Modeling of Integrated Environmental Control Systems for Coal-Fired Power Plants

Technical Progress Report

To

Pittsburgh Energy Technology Center
U.S. Department of Energy
Pittsburgh, Pennsylvania 15236

From

Center for Energy and Environmental Studies
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

Prepared by

E.S. Rubin, Principal Investigator

October 1989

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

*John
Columbian*

RECEIVED
DOE/PC/79864
89 OCT 19 AM 10:29

MASTER

John
DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

Table of Contents

1 Introduction	1
2 Model Documentation	1
3 Model Applications	2
4 An Advanced User Interface	2
5 Future Activities	3
6 References	3
Appendix A: Draft User's Manual for the IECM	
Appendix B: Evaluation of the Fluidized Bed Copper Oxide Process Using a Probabilistic Engineering Model	
Appendix C: Prototype Screens for A User-Friendly Interface	

1 Introduction

This is the seventh quarterly report of DOE Contract No. DE-AC22-87PC79864, entitled "Modeling of Integrated Environmental Control Systems for Coal-Fired Power Plants." This report summarizes accomplishments during the period June 1, 1989 to September 30, 1989.

The general goal of this research project is to enhance, and transfer to DOE, a new computer simulation model for analyzing the performance and cost of environmental control systems for coal-fired power plants. Systems utilizing pre-combustion, combustion, or post-combustion control methods, individually or in combination, may be considered. A unique capability of this model is the probabilistic representation of uncertainty in model input parameters. This stochastic simulation capability allows the performance and cost of environmental control systems to be quantified probabilistically, accounting for the interactions among all uncertain process and economic parameters. This method facilitates more rigorous comparisons between conventional and advanced clean coal technologies promising improved cost and/or effectiveness for SO₂ and NO_x removal. Detailed modeling of several pre-combustion and post-combustion processes of interest to DOE/PETC have been selected for analysis as part of this project.

2 Model Documentation

The major effort of the current project involves preparation of user documentation for the Integrated Environmental Control Model (IECM) being developed for DOE/PETC. Analytical documentation of model components was provided in the final report of an earlier project [1], and in earlier quarterly reports for the current project. In order to operate the model, however, a User's Manual also is required, which is the task currently at hand.

As described in previous quarterly reports, the IECM utilizes the Demos software system developed at Carnegie Mellon University. User documentation for the IECM assumes a working knowledge of the Demos environment via a separate user's manual accompanying the software [2]. The user's manual now being developed for the IECM provides a step-by-step description of how the IECM operates and how each of its modules are run. Appendix A is a draft of the text that has been completed to date. The remaining chapters, still in preparation, will deal more extensively with each of the environmental control technologies currently included in the

integrated model. The documentation is designed for operation of the IECM on either an Apple (Macintosh II) or Sun (3/52) computer workstation. However, the Macintosh II version is emphasized as this is the configuration to be delivered to DOE/PETC as part of this project.

3 Model Applications

Part of our work during the past quarter involved additional testing and case studies employing the IECM. Appendix B is a technical paper presented at the September 1989 *Pittsburgh Coal Conference* in which the model was used to compare an innovative clean coal technology for combined SO_2/NO_x removal (the fluidized-bed copper oxide process) to a "conventional" plant design employing separate processes for SO_2 and NO_x removal (flue gas desulfurization and selective catalytic reduction). This paper illustrates how the IECM can be used to evaluate the potential risks, benefits and markets of new technology for environmental control of coal-fired power plants.

4 An Advanced User Interface

Another significant accomplishment of the past quarter was the design of a prototype system to allow the IECM to be used more easily, without the need to learn the Demos control language. This prototype takes advantage of recent advances in menu-driven computer software for the Macintosh systems. A series of interactive computer screens was formulated through which the user can configure a power plant, run the model, and obtain output results in tabular or graphical form simply by clicking on the computer "mouse," or by entering numerical data in a "user-friendly" format. Substantial flexibility in model operation is retained in this advanced software system, in contrast to earlier generations of PC-based menus which were relatively simple and inflexible.

Appendix C displays examples of the prototype screens that have been developed. A demonstration of the IECM framework employing these screens was presented at the August 1989 DOE Contractors' Meeting, and was well received. The full-scale implementation of an advanced user interface of this type is not within the scope of the current project, however, but it could readily be implemented as part of a follow-on activity. This would allow the model to be used more widely since it would eliminate the need for the computer expertise now required to run any computer-based modeling system.

5 Future Activities

Activities for the coming quarter will continue to focus on model documentation and transfer. Delivery of the current version of the model to DOE/PETC is planned for late October 1989, to be followed by delivery of the completed user's manual. The development of new performance and economic models for selected advanced coal cleaning processes also will get underway in the coming quarter.

6 References

1. Rubin, E.S., R.J. Marnicio, M. Henrion, M.J. Small and G.J. McRae. *Development of an Operational Level II Acid Deposition Assessment Model*. Final Report from Center for Energy and Environmental Studies, Carnegie Mellon University, Pittsburgh, PA, to Oak Ridge National Laboratory, Oak Ridge, TN, May 1986.
2. Henrion, M. and N. Wishbow. *Demos User's Manual: Version Three*. Carnegie Mellon University, Department of Engineering and Public Policy, Pittsburgh, PA, August 1987.

Table of Contents

1. Introduction	1
1.1 Model Overview	1
1.2 Computer Hardware Options	3
1.3 Scope and Organization of this Manual	4
2. Tutorial	5
2.1 Loading the IECM Model	6
2.2 Analyzing the IECM Hierarchy	12
2.3 Displaying and Specifying Input, Control Parameters and Variables	15
2.4 Specifying Vectors and Tables	18
2.5 Typical Demos Commands	21
2.6 Graphing and Probabilistic Analysis	25
2.7 Changing Coals and Coal Cleaning	32
2.8 Redefining Variables	37
2.9 Exiting and Saving the Model	40
3. Initialization and IECM Structure	42
3.1 Summary Tables and Conventions	45
3.2 Model Hierarchy and Layout of IECM	46
3.3 Initiecm.mod File	48
3.3.1 Initiecm Model	55
3.3.2 Indices Model	55
3.3.3 Configuration Model	56
3.3.4 Iecmfunction Model	56
3.3.5 Miscfunction Model	57
3.3.6 Gasfunctions Model	57
3.3.7 Solidsfunctions Model	59
3.3.8 Uncertfunctions Model	59
3.3.9 Enthalpy Model	61
3.3.10 Iecmconstants Model	61
3.3.11 Nodevices Model	62
3.4 Initiecm.petc File	62
3.4.1 Indices Model	62
3.4.2 Econfunctions Model	64
3.4.3 Nodevices Model and Plant Model	64
4. Basic Power Plant Module	65
4.1 Plant.Mod File	65
4.1.1 Basecontrols Model	65
4.1.2 CoalInputs Model	67
4.1.3 Basepltinputs Model	68
4.1.3.1 Unc_plant Model	69
4.1.4 Airemissions Model	69
4.1.5 Plant Model	70
4.1.6 Plantperf Model	70
4.1.7 Fuel_char Model	71
4.1.8 Efficiency Model	72
4.1.9 Airheater Model	72
4.1.10 Fluegas Model	74
4.1.11 Tempfluegas Model	76
4.1.12 Solidsfluegas Model	76
4.1.13 Fgsummary Model	77

