     ') GO TO 57
        CALL SYMBOL(X,-0.25,.15,XLABEL(J),0.0,5)
        X=X+.75
56      CONTINUE
C
C       DRAW TICKS ON THE X-AXIS
57      DO 60 I=ITICK(1),1,-1
        XPOINT=MIN(1)+XINC(1)*(I-1)
        XLOC=XINC(1)*(I-1)/SCALE(1)
        CALL PLOT(XLOC,0.0,3)
        CALL PLOT(XLOC,-.1,2)
        X=XLOC-(ISIG(1)*0.075)
        CALL NUMBER(X,-.25,.15,XPOINT,0.0,IDEC(1))
60      CONTINUE
C
C       DRAW Y-AXIS
        Y=WIDTH(2)
        CALL PLOT(0.0,0.0,3)
        CALL PLOT(0.0,Y,2)
C
C       LABEL THE Y-AXIS
        Y=WIDTH(2)+0.3
        X=.1
        DO 64 I=1,46
        IF(YLABEL(I).NE.';') GO TO 61
        Y=Y-.2
        X=.1
        GO TO 64
61      CALL SYMBOL(X,Y,0.15,YLABEL(I),0.0,1)
        X=X+0.15
64      CONTINUE
C
C       DRAW TICKS ON THE Y-AXIS
65      DO 70 I=ITICK(2),1,-1
        YPOINT=MIN(2)+XINC(2)*(I-1)
        YLOC=XINC(2)*(I-1)/SCALE(2)
        CALL PLOT(0.0,YLOC,3)
        CALL PLOT(-.1,YLOC,2)
        Y=YLOC-.05
        X=ISIG(2)*-0.15-0.1
        CALL NUMBER(X,Y,.15,YPOINT,0.0,IDEC(2))
```

```
70      CONTINUE
C
C       CALCULATE BASELINE; POINT WHERE (0,0) WOULD PLOT
        XZERO=MIN(1)/SCALE(1)
        YZERO=MIN(2)/SCALE(2)
C
C       MOVE PEN TO START OF LINE
        DO 90 LINE=1,LINES
        X=DATA(1,1,LINE)/SCALE(1)-XZERO
        Y=DATA(1,2,LINE)/SCALE(2)-YZERO
        CALL PLOT(X,Y,3)
        IF(POINT(LINE).LE.1) GO TO 80
C
C       DRAW A LINE
        DO 80 I=2,POINT(LINE)
        X=DATA(I,1,LINE)/SCALE(1)-XZERO
        Y=DATA(I,2,LINE)/SCALE(2)-YZERO
        CALL PLOT(X,Y,2)
80      CONTINUE
90      CONTINUE
C
C       PUT LABEL TO THE RIGHT OF EACH LINE
C
C       FIND ENDPOINTS OF LINES
        IF(LINES.EQ.1) GO TO 140
        DO 100 LINE=1,LINES
        IEND(LINE)=1
        ISORT(LINE)=LINE
        IF(POINT(LINE).LT.2) GO TO 135
        DO 100 IBUBLE=2,POINT(LINE)
        IF(DATA(IBUBLE,1,LINE).GT.DATA(IEND(LINE),1,LINE))
        1 IEND(LINE)=IBUBLE
100     CONTINUE
C
C       SORT ENDPOINTS OF LINES SO LABELS WILL APPEAR IN THE RIGHT ORDER
110     IBUBLE=0
        IDONE=1
120     IBUBLE=IBUBLE+1
        JBUBLE=IBUBLE+1
        ILINE=ISORT(IBUBLE)
        JLINE=ISORT(JBUBLE)
        YI=DATA(IEND(ILINE),2,ILINE)
        YJ=DATA(IEND(JLINE),2,JLINE)
        IF(YI.LE.YJ) GO TO 130
C
C       SWITCH POINTERS (USE JPOINT AS TEMPORARY STORAGE)
        JPOINT=ISORT(IBUBLE)
        ISORT(IBUBLE)=ISORT(JBUBLE)
        ISORT(JBUBLE)=JPOINT
        IDONE=0
130     IF(JBUBLE.LT.LINES) GO TO 120
```

```
C
C       IF STILL OUT OF ORDER RETURN FOR ANOTHER PASS
        IF(IDONE.EQ.0) GO TO 110
135     X=WIDTH(1)+.75
        YJ=0.0
        DO 150 IBUBLE=1,LINES
        LINE=ISORT(IBUBLE)
        Y=DATA(IEND(LINE),2,LINE)/SCALE(2)-YZERO-0.1
C
C       CHECK FOR OVERWRITE
        IF(IBUBLE.EQ.1) GO TO 140
        JBUBLE=IBUBLE-1
        JLINE=ISORT(JBUBLE)
C
C       MOVE LABEL UP IF IT WILL OVERWRITE PREVIOUS LABEL
        YJ=Y-YJ
        IF(YJ.GT.0.2) GO TO 140
        Y=Y+0.2-YJ
C
C       PUT LABEL NEXT TO ENDPOINT
140     IF(LABEL(LINE,1).EQ.'     ') GO TO 145
        CALL SYMBOL(WIDTH(1),Y,0.15,LABEL(LINE,1),0.0,5)
        IF(LABEL(LINE,2).EQ.'     ') GO TO 145
        CALL SYMBOL(X,Y,0.15,LABEL(LINE,2),0.0,5)
145     YJ=Y
150     CONTINUE
C
C       PUT TITLE UNDER GRAPH
        Y=TITLEY-0.2
        X=TITLEX
        DO 155 I=1,50
        IF(TITLE(I).NE.';') GO TO 152
        X=TITLEX
        Y=Y-0.25
        GO TO 155
152     CALL SYMBOL(X,Y,0.2,TITLE(I),0.0,1)
        X=X+0.2
155     CONTINUE
160     TYPE 161
161     FORMAT(' TYPE YES IF YOU ARE FINISHED WITH THE',/
        1,' DATA JUST GRAPHED. TYPE NO IF YOU WISH TO',/
        2,' USE IT AGAIN. CLEAR DATA?',$)
        ACCEPT 162,ANS
162     FORMAT(A3)
        IF(ANS.EQ.'NO') GO TO 165
        IF(ANS.NE.'YES') GO TO 160
        LINES=0
165     RETURN 1
170     TYPE 180
180     FORMAT(' THERE ARE NO LINES TO PLOT')
        RETURN 1
```

```
190     TYPE 200
200     FORMAT(' WIDTH IS TOO SMALL. PLEASE CORRECT')
        RETURN 1
        END
C
C
C       PROVIDE USER INSTRUCTIONS AT VARIOUS POINTS OF THE PROGRAM
        SUBROUTINE HELP($,MATCH,LEVEL)
        MATCH=1
        GO TO (20,40,80,100,110,120,130,140,150,160,170,180,190
        1,200,60,210,230,250,270,290) LEVEL
C
C       IF LEVEL = 0, NO MORE INFORMATION IS AVAILABLE
        TYPE 10
10      FORMAT(' THERE IS NO MORE INFORMATION AVAILABLE'
        1,' FOR THIS SECTION.' )
        RETURN 1
C
C       MODEL HAS NOT YET BEEN CALLED
20      TYPE 30
30      FORMAT(' THIS PROGRAM CONTAINS A NUMBER OF SUBPROGRAMS'
        1,' TO PERFORM SPECIFIC TASKS.',/,' TO USE ONE '
        2,'TYPE THE KEYWORD FOR THAT UNIT.'
        3,//,' KEYWORD',T20,'FUNCTION'
        4,/,'  HELP',T14,'INFORMATION ABOUT A PARTICULAR QUESTION'
        5,/,'  MODEL',T14,'SET PARAMETERS FOR BRAKING'
        5,' AND COUPLING MODEL'
        6,/,'  CHANGE',T14,'CHANGE SPECIFIC PARAMETERS IN THE MODEL'
        7,/,'  SOLVE',T14,'SOLVE FOR AVAILABLE DOLLARS PER CAR'
        8,' AND STORE THE RESULTS'
        9,/,'  GRAPH',T14,'PLOT THE DATA IN THE FILE'
        1,/,'  PRINT',T14,'PRINT THE DATA IN THE FILE'
        2,/,'  LIST',T14,'THE PARAMETERS AND THEIR VALUES IN THE MODEL'
        3,/,'  DELETE',T14,'REMOVE ONE OR MORE LINES FROM THE DATA FILE'
        4,/,'  CREATE',T14,'ENTER A LINE INTO THE FILE',//)
        TYPE 35
35      FORMAT(/,' THIS PROGRAM WILL NOW AUTOMATICALLY ENTER MODEL'
        6,' AND THEN LIST.',/,' YOU CAN THEN USE CHANGE TO CORRECT',
        7' ANY ERRORS. THEN TYPE SOLVE FOLLOWED',/,' BY PRINT OR GRAPH.'
        8,/,' NOTE THAT PRINT AND GRAPH WILL OUTPUT ALL THE SOLUTIONS'
        9,/,' MADE UP TO THAT TIME. PARTICULAR SOLUTIONS CAN BE REMOVED'
        1,' WITH DELETE.'//)
        CALL MODEL($36,IMATCH,LEVEL)
36      LEVEL=2
        RETURN 1
C
C       INSTRUCTION:HELP
40      TYPE 30
        TYPE 50
```

```
50      FORMAT(' IF THERE ARE ANY PARAMETERS THAT NEED TO'
        1,' BE CHANGED TYPE CHANGE,',/,' OTHERWISE TYPE SOLVE.'
        2,/,' YOU WILL BE ASKED FOR A LABEL THAT WILL'
        3,' BE PRINTED NEXT TO THE DATA JUST',/,' OBTAINED.'
        4,' THEN YOU MAY CHANGE THE PARAMETERS TO CONSTRUCT A NEW'
        5,/,' MODEL. THERE MAY BE UP'
        6,' TO TEN LINES ON THE GRAPH.')
        LEVEL=0
        RETURN 1
C
C       LIFETIME OF THE ASSET (MODEL,CHANGE)
60      TYPE 70
70      FORMAT(' THE DEPRECIATION SCHEDULE IS BASED ON'
        1,' THE LIFETIME ASSIGNED TO THE ',/,' EQUIPMENT.'
        2,' YOUR ANSWER SHOULD BE AN INTEGER BETWEEN 1 AND 99.')
        RETURN 1
C
C       TIME HORIZON
80      TYPE 90
90      FORMAT(' THE FIRST CASH FLOW WILL BE ASSUMED TO BE'
        1,' IN YEAR ONE. YOUR',/,' RESPONSE SHOULD BE BETWEEN 1 AND 26.'
        2,' CASH FLOWS OCCURING AFTER THIS '
        1,/,' LIMIT WILL BE IGNORED.')
        RETURN 1
C
C       COMPATIBLE
100     TYPE 10
        RETURN 1
110     TYPE 10
        RETURN 1
120     TYPE 10
        RETURN 1
130     TYPE 10
        RETURN 1
140     TYPE 10
        RETURN 1
150     TYPE 10
        RETURN 1
160     TYPE 10
        RETURN 1
170     TYPE 10
        RETURN 1
180     TYPE 10
        RETURN 1
190     TYPE 10
        RETURN 1
200     TYPE 10
        RETURN 1
210     TYPE 220
220     FORMAT(' THESE RATES REFER TO THE MAXIMUM AND MINIMUM'
        1,' RATES TO BE USED IN',/,' PLOTTING DOLLARS VERSUS'
        2,' DISCOUNT RATE.')
        RETURN 1
```