4.1.14 Emissions Model	78
4.2 Plant.petc File	78
4.2.1 Coalinputs Model	78
4.2.2 Basepltinputs Model	80
4.2.2.1 Unc_plant Model	80
4.2.3 Reagents Model	80
4.2.3.1 Unc_reagents	81
4.2.4 Fuel_char Model	81
4.2.5 Plantperf Model	82
4.2.6 Plantecon Model	82
5. Economics Module	85
5.1 Economic Overview	85
5.2 Economics.petc File	86
5.2.1 Econfunctions Model	86
5.2.2 Econmodelinputs Model	88
5.2.2.1 Unc_econ Model	89
5.2.3 Economics Model	89
5.2.4 Fixcharge Model	89
5.2.5 Finalcostmodel Model	90
5.2.6 Finalryrmodel Model	91
6. References	92

List of Figures

Figure 1-1:	Selected air pollution control options for pulverized coal-fired boilers	2
Figure 1-2:	Some ways of expressing uncertainty	2
Figure 2-1:	Picture of the Screen before Starting Demos	8
Figure 2-2:	Picture of Macintosh Screen before Starting Demos	9
Figure 2-3:	Picture of Macintosh Screen after Starting Demos	9
Figure 2-4:	Picture of Macintosh Screen When Demos Needs the Name of the Project	10
Figure 2-5:	Picture of Macintosh Screen When Demos is Ready to Read in a New Model	10
Figure 2-6:	Hierarchy for the IECM Configured with a Wet FGD, ESP, and SCR.	14
Figure 2-7:	Contents of the File, Example.mod	24
Figure 2-8:	Picture of Screen after Graph Elecpc Mwg Command	28
Figure 2-9:	Picture of Screen after Graph Elecpc Mwg Command on the Macintosh	28
Figure 2-10:	Picture of Screen after Pdf Elecpc Command	29
Figure 2-11:	Picture of Screen after Pdf Elecpc Command on the Macintosh	29
Figure 2-12:	Picture of Screen after Cdf Elecpc Command	31
Figure 2-13:	Picture of Screen after Cdf Elecpc Command on the Macintosh Version.	31
Figure 2-14:	Contents of iec.coal File Created by the Fortran Coal Cleaning Program.	35
Figure 3-1:	Hierarchy Setup by Initiecm.mod	47
Figure 3-2:	Flue Gas Schematic Defined by Plant.mod	49
Figure 3-3:	Flue Gas Schematic for a Wet FGD and SCR Configuration	50
Figure 3-4:	Flue Gas Schematic for a Spray Dryer and SCR Configuration	51
Figure 3-5:	Flue Gas Schematic for the Copper Oxide Configuration	52
Figure 3-6:	Flue Gas Schematic for the NOXSO Configuration	53
Figure 3-7:	Flue Gas Schematic for the Electron Beam Configuration	54
Figure 3-8:	Hierarchy Setup by Initiecm.mod and Initiecm.petc	63
Figure 4-1:	Hierarchy of IECM created by Plant.mod after Initiecm.mod has loaded into Demos	66
Figure 4-2:	Schematic Diagram of the Air Preheater	73
Figure 4-3:	Hierarchy of IECM created by Plant.petc after Initiecm.mod and Initiecm.petc has been Loaded into Demos	79
Figure 5-1:	Hierarchy Created by Economics.petc after Initiecm.mod and Initiecm.petc have been Loaded into Demos	87

Appendix A: Draft User's Manual for the IECM

DRAFT

Not for Citation or
Attribution

**User's Manual for the
Integrated Environmental Control Model
(IECM)**

John S. Salmento

Center for Energy and Environmental Studies
Carnegie Mellon University
Pittsburgh, PA 15213

October 1989

1. Introduction

1.1 Model Overview

For the past several years, the U.S. Department of Energy (DOE) has been supporting the development of the Integrated Environmental Control Model (IECM) for analyzing integrated environmental control system designs for new coal-fired power plants. It emphasizes the role of advanced technologies for pre-combustion, combustion, and post-combustion control of air pollutant emissions, and the systematic comparison of conventional and advanced integrated system designs based on site-specific plant characteristics. It also facilitates the explicit characterization and analysis of technical and economic uncertainties in model parameters and results.

The IECM currently simulates the performance of 16 pre-combustion, combustion, and post-combustion processes (Figure 1-1), and allows integrated systems to be configured by combining any desired combinations of these options. Each of the boxes in Figure 1-1 represents a module in the IECM model. The technologies above the dashed line represent conventional designs, while the technologies below the dashed line represent advanced designs. The conventional technologies include various levels of coal preparation for pre-combustion, low NO_x burners for combustion modifications, and a variety of post-combustion control measures for the control of particulates, sulfur dioxide, and NO_x . Advanced physical coal cleaning processes include high gradient magnetic separation (HGMS), coal-pyrite flotation and various heavy media separation processes. Other advanced processes that have been modeled to date are furnace sorbent injection, electron beam, NOXSO, and the copper oxide process.

A unique feature of this model is that it was written in the Demos language recently developed by Henrion and his co-workers [1] to facilitate the testing of alternative models and the analysis of uncertainty. Demos is run on computers using the Unix operating system. A more recent version is designed to operate on the Apple Macintosh computer line. Unlike conventional programming languages such as Fortran, Demos employs a nonprocedural format which allows model relationships to be specified or altered quite easily. Thus, modules for individual components can be developed and easily linked to test alternative system configurations. In addition to producing point estimates (deterministic values) of system performance and costs for a particular set of assumptions, the model also can produce probabilistic results based on an explicit characterization of the uncertainty in model parameters.