```
230     TYPE 240
240     FORMAT(' TYPE LIST FOR A LIST OF PARAMETERS. THEN'
        1,' TYPE THE KEYWORD OF THE',/,' PARAMETER YOU WISH'
        2,' TO CHANGE. WHEN YOU ARE FINISHED CHANGING, TYPE STOP.')
        RETURN 1
250     TYPE 260
260     FORMAT(' AFTER X TYPE THE VALUE YOU WISH TO REFER TO THE'
        1,' X-COORDINATE OF A POINT',/,' ON THE LINE. NOTE THAT THE'
        2,' POINTS WILL BE CONNECTED IN THE ORDER'
        1,/,' YOU ENTER THEM.')
        RETURN 1
270     TYPE 280
280     FORMAT(' ENTER THE LINE NUMBER (AS LISTED AFTER TYPING'
        1,' PRINT) CORRESPONDING TO',/,' THE LINE YOU WISH TO DELETE'
        2,'. TYPE STOP WHEN FINISHED.')
        RETURN 1
290     TYPE 300
300     FORMAT(' THE TITLE WILL APPEAR ON THE GRAPH. YOU CAN SELECT'
        1,' SELECT THE LOCATION',/,' RELATIVE TO THE AXES AND'
        2,' MORE THAN ONE LINE MAY BE USED.')
        RETURN 1
        END
C
C
C
C       LABEL THE LAST LINE GENERATED
        SUBROUTINE LABL($,MATCH)
        COMMON LINES,DATA(26,2,10),POINT(10),LABEL(10,2)
        INTEGER POINT
        MATCH=1
10      TYPE 20
20      FORMAT(' LABEL (MAXIMUM 10 CHARACTERS):',$)
        ACCEPT 30,LABEL(LINES,1),LABEL(LINES,2)
30      FORMAT(2A5)
40      TYPE 50,LABEL(LINES,1),LABEL(LINES,2)
50      FORMAT(1X,2A5,5X,'IS THE LABEL CORRECT?',$)
        ACCEPT 60,ANS
60      FORMAT(A5)
        IF(ANS.EQ.'NO') GO TO 10
        IF(ANS.NE.'YES') GO TO 40
        IF(LINES.NE.1) TYPE 70,LINES
70      FORMAT(' THERE ARE NOW ',I2,' LINES ON THE GRAPH.')
        IF(LINES.EQ.1) TYPE 80
80      FORMAT(' THERE IS NOW 1 LINE ON THE GRAPH.')
        RETURN 1
        END
C
C
C
```

```
C       LIST THE PARAMETERS AND THE CURRENT VALUE OF EACH
        SUBROUTINE LIST($,MATCH)
        COMMON LINES,DATA(26,2,10),POINT(10),LABEL(10,2),WIDTH(2)
        1,YLABEL(46),XLABEL(2),LIMIT,ICOMP,X,NCAR,ATTRAT,MATINF
        2,LABINF,LABORS,LOW,HIGH,PAYSTP,FRAC,FRIDT,TAXRAT,UPAYRT
        3,SCHED(49),SNSUP,OINF,METHOD,LIFE,TECH,SAVL(50),OSAV(50)
        4,IDEX1,IDEX2
        REAL NCAR,MATINF,LABINF,LABORS
        INTEGER POINT,YEAR,HIGH,LOW,PAYSTP
        MATCH=1
        TYPE 10
10      FORMAT(' VARIABLE',T37,'KEYWORD',T55,'CURRENT VALUE')
        TYPE 20,LIMIT,ICOMP,NCAR,ATTRAT,FRAC,FRIDT,TAXRAT,UPAYRT,TECH
20      FORMAT(' NUMBER OF YEARS IN ANALYSIS',T38,'LIMIT',T63,I2,/
        1,' YEARS BEFORE SYSTEM IS COMPATIBLE',T38,'COMPATIBLE',T63,I2,/
        2,' NUMBER OF CARS',T38,'NUMBER',T55,F14.3,/
        3,' ATTRITION RATE',T38,'ATTRITION',T55,F14.3,/
        4,' NEW COST OF EQUIPMENT',T38,'FRACTION',T55,F14.3,/
        5,' INVESTMENT TAX CREDIT',T38,'INVESTMENT',T55,F14.3,/
        6,' TAX RATE',T38,'TAX',T55,F14.3,/
        7,' LOSS TO UNION',T38,'UNION',T55,F14.3,/
        8,' EXPECTED LIFETIME OF THE TECHNOLOGY',T38,'TECH',T55,F14.3)
        IF(IDEX1 .EQ. 0)  GO TO 27
        TYPE 21
21      FORMAT(' LABOR SAVINGS',T38,'SAVINGS')
        DO 22 I=1,LIMIT
        TYPE 23,I,SAVL(I)
22      CONTINUE
23      FORMAT(T8,' YEAR',I3,T55,F14.3)
        TYPE 24,PAYSTP
24      FORMAT(' YEARS SAVINGS ARE LOST TO UNION',T38,'LOSE',T63,I2,/
        1,' OTHER SAVINGS',T38,'OTHER')
        DO 25 I=1,LIMIT
        TYPE 26,I,OSAV(I)
25      CONTINUE
26      FORMAT(T8,' YEAR',I3,T55,F14.3)
        GO TO 29
27      TYPE 28,LABORS,PAYSTP,SNSUP
28      FORMAT(' LABOR SAVINGS',T38,'SAVINGS',T55,F14.3,/
        2,' YEARS SAVINGS ARE LOST TO UNION',T38,'LOSE',T63,I2,/
        7,' OTHER SAVINGS',T38,'OTHER',T55,F14.3)
29      TYPE 30,MATINF,LABINF,OINF
30      FORMAT(' INFLATION:',T38,'INFLATION',/
        4,T8,'MATERIALS',T55,F14.3,/
        5,T8,'LABOR',T55,F14.3,/
        6,T8,'OTHER',T55,F14.3)
        TYPE 40,LOW,HIGH,LIFE
40      FORMAT(' MINIMUM INTERNAL RATE OF RETURN',T38,'RATES',T63,I2,/
        1,' MAXIMUM INTERNAL RATE OF RETURN',T38,'RATES',T63,I2,/
        2,' DEPRECIATION',T38,'DEPRECIATION',/
        3,T8,' LIFETIME OF ASSETS',T63,I2,/,8X,$)
        IF(METHOD.EQ.1) TYPE 50
        IF(METHOD.EQ.2) TYPE 60
        IF(METHOD.EQ.3) TYPE 70
50      FORMAT(' STRAIGHT LINE',$)
```

```
60      FORMAT(' DOUBLE DECLINING BALANCE',$)
70      FORMAT(' SUM OF YEARS DIGITS',$)
        TYPE 80
80      FORMAT(' DEPRECIATION USED.')
        TYPE 90,XLABEL(1),XLABEL(2),(YLABEL(I),I=1,46)
90      FORMAT(' THE AXES ARE LABELED AS FOLLOWS:',/
       1,' X-AXIS: ',2A5,/,' Y-AXIS: ',46A1,/,20X,'KEYWORD IS AXES')
        RETURN 1
        END
C
        SUBROUTINE LOOK(IMATCH,ANS,LEVEL)
        COMMON LINES,DATA(26,2,10),POINT(10),LABEL(10,2),WIDTH(2)
       1,YLABEL(46),XLABEL(2),LIMIT,ICOMP,X,NCAR,ATTRAT,MATINF
       2,LABINF,LABORS,LOW,HIGH,PAYSTP,FRAC,FRIDT,TAXRAT,UPAYRT,
       3SCHED(49),SNSUP,OINF,METHOD,LIFE,TECH,SAVL(50),OSAV(50)
       4,IDEX1,IDEX2
        REAL NCAR,MATINF,LABINF,LABORS
        INTEGER POINT,YEAR,HIGH,LOW,PAYSTP
        MATCH=1
        IF(ANS.EQ.'CREAT') CALL CRE($10,IMATCH)
        IF(ANS.EQ.'GRAPH') CALL GRA($10,IMATCH)
        IF(ANS.EQ.'CHANG') CALL CHANG($10,IMATCH)
        IF(ANS.EQ.'DELET') CALL DELET($10,IMATCH)
        IF(ANS.EQ.'PRINT') CALL PRINT($10,IMATCH)
        IF(ANS.EQ.'SOLVE'.AND. IDEX1 .EQ. 0) CALL SOLVE($10,IMATCH)
        IF(ANS.EQ.'SOLVE'.AND. IDEX1 .EQ. 1) CALL VARIA($10,IMATCH)
        IF(ANS.EQ.'SOLVE') CALL LABL($10,IMATCH)
        IF(ANS.EQ.'HELP') CALL HELP($10,IMATCH,LEVEL)
        IF(ANS.EQ.'LIST') CALL LIST($10,IMATCH)
        IF(ANS.EQ.'TRACE') CALL TRACE
        IF(ANS.EQ.'NULL') IMATCH=1
        IF(ANS.EQ.'MODEL') CALL MODEL($10,IMATCH,LEVEL)
        RETURN
10      ANS='NULL'
        IMATCH=1
        RETURN
        END
C
C
C
C       CONSTRUCT A SET OF PARAMETERS FROM SCRATCH
        SUBROUTINE MODEL($,MATCH,LEVEL)
        COMMON LINES,DATA(26,2,10),POINT(10),LABEL(10,2),WIDTH(2)
       1,YLABEL(46),XLABEL(2),LIMIT,ICOMP,X,NCAR,ATTRAT,MATINF
       2,LABINF,LABORS,LOW,HIGH,PAYSTP,FRAC,FRIDT,TAXRAT,UPAYRT,
       3SCHED(49),SNSUP,OINF,METHOD,LIFE,TECH,SAVL(50),OSAV(50)
       4,IDEX1,IDEX2
        REAL NCAR,MATINF,LABINF,LABORS
        INTEGER POINT,YEAR,HIGH,LOW,PAYSTP
        MATCH=1
```

```
C       TYPE HEADING
        TYPE 1
1       FORMAT(' THIS IS A MODEL TO ESTIMATE THE AMOUNT THAT CAN',/
        1,' BE SPENT PER FREIGHT CAR FOR ADVANCED BRAKING',/
        2,' AND COUPLING.')
        GO TO 21
C
C       ERROR PROCEDURE
10      REREAD 12,ANS
        IMATCH=0
        CALL LOOK(IMATCH,ANS,LEVEL)
12      FORMAT(A5)
        IF(IMATCH.EQ.0) TYPE 11
11      FORMAT(' PLEASE USE ONLY 1 TO 9,0,+,-,. IN YOUR RESPONSE.',/
        1,' TYPE HELP FOR MORE INFORMATION.')
        GO TO (21,21,21,21,22,23,24,25,26,27,28,29,30,31,32,33)
        1,(LEVEL+1)
        TYPE 5,LEVEL
5       FORMAT(' THERE HAS BEEN AN ERROR. LEVEL=',I2)
        RETURN 1
C
C       GO TO THE NEXT QUESTION
21      GO TO 400
22      GO TO 430
23      GO TO 480
24      GO TO 520
25      GO TO 550
26      GO TO 650
27      GO TO 690
28      GO TO 630
29      GO TO 670
30      GO TO 610
31      GO TO 580
32      GO TO 720
33      GO TO 100
34      CALL LIST($35,IMATCH)
35      RETURN 1
C
C       COMPUTE DEPRECIATION SCHEDULE
100     TYPE 110
110     FORMAT(' WHAT IS THE DEPRECIATION LIFETIME OF THE ASSET?',$)
        LEVEL=15
        READ(5,120,ERR=10) LIFE
120     FORMAT(I3)
        IF(LIFE.GE.1) GO TO 140
        TYPE 130
130     FORMAT(' THE LIFETIME MUST BE ONE OR MORE. PLEASE REENTER.')
        GO TO 100
140     TYPE 150
150     FORMAT(' WHICH METHOD OF DEPRECIATION DO YOU WANT TO USE?',/
        1,T10,' STRAIGHT',T25,'-STRAIGHT LINE'
        2,/,T10,' DOUBLE',T25,'-DOUBLE DECLINING BALANCE'
        3,/,T10,' SUM',T25,'-SUM OF YEARS DIGITS',/,' METHOD:',$)
        ACCEPT 151,ANS
151     FORMAT(A5)
        IF(ANS.EQ.'STRAI') GO TO 170
        IF(ANS.EQ.'DOUBL') GO TO 190
        IF(ANS.EQ.'SUM') GO TO 240
C
```