Figure 1-2 shows some of the ways in which this uncertainty may be quantified. The curves shown are Probability Distribution Functions (PDF). Which show the probability of a parameter having a specified value. A normal (Gaussian) distribution reflects a symmetric (but varying) probability of a parameter value being above or below the mean value (Figure 1-2a). A characteristic of this distribution is the standard deviation, σ . The properties of a Gaussian distribution are such that there is an 68% probability that a value lies within one standard deviation of the mean, and a 95% probability that a value lies within 2σ of the mean. In contrast, for a uniform distribution (Figure 1-2b) there is an equal likelihood of a value lying anywhere within a specified range. A lognormal distribution, on the other hand, is skewed (Figure 1-2c). This distribution is used when the lower bound is more certain than the upper bound. A user-specified distribution also may be employed by assigning probabilities to various fractiles across a range or to discrete point (Figure 1-2d).

Figure 1-1: Selected air pollution control options for pulverized coal-fired boilers

Figure 1-2: Some ways of expressing uncertainty

Given a specified set of uncertainties in model parameters, a form of Monte Carlo simulation using Latin Hypercube sampling is used to obtain a probabilistic description of output results by randomly sampling from the specified distributions. Thus, a range of output results is achieved which describes statistically the possible effects of simultaneous variations in many parameters. A cumulative frequency distribution then can be generated which characterizes the probability of some result (e.g., cost) exceeding a specified value, or lying within some specified range. Such results are much more comprehensive than the deterministic results of most modeling studies. As a way of characterizing uncertainty, it also offers an advantage over sensitivity analysis, which typically treats variations in only one or two parameters at a time, or establishes bounds on some result with no information as to its likelihood.

1.2 Computer Hardware Options

Demos is capable of running on any Sun 3 computer running the Unix operating system or any Macintosh computer running system 6.0, which will be described in more detail below. At Carnegie Mellon University, Demos is currently being used on MacIIcx and Sun 3/52 workstations. These workstations have large, high resolution graphic monitors, 70 megabyte disk drives or larger, and window managing software. A window manager allows the screen to be divided into different sections, which act as individual graphic screens. The Sun 3/52 uses the Andrew window manager, which is being developed and tested at CMU for IBM. A workstation configuration allows Demos to produce high resolution graphics on the screen and allows multiple programs to be run simultaneously. With a 70 Megabyte disk drive on the Sun 3, only about 10 to 15 Megabytes of space is left once all the software is on the disk. So an additional 70 Megabyte disk or one 140 Megabyte disk is recommended.

Demos can also be run from a regular terminal that is connected to the Sun 3 workstations; however, only alphanumeric graphs are shown on the terminal. If an IBM PC with a graphics card and color monitor is available, then color graphs can be displayed on its screen.

There are a number of ways to print the graphs produced by Demos on the Sun 3/52. Demos has some built in commands which create files which can be printed on any Postscript printer, such as the Apple Laserwriter. The Andrew window manager can take a snapshot of the screen, which can be printed out to any Postscript or IBM 3812/3820 printers. If an IBM PC is used then, the print screen key can be used to print out the graphs to any dot matrix printer connected to the IBM PC.

For the Macintosh, only two floppy drives and one and half megabytes of RAM memory is required to run Demos for simple models; however, eight megabytes of memory and an 80 megabyte disk drive are recommended. Demos can take advantage of the color monitors connected to a Macintosh. Also, it can print out directly to any Apple printer, or with the proper software and hardware Demos can print to most printers. Two major differences between the Unix version of Demos and the Macintosh version, are that most of the Demos commands and functions are accessible in pull-down menus. Demos will put up a little window whenever it needs information in the Macintosh version.

1.3 Scope and Organization of this Manual

This manual is intended to allow anyone with a basic understanding of Unix or the Macintosh operation system to use the IECM. Chapter 2 provides a tutorial for beginners and a basic description of the IECM structure. Further details about the Demos language can be found in Reference [1]. It is assumed here that the IECM user has reviewed the Demos User's Manual and has at least some familiarity with the structure of Demos variables and commands. People who have a good understanding of Demos can skip the Tutorial chapter. Chapter-3 provides a detailed description of the IECM hierarchy, including inputs, outputs, and various intermediate parameters. Analytical documentation for the IECM can be found in Rubin et. al. [2]. The latest analytical documentation for the Copper Oxide process can be found in Frey [3]. Chapter 16 contains documentation on the conventional coal cleaning and HGMS preprocessors.

2. Tutorial

This chapter is a tutorial on how to use the IECM. It shows how to use the Demos language for a typical analysis, and discusses many common errors that may be encountered. This tutorial provides hints on how to use the Andrew window manager and Macintosh operating systems, but it assumes that the user already is familiar with the operating systems and knows how to use a mouse. Where specific differences occur between operating systems, they will be distinguished. The following will be demonstrated in this tutorial:

- Loading a specific configuration into the IECM
- Analyzing the hierarchy of the IECM structure
- Specifying input parameters and variables
- Specifying vectors and tables
- Basic Demos commands
- Probabilistic analysis and graphing
- Changing coals and coal cleaning
- Redefining variables
- Exiting and saving the current configuration

A typical configuration of the IECM contains about 900 objects. Objects can be commands, functions, models, or variables. Commands can be divided into four groups: inspection, file, graphical, and miscellaneous. Inspection commands are used to display and analyze other objects. File commands are used to create and manipulate files. Graphing commands are used to create and modify graphs. Miscellaneous commands including quitting, help, etc. The names of objects are limited to twelve characters.

Variables	A generic model parameter, which may be represented in term as of other objects via algebraic relationships. For example, $HR_{cycle} = HR_{steam} / Boiler_{eff}$.
Models	Objects which group other objects for organization and structure. For example the model Wetscresp contains the models Allinputs, Plant, Esp, Wetfgd, Solwaste, Scr, Economics, Iecmf functions.
Functions	Objects which are similiar to functions in other languages, that is they are algorithms which can have parameters pass into them and return a value. There are two classes of functions, system functions and user-defined functions. For example, $Quadeqn(A,B,C) = (-b - \sqrt{b^2 - 4 * a * c}) / (2 * a)$, returns a root of a quadratic equation with the coefficients A, B, C.

The value of function and variables can be scalars, character strings, arrays, or probabilistic. Any of these values can be entered as an input parameter or calculated by Demos. In addition, it is very easy to convert between scalars, arrays, and probabilistic values. A brief description is given below:

Scalar	A scalar is a single number. For example, $Mwg = 500$.
Character string	A character string is a list of characters. For example, $Disposal = 'landfill'$. The single quotes are used to enter character strings.