```
C       PRINT ERROR MESSAGE SINCE ENTRY COULD NOT BE IDENTIFIED
        TYPE 160
160     FORMAT(' PLEASE TYPE STRAIGHT,DOUBLE, OR SUM')
        GO TO 140
C
C       STRAIGHT LINE METHOD
170     METHOD=1
        DO 180 I=1,49
        SCHED(I)=1.0/FLOAT(LIFE)
        IF(I.GT.LIFE) SCHED(I)=0.0
180     CONTINUE
        GO TO 300
C
C       DOUBLE DECLINING BALANCE METHOD
C
C       RATE OF DEPRECIATION IS TWICE THAT OF STRAIGHT LINE METHOD
190     METHOD=2
        PERC=2.0/FLOAT(LIFE)
        BALANC=1.0
        DO 220 I=1,48
        SCHED(I)=BALANC*PERC
220     BALANC=BALANC-SCHED(I)
        SCHED(49)=BALANC
        GO TO 300
C
C       SUM OF YEARS DIGITS METHOD
240     METHOD=3
        SUM=(LIFE**2+LIFE)/2
        DO 250 I=1,49
        SCHED(I)=FLOAT(LIFE-I+1)/SUM
        IF(I.GT.LIFE) SCHED(I)=0.0
250     CONTINUE
        GO TO 300
C
C       SHOW SCHEDULE
300     SUM=0.0
        TYPE 310
310     FORMAT(' YEAR  FRACTION WRITTEN OFF IN THAT YEAR')
        DO 330 I=1,49
        TYPE 320,I,SCHED(I)
320     FORMAT(2X,I2,5X,F5.3)
330     SUM=SUM+SCHED(I)
        TYPE 340,SUM
340     FORMAT(/,' TOTAL=',F5.3,//)
        GO TO 34
C
C       CHANGE TIME HORIZON
400     TYPE 410
410     FORMAT(' FOR HOW MANY YEARS SHOULD THE CASH FLOWS',
        1' BE CALCULATED?',$)
        LEVEL=3
        READ (5,420,ERR=10) LIMIT
420     FORMAT(I3)
        IF(LIMIT.LT.1) GO TO 450
        IF(LIMIT.GT.50) GO TO 450
        GO TO 22
C
```

```
C       CHANGE THE YEAR FLEET BECOMES COMPATIBLE
430     TYPE 440
440     FORMAT(' HOW MANY YEARS BEFORE THE SYSTEM BECOMES'
        1,' COMPATIBLE?',$)
        LEVEL=4
        READ (5,420,ERR=10) ICOMP
        IF(ICOMP.LE.0) GO TO 470
        IF(ICOMP.GE.50) GO TO 470
        GO TO 23
450     TYPE 460
460     FORMAT(' YEAR MUST BE BETWEEN 1 AND 50, PLEASE REENTER.')
        GO TO 400
470     TYPE 460
        GO TO 430
C
C       NUMBER OF CARS IN THE SYSTEM
480     TYPE 490
490     FORMAT(' HOW MANY CARS ARE IN THE SYSTEM?',$)
        LEVEL=5
        READ (5,500,ERR=10) NCAR
500     FORMAT(E10.0)
        IF(NCAR.GT.0.0) GO TO 24
        TYPE 510
510     FORMAT(' THERE HAS TO BE MORE THAN ZERO CARS.')
        GO TO 480
C
C       ATTRITION RATE
520     TYPE 530
530     FORMAT(' WHAT IS THE EXPECTED LIFETIME OF THE CARS? ',$)
        LEVEL=6
        READ (5,500,ERR=10) ATTRAT
        IF (ATTRAT .NE. 0.0) ATTRAT=1.0/ATTRAT
        IF(ATTRAT.GE.0.0.AND.ATTRAT.LE.1.0) GO TO 900
        TYPE 540
540     FORMAT(' THE EXPECTED LIFETIME OF THE CARS MUST BE',/,
        1 ' GREATER THAN OR EQUAL TO ZERO.')
        GO TO 520
C
C       ORIGINAL COST AS FRACTION OF NEW COST
550     TYPE 560
560     FORMAT(' WHAT FRACTION OF RETROFIT COST IS REQUIRED FOR',/
        1,' NEW PRODUCTION (PER CAR)?',$)
        LEVEL=7
        READ (5,500,ERR=10) FRAC
        XFRAC=FRAC*100.0
        TYPE 570,XFRAC
570     FORMAT(' FRACTION=',F6.1,'%')
        TYPE 571
571     FORMAT(' IS THIS CORRECT?',$)
        ACCEPT 151,XFRAC
        IF(XFRAC.NE.'YES') GO TO 550
        GO TO 26
C
```

```
C       FRACTION DEDUCTIBLE FOR INVESTMENT TAX CREDIT
580     TYPE 590
590     FORMAT(' WHAT FRACTION OF INVESTMENTS ARE DEDUCTIBLE FOR',/
        1,' INVESTMENT TAX CREDIT?',$)
        LEVEL=13
        READ (5,500,ERR=10) FRIDT
        IF(FRIDT.GE.0.0) GO TO 32
        TYPE 600
600     FORMAT(' FRACTION CANNOT BE LESS THAN ZERO.')
        GO TO 580
C
C       TAX RATE
610     TYPE 620
620     FORMAT(' WHAT IS THE TAX RATE FOR THE RAILROAD INDUSTRY?',$)
        LEVEL=12
        READ (5,500,ERR=10) TAXRAT
        IF(TAXRAT.GT.1) TAXRAT=TAXRAT/100.
        GO TO 31
C
C       UNION PAYOFF RATE
630     TYPE 640
640     FORMAT(' WHAT FRACTION OF LABOR SAVINGS ARE PAID'
        1,' TO THE UNION?',$)
        LEVEL=10
        READ (5,500,ERR=10) UPAYRT
        IF(UPAYRT.GT.1.) UPAYRT=UPAYRT/100.
        GO TO 29
C
C       SAVINGS SUBJECT TO UNION
650     TYPE 655
655     FORMAT(' TYPE YES IF YOU WANT TO ENTER LABOR SAVINGS',/
        1, ' THAT IS SUBJECT TO UNION PAYOFF ON A YEAR BY ',/
        2, ' YEAR BASIS.  TYPE NO OTHERWISE:  ',$)
        LEVEL=8
        ACCEPT 656,ANS
656     FORMAT(A5)
        IDEX1=0
        IF (ANS .EQ. 'NO')  GO TO 657
        IF (ANS .NE. 'YES') GO TO 650
        CALL NEW(SAVL,LIMIT,IDEX1)
        GO TO 27
657     TYPE 660
660     FORMAT(' WHAT IS THE LABOR SAVINGS PER YEAR THAT IS',/
        1,' SUBJECT TO UNION PAYOUT?',$)
        READ (5,500,ERR=10) LABORS
        GO TO 27
C
```

```
C       SAVINGS NOT SUBJECT TO UNION PAYOFF
670     TYPE 680
680     FORMAT(' TYPE YES IF YOU WANT TO ENTER SAVINGS',/
        1,' THAT IS NOT SUBJECT TO UNION PAYOFF ON A YEAR BY ',/
        2,' YEAR BASIS.',/
        1,' TYPE NO OTHERWISE:  ',$)
        LEVEL=11
        ACCEPT 656,ANS
        IF(ANS .EQ. 'NO') GO TO 673
        IF(ANS .NE. 'YES') GO TO 670
        CALL NEW(OSAV,LIMIT,IDEX2)
        IF(IDEX1.EQ.1)GOTO 30
        IDEX1=1
        DO 671 I=1,ICOMP
671     SAVL(I)=0.0
        J=1+ICOMP
        DO 672 I=J,LIMIT
672     SAVL(I)=LABORS
        GOTO 30
673     TYPE 681
681     FORMAT(' WHAT IS THE ANNUAL SAVINGS NOT SUBJECT TO UNION',/
        1, ' PAYOUT? ',$)
        READ (5,500,ERR=10) SNSUP
        IF(IDEX1.EQ.0)GO TO 30
        IDEX2=1
        DO 682 I=1,ICOMP
682     OSAV(I)=0.0
        J=ICOMP+1
        DO 683 I=J,LIMIT
683     OSAV(I)=SNSUP
        GOTO 30
C
C       STOP PAYING OFF UNION
690     TYPE 700
700     FORMAT(' FOR HOW MANY YEARS WILL SAVINGS BE PAID TO THE UNION?'
        1,$)
        LEVEL=9
        READ (5,420,ERR=10) PAYSTP
        I=LIMIT-ICOMP
        IF(PAYSTP.GE.0.0.AND.PAYSTP.LE.I) GO TO 28
        TYPE 710,LIMIT
710     FORMAT(' THERE MUST BE BETWEEN ZERO AND ',I2,' YEARS.')
        GO TO 690
C
```

```
C       INFLATION RATES
720     TYPE 730
730     FORMAT(' WHAT IS THE INFLATION RATE FOR:',/
        1,T10,' MATERIALS (IN PERCENT)?',$)
        LEVEL=14
        READ (5,500,ERR=10) MATINF
        MATINF=1+(MATINF/100.)
        TYPE 740
740     FORMAT('+',T10,' LABOR (IN PERCENT)?',$)
        READ (5,500,ERR=10) LABINF
        LABINF=1.+(LABINF/100.)
        TYPE 750
750     FORMAT('+',T10,' SAVINGS NOT SUBJECT TO UNION PAYOUT'
        1,' (IN PERCENT)?',$)
        READ (5,500,ERR=100) OINF
        OINF=1.+(OINF/100.)
        GO TO 33
C
C       THE EXPECTED LIFETIME OF THE TECHNOLOGY
900     TYPE 910
910     FORMAT(' WHAT IS THE EXPECTED LIFETIME OF THE TECHNOLGY',/
        1, ' BEING IMPLEMENTED ON THE RAILROAD SYSTEM? ',$)
        LEVEL=6
        READ (5,500,ERR=10) TECH
        IF(TECH.GT.0.0) GO TO 25
        TYPE 930
930     FORMAT(' THE EXPECTED LIFETIME OF THE TECHNOLOGY MUST BE',/,
        1 ' GREATER THAN ZERO.')
        GO TO 900
C
        END
C
C
C
C       PRINT THE CONTENTS OF DATA
        SUBROUTINE PRINT($,MATCH)
        COMMON LINES,DATA(26,2,10),POINT(10),LABEL(10,2),WIDTH(2)
        1,YLABEL(46),XLABEL(2),LIMIT,ICOMP,X,NCAR,ATTRAT,MATINF
        2,LABINF,LABORS,LOW,HIGH,PAYSTP,FRAC,FRIDT,TAXRAT,UPAYRT
        3,SCHED(49),SNSUP,OINF,METHOD,LIFE
        REAL NCAR,MATINF,LABINF,LABORS
        INTEGER POINT,YEAR,HIGH,LOW,PAYSTP
        MATCH=1
        DO 30 LINE=1,LINES
        TYPE 10,LINE,LABEL(LINE,1),LABEL(LINE,2)
        1,XLABEL(1),XLABEL(2),(YLABEL(I),I=1,46)
10      FORMAT(//,' LINE NUMBER: ',I2,T20,'LABEL: ',2A5
        1,/,8X,2A5,2X,46A1)
        DO 30 I=1,POINT(LINE)
        TYPE 20,I,DATA(I,1,LINE),DATA(I,2,LINE)
20      FORMAT(1X,I2,5X,F10.3,2X,F10.3)
30      CONTINUE
        RETURN 1
        END
C
C
```