- Arrays** Arrays can be single or multi dimensional and consist either of scalars or character strings. Most functions and commands in Demos operate on numeric arrays as easily as on scalars. This allows changing the value of a variable to an array, without changing the definitions of any variable that depend upon it, which is very useful for parametric analysis. Single dimensional arrays are called vectors, while multi dimensional arrays are called tables. An example of a vector of character strings is `gascomp = ['N2','O2','H2O','CO2','CO','SO2','SO3','NO','NO2']`. The square brackets are used to define arrays. An example of a table which contains the coal characteristics is `CoalChar_table = table(Coalindex,Fuelchar_index) (10194. .5695 .0370 .0719 .0436 .0112 .1438 .1230)`
- Probabilistic** A probabilistic value is an array of scalars generate by either Latin Hypercube or Monte Carlo sampling. The size of the array is controlled by the system variable, `samplesize`. Probabilistic values are generated when one or more probability density functions (normal, lognormal, uniform, fractile, and chance) are used in the definitions of variables. Demos has a number of commands for analyzing and graphing probabilistic values. Some of these commands are described in section 2.6.

Variables are the basic building blocks of any model and do not have any restrictions on their type of value. In other words variables can be scalars, character strings, arrays, or probabilistic. However many times it is useful to constrain the value of input parameter. Demos has five special classes of variable which have restrictions on their value. Of these five, the IECM uses the following three:

- Constant** A variable that has a scalar value, such as the universal gas constant, `gasR = 1545 ft-lbf/lb mole R`.
- Decision** A variable that is used to make a choice, such as `Furnacetype = 'wall'`. The only restriction on these variables is that they can not be probabilistic.
- Index** A variable which must be a single dimensional array. For example, the variable with the labels for the coal composition table, `Fuelchar_index = ['HHV', 'Carbon', 'Hydrogen', 'Oxygen', 'Sulfur', 'Nitrogen', 'Ash', 'Moisture']`.

The following conventions will be used in this example. Anything that has to be typed by the user will appear in bold. A "Return" key has to be pressed to enter a command; however, it will not be explicitly shown in the example for clarity. Many times Demos will prompt the user with questions, which have a default answer in brackets. The user can either enter an answer or type a "return" for the default. For the Unix version, it is important to enter the proper case for the letters in a file name since Unix is case sensitive. Useful hints will appear in *italics* on the right side. Finally, this example does not show all the commands, shortcuts, or hints possible, but it is enough to start using the IECM productively. For more detail see the Demos User's Manual [1].

2.1 Loading the IECM Model

If you are on a Macintosh skip to the next paragraph. The starting point for the Unix version for this example is that the user, John Salmento, is logged on to a Sun 3 workstation, with a large graphic monitor, and has started the Andrew window manager. The screen should look similar

to Figure 2-1. Everything that occurs in the tutorial will appear in the long blank window on the left. Whenever a command causes a new window to appear, a figure of the entire screen will be shown. To start Demos, type Demos at the prompt.

```
js04>Demos
Initializing...
```

```
Welcome to Demos (version V -- Andrew/Unix, April, 1988)
Copyright (C) 1983 C-MU.
```

```
Do you want to start a New Model? [Yes] :
Name of Project? [Test] : Example
```

```
Declaring Model Example
Description: Example on how to use the IECM
Author: John Saimento
```

```
There is no new News
Type "Help" if needed.
```

```
Example>
```

The starting point for the Macintosh version for this example is that the Macintosh is turned on, and the user has opened up the Demos folder on the Applications volume. The screen looks like Figure 2-2. To start Demos, double click on the icon (picture) which is labeled Demos. Demos will create a window when it is started, as shown in Figure 2-3. The biggest difference between the Macintosh version and the Andrew versions is that the Macintosh version will display a little window whenever Demos needs some information. For example, in Figure 2-3, Demos is asking if you want to start a new model. Move the mouse pointer to the "yes" button and click the button on the mouse once. Then Demos will open another window requesting the name of the project. Just type in **Example**. Figure 2-4 shows the result after typing in **Example**. Finally, Demos displays a window requesting a description and author for the project. Just type in the information that appears above for these two windows. Figure 2-5 shows the screen when Demos is ready to read in a new model.

The first step is to specify the power plant configuration to be analyzed. While the focus of the IECM is on air pollution control systems, the performance and cost for a complete power plant are estimated since the effects of coal cleaning (if used) can only be properly accounted for by looking at its net effect on power production costs. Thus, the elements of the IECM power plant design which must be specified are:

- Base power plant
- Particulate control system
- SO₂ control system
- NO_x system
- Solid waste disposal system

In addition, various economic and financial parameters must be specified, as well as coal properties, plant design and engineering parameters, and environmental regulatory constraints.

The structure of the IECM uses a series of files to organize information about individual plant

Figure 2-1: Picture of the Screen before Starting Demos

Figure 2-2: Picture of Macintosh Screen before Starting Demos

Figure 2-3: Picture of Macintosh Screen after Starting Demos

Figure 2-4: Picture of Macintosh Screen When Demos Needs the Name of the Project

Figure 2-5: Picture of Macintosh Screen When Demos is Ready to Read in a New Model

components and other necessary data. To configure and analyze a plant, the appropriate series of files is read and loaded into the model (details on each of these files appears later). These files contain various models arranged in a hierarchy, which allows a user to easily examine and find variables. For example, the file "wetfgd.mod" models the performance of a wet scrubber and consists of the main model wetfgd. The model wetfgd consists of three models, wetfgdlink, wetmodelinput, and wetperf. The economic algorithms for the wet scrubber are located in the file "wetfgd.petc." Many times the term, module, is used with the name of a particular device, such as "wetfgd module." This term refers to all the variables, models, and files which describe the performance and economics of that device. So "wetfgd module" refers to both "wetfgd.mod" and "wetfgd.petc."

The IECM also has several files which make it easy to load standard plant configurations involving different combinations of control technology. These files simply contain a series of Demos commands calling the various plant components. For example, the contents of "loadscrwet.mod" is shown below.

Model wetscresp	<i>Gives a name to the top model in the hierarchy.</i>
Description: Configuration with an ESP, wet FGD, and SCR.	
verb:2	<i>Prevents extremely verbose variable checking.</i>
	<i>"srccom/" is one directory with some of the IECM source code.</i>
	<i>On the Macintosh the "srccom/" would be a "files:srccom:"</i>
read srccom/initiecm.mod	
read srccom/plant.mod	
read srccom/esp.mod	
read srccom/wetfgd.mod	
read srccom/solwaste.mod	
	<i>"srcpetc/" is another directory with additional source code.</i>
	<i>On the Macintosh the "srcpetc/" would be a "files:srcpetc:"</i>
read srcpetc/initiecm.petc	
read srcpetc/plant.petc	
read srcpetc/scr.petc	
read srcpetc/esp.petc	
read srcpetc/wetfgd.petc	
read srcpetc/solwaste.petc	
read srcpetc/economics.petc	
Close wetscresp	<i>The following commands are optional.</i>
heapsize:10M	<i>Sets the amount of memory that Demos can use; not applicable on the Macintosh</i>
tabwidth:20	<i>Width of columns in tables, can have a value between 1 and 20.</i>
numberwidth:5	<i>Width of numbers printed to the screen.</i>

A list of the files specifying common plant configurations is given below, while a complete description is provided in Chapter 3. These files are located in the "loadfiles" directory.

- loadcuo.mod
- loadbeam.mod
- loaddry.mod
- loadnoxso.mod
- loadscrwet.mod
- loadscrdry.mod
- loadwet.mod

For this example, an analysis of a coal-fired power plant with a wet FGD, a cold-side ESP, and a SCR device will be demonstrated, so the file "loadscrwet.mod" (which specifies the elements for this configuration) needs to be loaded. Each hash mark represent 10 lines of source code.

Example>read loadfiles/loadscrwet.mod

*On the Macintosh, use the Read Model command in the File menu,
or type read files:iec:loadfiles:loadscrwet.mod*

Reading from loadwet.mod

*On the Macintosh, the characters "srccom" would be replace with
"files:iec:srccom:".*

Reading from srccom/initiecm.mod

#####

Reading from srccom/plant.mod

#####

Reading from srccom/esp.mod

#####

Reading from srccom/wetfgd.mod

#####

Reading from srccom/solwaste.mod

#####

*On the Macintosh, the characters "srcpetc" would be replace with
"files:iec:srcpetc:".*

Reading from srcpetc/initiecm.petc

#####

Reading from srcpetc/plant.petc

#####

Reading from srcpetc/scr.petc

#####

Reading from srcpetc/esp.petc

#####

Reading from srcpetc/wetfgd.petc

#####

Reading from srcpetc/solwaste.petc

####

Reading from srcpetc/economics.petc

#####

The definitions of all Variables and Functions are OK

Example>

2.2 Analyzing the IECM Hierarchy

At this point the basic configuration for a wet FGD, with a cold ESP and SCR device has been loaded into Demos, and the user is ready to begin the analysis. The upper four levels of the hierarchy for this configuration are shown in Figure 2-6. All of the models shown in Figure 2-6 contain variables or other models. As can be seen the top model is Example, which contains the model Wetscresp. Wetscresp contains a model for all the devices in this configuration (Plant, Esp, Wetfgd, Solwaste, and Scr) and three other models, Allinpu, Initiecm, and Economics.

For now we are only interested in the Allinputs model, which contains all the input models for this configuration. These input models contain the input variables needed for their respective models. A complete list of input parameters can be found in the source code or in Appendix A.

Demos has two basic commands which are useful examining the hierarchy of models, Show and Isin. The Show command displays the description and content of any object. When used on a model it displays the description, its contents, the author, date created, and what other objects it needs. The Isin command displays the model which contains the current object. For example, Allinputs is inside of the model Wetscresp. The example below demonstrates the results of using the Show and Isin commands.

Example>show example

Model Example

Description: Example on how to use the IECM

Contains: [Model Wetscresp]

Author: John Salmento

Date: Thu May 11 11:27:01

Savedate: Thu May 11 11:27:01

Needs: [Model Wetscresp]

This attribute indicates that this model needs some objects in model Wetscresp.

Example>show wetscresp

Model Wetscresp

Description: Configuration with an ESP, wet FGD, and SCR

Contains: [Model Initiecm, Model Allinputs, Model Plant, Model Esp, Model Wetfgd, Model Solwaste, Model Scr, Model Economics]

Author: js04

Needs: [Model Initiecm, Model Allinputs, Model Plant, Model Esp, Model Wetfgd, Model Solwaste, Model Scr, Model Economics]

Example>show allinputs

Model Allinputs

Description: contains all the input models for all the submodels

Contains: [Model Basecontrols, Model Coalinputs, Model Basepltininput, Model Airemissions, Model Espmodelinpu, Model Wetmodelinpu, Model Swdmodelinpu, Model Reagents, Model Scrmodelinpu, Model Econmodelinp]

Author: js04

Date: Tue Jan 21

Needs: [Model Basecontrols, Model Coalinputs, Model Basepltininput, Model Airemissions, Model Espmodelinpu, Model Wetmodelinpu, Model Swdmodelinpu, Model Reagents, Model Scrmodelinpu, Model Econmodelinp]

Example>isin allinputs

Wetscresp

Example>show basepltininput

Model Basepltininput

Description: input parameters for base power plant

Contains: [Variable Mwg, Variable Hrsteam, Variable Cf, Variable Exair_boiler, Variable Exair_airhea, Variable Orgtemp_furn, Variable Orgtemp_econ, Variable Orgtemp_fgap, Variable Xpulverizers, Variable Xsteampumps, Variable Xfans, Variable Xtowers, Variable Xmisc, Variable Unacc_loss, Index Pltinputlabe, Index Pltvalue, Variable Plantinputs, Model Unc_plant, Variable Icfbp, Variable Oplaborrate, Variable Lc]

Author: js04

Needs: [Model Tempfluegas, Model Unc_plant]

Example>

Figure 2-6: Hierarchy for the IECM Configured with a Wet FGD, ESP, and SCR.