```
C
C
C       GIVEN PARAMETERS FIND THE AMOUNT WHICH CAN BE SPENT PER
C       CAR FOR A VARIETY OF DISCOUNT RATES. STORE THE RESULTS
        SUBROUTINE SOLVE($,MATCH)
        DIMENSION A(50),B(50),START(50),ANNUAL(50),TAXCR(50)
        1,DEPRT(50),OTHER(50),SAVING(50)
        COMMON LINES,DATA(26,2,10),POINT(10),LABEL(10,2),WIDTH(2)
        1,YLABEL(46),XLABEL(2),LIMIT,ICOMP,X,NCAR,ATTRAT,MATINF
        2,LABINF,LABORS,LOW,HIGH,PAYSTP,FRAC,FRIDT,TAXRAT,UPAYRT
        1,SCHED(49),SNSUP,OINF,METHOD,LIFE,TECH,SAVL(50),OSAV(50)
        4,IDEX1,IDEX2
        REAL NCAR,MATINF,LABINF,LABORS
        INTEGER POINT,YEAR,HIGH,LOW,PAYSTP
        MATCH=1
C
C       COMPUTE CASH FLOWS FOR EACH YEAR, PER DOLLAR OF RETROFIT COST
        ISTP=PAYSTP+ICOMP
        TAXCR(1)=0.0
        IF(LINES.GE.10) GO TO 310
        LINES=LINES+1
        DO 100 YEAR=1,LIMIT
C
C       IF SYSTEM NOT COMPATABLE THERE IS A START-UP COST
        START(YEAR)=0.0
        IF(YEAR.LE.ICOMP) START(YEAR)=(1/FLOAT(ICOMP)-ATTRAT)*NCAR*
        1(MATINF**(YEAR-1))*-1
C
C       ANNUAL EXTRA COST OF ADV. BRAKING & COUPLING
        IF(YEAR.LE.TECH)ANNUAL(YEAR)=FRAC*ATTRAT*NCAR*
        1(MATINF**(YEAR-1))*-1
        IF(YEAR.GT.TECH)ANNUAL(YEAR)=FRAC*(1/TECH)*NCAR*
        1(MATINF**(YEAR-1))*-1
C
C       TAX CREDIT ONE YEAR AFTER INVESTMENT
        IF(YEAR.EQ.1) GO TO 40
        TAXCR(YEAR)=(START(YEAR-1)+ANNUAL(YEAR-1))*FRIDT*-1
C
C       DEPRECIATION TAX CREDIT
40      DEPRT(YEAR)=0.0
        IF(YEAR.LT.2) GO TO 55
        DO 50 I=1,YEAR-1
50      DEPRT(YEAR)=DEPRT(YEAR)-TAXRAT*(START(YEAR-I)+ANNUAL(YEAR-I))
        1*SCHED(I)
C
C       LABOR SAVINGS
55      SAVING(YEAR)=0.0
        UNION=1.0
        IF(YEAR.LE.ISTP) UNION=1.0-UPAYRT
        IF(YEAR.GT.ICOMP) SAVING(YEAR)=LABORS*(1.0-TAXRAT)*UNION
        1*(LABINF**(YEAR-1))
C
```

```
C       SAVINGS NOT SUBJECT TO UNION PAYOFF
        OTHER(YEAR)=0.0
        IF(YEAR.GT.ICOMP) OTHER(YEAR)=SNSUP*(1.-TAXRAT)*(OINF**(YEAR-1))
C
C       FIND SUM OF PER COST CASH FLOWS
        A(YEAR)=START(YEAR)+ANNUAL(YEAR)+TAXCR(YEAR)+DEPRT(YEAR)
C
C       FIND SUM OF FIXED FLOWS
        B(YEAR)=SAVING(YEAR)+OTHER(YEAR)
C
C       CASH FLOWS IN YEAR = AX+B WHERE X=COST OF RETROFITTING ONE CAR
100     CONTINUE
C
C       FIND PRESENT VALUE OF A AND B FOR ALL DISCOUNT RATES
        DO 300 I=LOW,HIGH
        SUMA=0.0
        SUMB=0.0
        R=1.0+FLOAT(I)/100.0
        DO 200 YEAR=1,LIMIT
        FACTOR=R**(YEAR-1)
        SUMA=SUMA+A(YEAR)/FACTOR
        SUMB=SUMB+B(YEAR)/FACTOR
200     CONTINUE
C
C       FILE RESULTS
        ROW=I-LOW+1
        DATA(ROW,1,LINES)=I
        XX=0.0-SUMB/SUMA
        DATA(ROW,2,LINES)=XX
        IF(I.EQ.12) XY=XX
300     CONTINUE
        POINT(LINES)=HIGH-LOW+1
C
C       COMPUTE PAYBACK PERIOD
        CUME=0.0
        DO 400 I=1,LIMIT
        CUME=CUME+XY*A(I)+B(I)
        IF(CUME.GE.0.0) GO TO 410
400     CONTINUE
        TYPE 405
405     FORMAT(' PAYBACK NOT REACHED.')
        RETURN
410     TYPE 420,I
420     FORMAT(' PAYBACK REACHED ',I2,' YEARS AFTER START-UP.')
        RETURN
        RETURN
310     TYPE 320
320     FORMAT(' DATA FILE IS FULL.')
        RETURN 1
        END
C
C
```