2.3 Displaying and Specifying Input, Control Parameters and Variables

To display any input parameter, variable, vector, or table, just type its name. If the variable does not depend on any arrays, the value will be displayed; however, if it does depend on arrays then the array dependency will be displayed. For these variables, the Mid command is used to display the value. Note that each module, (i.e., wetfgd, scr, etc.) has its own input parameters, whose names can be found in the source code, in Appendix A, or by using the Show command as displayed above. The default values of some input parameters and variables are shown below. To set up a new run the user simply specifies new values for any model parameter.

```
Example>mwg
Mwg: Gross Electrical Output (MW) = 540
```

```
Example>exair_airheater
Exair_airhea: Leakage in Air Preheater (fraction) = 0.19
```

```
Example>exair_airboiler      Response when a variable is incorrectly typed.
?
Undefined name
```

```
Example>exair_boiler
Exair_boiler: Excess Air at Boiler (fraction) = 0.2
```

```
Example>hrsteam
Hrsteam : Steam Cycle Heat Rate (Btu/kWh) = 7800
```

```
Example>cf
Cf : Capacity factor (fraction) = 0.65
```

```
Example>show airemissions
Model Airemissions
Description: model containing the input parameters for the ambient air,~
            emission constraints, and emission factors for furnace
Contains: [Variable Tai_air, Variable Pai_air, Variable Humdt, Variable
Noxef, Variable Fao, Variable So2ashret, Variable Fso, Variable Fracco,
Variable Cinrefuse, Variable Fracc, Variable Fracso2, Variable Fracno,
Variable Emisconso2, Variable Emisconsnox, Variable Fmisconsash, Index
Air_emislab, Index Airvalue, Variable Airemisinput]
Author: js04
```

```
Example>humdt
Humdt : Specific Humidity Ratio (fraction) = 0.018
```

```
Example>emisconso2
Emisconso2 : Emission Constraint SO2 (lb/MBtu) = 0.855
```

```
Example>coalchar_table      Input table with coal characteristics, which is in the model,
                             Coalinputs. Since Coalchar_tab depends upon the arrays,
                             Coalindex and Fuelchar_ind, those arrays are displayed, when
                             its name is typed. To get its value type mid coalchar_tab.
Coalchar_tab: Input Coal Characteristics =
Table(Coalindex, Fuelchar_ind)
```

```
Example>mid coalchar_tab      This shows the ultimate analysis of the coal.
Fuelchar_ind      HHV,      Carbon, Hydrogen, Oxygen, Sulfur,
Nitrogen, Ash, Moisture
Coalindex
1 [ 10.2K, 0.57, 0.037, 0.072, 0.044,
0.011, 0.144, 0.123]
..
Demos uses suffixes to denote scientific
notation. For example, the suffix K
stands for 103.
```

```
Example>raw_coal_cst
Raw_coal_cst: raw coal cost ($/wet ton) = Table(Cost_yr)
```

```
Example>mid raw_coal_cst
Cost_yr [ 1988
          26.1]
```

```
Example>ctrans
Ctrans : Transportation Cost ($/wet ton) = Table(Cost_yr)
```

```
Example>mid ctrans
Cost_yr [ 1988
          5.9]
```

Changing the value of input parameters is easy. The example below shows how to change the gross capacity to 600 MWg, the steam cycle heat rate to 8,000 Btu/kWh, and the capacity factor to 75%. The IECM uses the range checking feature built into Demos for all input parameters and some intermediate variables. This allows the IECM to catch many typographical errors and other mistakes. The example below shows the result of entering 6000 instead of 600 for the variable, MwG. Note that Demos does not issue warning until either the value of the variable is requested or Demos evaluates a variable which depends upon MwG.

```
Example>mwg:6000
```

The fact that Demos did not complain about the value entered does not mean that the value is okay. It just indicates that the syntax for the definition is okay.

This shows the type of warning Demos issues for range check errors. For the question, below just hit a return. Notice that Demos, prints out the variable and its attributes.

```
Example>mwg
Mwg : Gross electrical output (MW)
? Warning:

Variable Mwg fails to satisfy Check : (( 100<=Mwg) AND (Mwg<= 1000))
Would you like to stop to try and fix the problem? [Yes]:
Variable Mwg
Title: Gross electrical output
Units: MW
Description: gross electrical output of generator not including ~
            auxiliary power requirements.
Definition: 6000
Check: 100 <= mwg and mwg <= 1000
```

The line above is the source code that checks that MwG is between 100 and 1000 MW. Any variable can have a Check attribute.

```
Example>mwg:600
```

```
Example>hrsteam:8000
```

```
Example>cf:0.75
```

Changing the equation for variables is a little more complicated. It is demonstrated in Section 2.8; however, it should only be done after a person has a good understanding of the IECM structure and Demos.

There are several variables in the IECM which describe the technologies which are used for SO₂, NO_x, TSP, and coal cleaning. These are So2config, Noxconfig, Tspconfig, and Ccconfig,

and are in the model Configuration. By typing their names, it is possible to see which technologies have been loaded into the IECM. The value of these parameters should not be changed by the user, since they are used by the other models to set options.

```
Example>so2config
So2config      : SO2 Control Technology          =      Wet FGD
```

```
Example>noxconfig
Noxconfig      : NOx Control Technology          =              SCR
```

```
Example>tspconfig
Tspconfig      : Tsp Control Technology          =              ESP
```

```
Example>ccconfig
Ccconfig       : Coal Cleaning Technology        = No Coal Cl
```

There are also several decision variables which use character strings. They are used to control the type of firing in the furnace, Furnacectl; the type of waste disposal, Disposal; whether the flyash is mixed with the FGD sludge, Mixedctl; and to control the probability distribution functions, Ctl_global. These variables are found in the model, Basecontrols. The individual technologies may also use their own decision variables to set technology specific options. For example, the model Wetfgd uses the variable, Bpsopt (ByPaSs OPTions), to control the bypass around the FGD. To determine the valid options for these variables use the Show command. Also note that character strings are case sensitive. A complete listing appears in Appendix A.

```
Example>furnacectl
Furnacectl     : Furnace Firing Type              =      tangential
```

```
Example>disposal
Disposal       : Disposal Option Chosen           =      landfill
```

```
Example>show disposal
Decision Disposal
Title: Disposal Option Chosen
Description: Waste Disposal control variable, valid values are 'ponding' ~
            or 'landfill'
Definition: 'landfill'
Check: if so2config = 'Wet FDG' or so2config = 'No SC2 Technology' then ~
        (disposal = 'ponding' or disposal = 'landfill') else ~
        (disposal = 'landfill')
```

```
Example>mixedctl
Mixedctl       : Flyash Mixing Control            =              yes
```

```
Example>ctl_global
Ctl_global     : Uncertainty Control Parameter    =      all off
```

```
Example>bpsopt
Bpsopt        : Bypass Option Chosen              =      no bypass
```

Furnacetype has three valid settings, 'wall', 'tangential', and 'cyclone'. Disposal has two settings: 'landfill' and 'ponding'; however, it should be set to 'landfill' unless So2Config equals 'base' or 'wetfgd.' In these cases, it can be set to either value. Mixedctl can either equal 'yes' or 'no'. Ctl_global has three possible settings: 'all off', 'all on', and 'on'. 'all off' makes the analysis deterministic by setting all the uncertainty variables equal to 1. 'all on' makes the analysis probabilistic by setting all the uncertainty variables equal to their probabilistic distributions. 'on' allows the user to control the uncertainty variables by module. This will be

explained more in Section 3.3.8. Bpsopt controls has two possible values 'bypass' and 'no bypass'.