```
C
C       SOLVE WITH VARIABLE CASH FLOWS
        SUBROUTINE VARIA($,MATCH)
        DIMENSION A(50),B(50),START(50),ANNUAL(50),TAXCR(50)
        1,DEPRT(50),SAVING(50),OTHER(50)
        COMMON LINES,DATA(26,2,10),POINT(10),LABEL(10,2),WIDTH(2)
        1,YLABEL(46),XLABEL(2),LIMIT,ICOMP,X,NCAR,ATTRAT,MATINF
        2,LABINF,LABORS,LOW,HIGH,PAYSTP,FRAC,FRIDT,TAXRAT,UPAYRT
        1,SCHED(49),SNSUP,OINF,METHOD,LIFE,TECH,SAVL(50),OSAV(50)
        4,IDEX1,IDEX2
        REAL NCAR,MATINF,LABINF,LABORS
        INTEGER POINT,YEAR,HIGH,LOW,PAYSTP
        MATCH=1
C
C       COMPUTE CASH FLOWS FOR EACH YEAR, PER DOLLAR OF RETROFIT COST
        TAXCR(1)=0.0
        IF(LINES.GE.10) GO TO 310
        LINES=LINES+1
        DO 100 YEAR=1,LIMIT
C
C       IF SYSTEM NOT COMPATABLE THERE IS A START-UP COST
        START(YEAR)=0.0
        IF(YEAR.LE.ICOMP) START(YEAR)=(1/FLOAT(ICOMP)-ATTRAT)*NCAR*
        1(MATINF**(YEAR-1))*-1
C
C       ANNUAL EXTRA COST OF ADV. BRAKING & COUPLING
        IF(YEAR.LE.TECH)ANNUAL(YEAR)=FRAC*ATTRAT*NCAR*
        1(MATINF**(YEAR-1))*-1
        IF(YEAR.GT.TECH)ANNUAL(YEAR)=FRAC*(1/TECH)*NCAR*
        1(MATINF**(YEAR-1))*-1
C
C       TAX CREDIT ONE YEAR AFTER INVESTMENT
        IF(YEAR.EQ.1) GO TO 40
        TAXCR(YEAR)=(START(YEAR-1)+ANNUAL(YEAR-1))*FRIDT*-1
C
C       DEPRECIATION TAX CREDIT
40      DEPRT(YEAR)=0.0
        IF(YEAR.LT.2) GO TO 55
        DO 50 I=1,YEAR-1
50      DEPRT(YEAR)=DEPRT(YEAR)-TAXRAT*(START(YEAR-I)+ANNUAL(YEAR-I))
        1*SCHED(I)
C
C       LABOR SAVINGS
55      UNION=1.0
        IF(YEAR.LE.PAYSTP) UNION=1.0-UPAYRT
        SAVING(YEAR)=SAVL(YEAR)*(1.0-TAXRAT)*UNION
        1*(LABINF**(YEAR-1))
C
C       SAVINGS NOT SUBJECT TO UNION PAYOFF
        OTHER(YEAR)=OSAV(YEAR)*(1.0-TAXRAT)
        1*(OINF**(YEAR-1))
C
```

```
C       FIND SUM OF PER COST CASH FLOWS
        A(YEAR)=START(YEAR)+ANNUAL(YEAR)+TAXCR(YEAR)+DEPRT(YEAR)
C
C       FIND SUM OF FIXED FLOWS
        B(YEAR)=SAVING(YEAR)+OTHER(YEAR)
C
C       CASH FLOWS IN YEAR = AX+B WHERE X=COST OF RETROFITTING ONE CAR
100     CONTINUE
C
C       FIND PRESENT VALUE OF A AND B FOR ALL DISCOUNT RATES
        DO 300 I=LOW,HIGH
        SUMA=0.0
        SUMB=0.0
        R=1.0+FLOAT(I)/100.0
        DO 200 YEAR=1,LIMIT
        FACTOR=R**(YEAR-1)
        SUMA=SUMA+A(YEAR)/FACTOR
        SUMB=SUMB+B(YEAR)/FACTOR
200     CONTINUE
C
C       FILE RESULTS
        ROW=I-LOW+1
        DATA(ROW,1,LINES)=I
        DATA(ROW,2,LINES)=-1*SUMB/SUMA
300     CONTINUE
        POINT(LINES)=HIGH-LOW+1
        RETURN
310     TYPE 320
320     FORMAT(' DATA FILE IS FULL.')
        RETURN 1
        END

C
C
C
C
C       THIS SUBROUTINE ALLOWS THE USER TO INPUT
C       VARIABLE SAVINGS INFORMATION.
C
C
        SUBROUTINE NEW(SAV,YREND,IDEX)
       DIMENSION SAV(50)
        INTEGER YREND
        REAL LABORS

       IDEX=1
       J=1
C
C
10     TYPE 20
20     FORMAT(' TYPE THE END YEAR FOR THE RANGE TO ENTER SAVINGS: ',$)
       READ(5,30,ERR=200) K
30     FORMAT(I2)
       IF((K.GT.YREND).OR.(K.LT.J)) GOTO 200
C
```

```
C
40      TYPE 50
50      FORMAT(' DO YOU WANT TO ENTER VARIABLE SAVINGS FOR THIS RANGE? '
     1,     $)
        READ(5,60) ANS
60      FORMAT(A5)
        IF(ANS.EQ.'YES') GOTO 100
        IF(ANS.NE.'NO') GOTO 40
C
C
65      IFLG=1
        TYPE 70,J,K
70      FORMAT(' WHAT IS THE SAVINGS FROM YEAR ',I2,' TO YEAR ',
     1 I2,'? ',    $)
       READ(5,80,ERR=300) LABORS
80      FORMAT(E10.0)
        DO 90 I=J,K
90           SAV(I)=LABORS
        J=K+1
        IF(K.LT. YREND) GOTO 10
        RETURN
C
C
100     IFLG=2
        TYPE 110
110     FORMAT(T3,'YEAR',T20,'SAVINGS')
        I=J
120     TYPE 130,I
130     FORMAT(2X,I2,T20,$)
        READ(5,80,ERR=300)SAV(I)
        I=I+1
        IF(I.LE. K) GOTO 120
        J=K+1
        IF(K.LT. YREND) GOTO 10
        RETURN
C
C       ERRORS
C
200     REREAD 205, AND
205     FORMAT(A5)
        TYPE 210,J,YREND
210     FORMAT(' THE END YEAR FOR THE SAVINGS RANGE MUST BE BETWEEN'
     1          ,I2,'AND',I2)
        GOTO 10
C
C
300     REREAD 205, AND
        TYPE 310
310     FORMAT(' PLEASE USE ONLY 1 TO 9,0,+,-,. IN YOUR RESPONSE')
        GOTO (65,120) IFLG
C
        END
*^C
```

Railroad Financial Evaluation Model: Description and Computer Program Users' Manual (Final Report), US DOT, FRA, JA Kane, CE Waldman, 1981 -18-Economics & Finance