2.4 Specifying Vectors and Tables

One of Demos's unique features are vectors and tables. This feature allows related data to be organized as one variable and for tables to be created to display results. There are several key vectors and tables in the IECM and all of them are listed in Appendix A. The most important vectors are described below.

Coalchar_table is the input table which contains the ultimate analysis data for the coal being used. It is created with two vectors, Coalindex and Fuelchar_index. Fuelchar_index can be visualized to be the labels of the columns in Coalchar_table, while Coalindex are the labels for the rows. Fuelchar_index is a vector of character strings, which corresponds to the ultimate analysis. Fuelchar_index equals ['HHV', 'Carbon', 'Hydrogen', 'Oxygen', 'Sulfur', 'Nitrogen', 'Ash', 'Moisture']. The definition of Fuelchar_index should not be changed since many variables depend upon its current definition. Coalindex is a sequence of numbers from 1 to the number of rows in Coalchar_table. If there is one row in Coalchar_table, then Coalindex equals [1]. If there are two rows in Coalchar_table, then Coalindex equals [1 2]. When Coalchar_table has more than one row, it indicates that the coal has been washed with the Fortran coal cleaning program. The first row is always the raw coal, while the other rows contain the characteristics of the coal washed to various levels. The variable, Coals, is used to choose the set of data for analysis from Coalchar_table, by setting it equal to the row that you want. Currently Coals equals 1.

Any coal can be specified by the user by entering the ultimate analysis data into Coalchar_table; however, several coals are provided in files in the directory "coalfiles." The coal files have the extension "coal." More details about Coalchar_table and how to use coal cleaning are in Section 2.7 and Chapter 16.

```
Example>show coalindex
Index Coalindex
Description: Index for the table coalchar_tab. Defined by Fortran coal ~
cleaning model if the conventional coal cleaning model is used.
Definition: [1]
```

```
Example>show Fuelchar_index
Index Fuelchar_index
Description: Index of fuel characteristics table
Definition: ['HHV', 'Carbon', 'Hydrogen', 'Oxygen', 'Sulfur', 'Nitrogen', ~
'Ash', 'Moisture']
check: Fuelchar_index = array(Fuelchar_index, ['HHV', 'Carbon', 'Hydrogen', ~
'Oxygen', 'Sulfur', 'Nitrogen', 'Ash', 'Moisture'])
```

```
Example>show coalchar_table
Variable Coalchar_table
Description: Table containing all the coal characteristics to be used
Definition: table(Coalindex, fuelchar_index) (
10194. .5695 .0370 .0719 .0436 .0112 .1438 .1230)
```

```
Example>coals
Coals          : Coals chosen          -          1
```

The vector, Cost_yr, is used to set the year in which costs are reported. It can vary between

[1977] and [1989]. Since economic algorithms were developed from sources which assumed different year dollars, Cost_yr is used to scale costs for each module to a consistent mid-year using the Chemical Engineering cost indices. Changing Cost_yr to another year causes all of the economic costs to be scaled accordingly. Cost_yr is not restricted to integer values. If Cost_yr equaled 1987.5, it would be equivalent to December 1987 or January 1988 dollars. It also should be noted that almost all of the economic variables depend on Cost_yr.

```
Example>cost_yr
Cost_yr      : Years Dollars Reported      =
               [      1988]                Comment: 1988 is the current default year
                                           for reporting costs.

Example>tdcscr
Tdcscr       : Scr Total Direct Cost.      (M $)      = Table(Cost_yr)

Example>mid tdcscr
Cost_yr      :      1988
               [      13.28]

Example>cost_yr: [1989]      Changes Cost_yr to 1989. The "[]" are important.

Example>mid tdcscr
Cost_yr      :      1989
               [      13.95]
```

Each module has two or more summary tables. Tables which have the suffix, "inputs," after the module name show the input assumptions for that module. Tables which have the suffix, "char," show the important performance characteristics that are calculated by the model. Those tables which have the suffix, "econchar," show the economic variables which are calculated. A complete list of these tables can be found in Appendix A. These tables are used for display purposes only and depend upon other variables in the model. Partial listings of three tables for the plant module are shown below for illustration. For these tables, the Mid command is needed to display them.

```
Example>mid plantinputs      Partial list of input values for power plant module.
Cost_yr      1989
Pltinputlabe
Gross Capacity, MW [      600]
Steam HR, Btu/kWh [     8000]
Capacity Factor, % [      75]
Excess Air Boiler, % [     20]
Inlet Air Leakage, % [     19]
Furnace Temp, R [    2960]
Economizer Temp, R [    1160]
Air Heater Temp, R [     300]
Pulverizer Elect, % [     0.6]
Steam Pump Elect, % [     0.65]
Fan Elect, % [      1.5]
Cooling Elect, % [     1.8]
Misc. Elect, % [     1.3]
Misc Loss Boiler, % [     0.5]
Plant Indir Charge % [     50]
Labor Rate, $/man-hr [    21.8]
Land Cost, $/acre [    7189]
```

Example>mid plantchar *Partial list of performance parameters for power plant module.*

Plant_param	Value
Plant_index	
Boiler Efficiency, % [	88.3]
Cycle HR, Btu/kWh [	9060]
Fuel Usage, tons/hr [	266.6]
Coal Credit, ton/hr [	0]
Energy Credit, MW [	0]
Adj. Gross Cap, MW [	564.9]
Net Capacity, MW [	533.1]
Net HR, Btu/kWh [	10.2K]
Annual Oper Hours [	6575]
Net Elec Gen, MWh/yr [	3.5M]

The summary of plant economic characteristics shown below represents results of an analysis in which many other model parameters not shown here also have been set. In particular, this analysis reports all costs in constant dollars from which effects of inflation are removed. (Had we used leveled costs in nominal dollars many of the numbers would be much higher.)

Example>mid planteconchar *Partial list of economic parameters for base power plant module, excluding pollution control systems*

Cost_yr	Value
Bpecon_param	1989
Bp_econ_ind	
Coal Credit, M\$/yr [	0]
Energy Cr, O&M M\$/yr [	0]
Total Base Plant O&M [	79.44]
Base Plt Dir Cost M\$ [	388.3]
Energy Cr Dir Cst M\$ [	0]
Ttl Base Plt Dir Cst [	388.3]
Total Capital Cst M\$ [	582.5]
Normal Capital, \$/kW [	1093]
RevReq w/oUtil M\$/yr [	139.7]
Electric Cost, \$/MWh [	37.6]
Utility Income M\$/yr [	-7.87]
Total Rev Req, M\$/yr [	131.8]
Total Rev Req, \$/MWh [	37.6]

Final_cost is the main table used for deterministic analyses. Each row represents a different process. The columns represent the the total capital cost (TCC) in M\$, the non-utility O&M cost (O&M) in M\$/yr, the utility cost (UTIL) in M\$/yr, the total revenue requirement (TRR) in \$M/yr, the total revenue requirement (ELECT) in \$/MWh, and finally the the total capital cost (TCC) in \$/net kW for all the processes.

Example>tabwidth:7 *Sets the character width of columns for tables, so that Final_cost fits on the screen neatly.*

Example>mid final_cost *This is the main summary table for the economics.*

Cost_yr =	1989	TCC,	O&M,	UTIL,	TRR,	TRR,	TCC
Costs							
Process							
Units [	\$M,	\$M/yr,	\$M/yr,	\$M/yr,	\$/MWh,	\$/kW]	
Base Pl [	582.5,	79.44,	-7.87,	131.8,	37.6,	1093]	
SCR [	39.46,	7.966,	1.234,	13.28,	3.789,	74.03]	
ESP [	25.13,	0.083,	0.353,	3.034,	0.866,	47.15]	
Wet FGD [	112.7,	14.51,	6.285,	32.45,	9.258,	211.4]	
Solid W [	32.2,	3.34,	0,	6.669,	1.903,	60.41]	
Total [	792.,	105.3,	0,	187.2,	53.42,	1486]	

2.5 Typical Demos Commands

The examples so far showed some of the basic input and output operations of IECM. To fully utilize the model, however, user experience with Demos commands is required. (In future model developments we hope to implement a menu-based system that will minimize the need for Demos expertise; but for now it's necessary.) Some of the most common Demos commands needed to get and display results are described and demonstrated below. They are described in more detail in the Demos Manual [1], along with other useful commands the user should become familiar with.

Bye	Used to exit Demos. This command will be demonstrated at the end of the tutorial.
Clock	Gives current time, date, and memory usage. The maximum heapsize depends on the available memory, and can be set with the system variable heapsize. On workstations with a 16 Megabyte swap space, the maximum that the heapsize should be set to is 12 Megabytes. As the memory usage approaches this limit, Demos will print out warnings. In the Macintosh version, the heapsize variable is set at startup by Demos and it cannot be changed while running Demos. Also, on the Macintosh version the amount of memory available is also displayed.
Help	For on-line Help about Demos.
Mid	Display the deterministic value of a variable regardless of the verbosity setting. This is useful for displaying tables, and has already been demonstrated.
Photo	Records everything displayed on the screen (except graphs) to a file. See the next section on graphing for creating files of graphs. Endphoto stops a photo session, while pausephoto and resumephoto are used to pause and continue photo sessions.
Quit	Used to exit Demos. Only available on the Macintosh version.
Save	Saves the local variables in the current model into a file.
Show	Display all the attributes of a variable.
Update	Allows variable definitions to be overwritten when files are being read. This mode is turned off with the command Endupdate. The commands, Update and Endupdate, are used at the beginning and end of all source code files. It lets the new models redefine variables that have already been loaded by previous files. If Update was not used, an error would occur for every variable in the file, stating the variables are already defined. Demos would then ask for a new name for each variable.
Verbosity	Used to control the amount of feedback that Demos prints to the screen. The most common verbosity settings are 1, 2, and 3. When verbosity equals 1 it, prints various messages about the status of variables after files are read and when variables are being evaluated. This can be very useful for tracking down errors in files, but it is tedious for large models like the IECM. Another feature of verbosity equal to 1 is that the numeric value of variables is always displayed. When verbosity equals 2 (the default), Demos prompts for undeclared variables when variables are created in Demos, and array dependencies are displayed instead of numeric values. Setting verbosity to

3, cause the combined effects of 1 and 2. This was set to 2 in the "loading" files to prevent Demos from printing the status of every variable in the IECM to the screen. In general, it is recommended that verbosity be set to 2.

Why

Display the variable definition, its inputs, and the values of the inputs. If verbosity equals 1, the numeric values of the definitions will be displayed. If verbosity equals 2, the array dependencies are displayed.

Example>clock
 Mon May 11 15:49:36 *The following appears on the Unix version. Heapused should be less than maxheap.*
 cpu totalcpu walltime totalwalltime heapused maxheap
 32.8660 32.8660 5:49:36 5:49:36 1.6000M 6.7000M

The following appears on the Macintosh version. Free Memory is the amount of memory available and should be greater than 20K.
 Mon, May 11, 1989 3:49 PM
 Heap Used Max Heap Free Mem
 1.073M 6.5000M 5.573M

Example>help

HELP:

The help text is organized hierarchically with Topics, subtopics, etc, corresponding to the Chapters, Sections and Subsections in the Users' Manual. After asking for Help on a Topic, it will print a list of its subtopics. From these you can choose one, none, all or any unlisted topic. Words that are capitalized (e.g. Kind, Topic, Session) in this text are Topics (or subtopics) with Help available for them.

Help prints this text on how to get help.
 Help ? prints a list of the main Help Topics.
 Help t prints help text for the Topic t or lists its subtopics.
 Help Topics Describes the main Help Topics.
 Help Intro prints some introductory information for new users.
 List k lists all the existing Objects of the given Kind k.
 MoreHelp continues printing Help text where it last left off.

Example>help ?

CONTENTS:

Help Topics and some subtopics:

Introduction	Basics	Prompts	Objects	Names
Numbers	Sessions	Special_Chars	Commands	Save
Bye	Variables	Dependencies	Attributes	Files
Photo	Functions	Distributions	Plotting	CDF
Framing	Arrays	Indexes	Sensitivity	Versions
System_Variabls	Syntax	Statements	Expressions	Arithmetic

There is Help on (almost) every system-defined Object, Command, Kind, Function, Attribute, SystemVar.

Example>photo

The files example.pho will have a record of this tutorial from this point on, except for any graphs.

Please enter name [example.pho]:

Return key accepts default file name.

Example>save
Please enter name [example.mod]:

*Each % mark represents ten lines of code.
Return key accepts default value.*

saving ...

Opening example.mod

Since only one variable, Model Wetscresp, is currently in Model Example, Demos only writes a short file stating that it contains the model Wetscresp. Figure 2-7 shows the contents of the file Example.mod.

Closing example.mod
done

Example>show coalinputs

Results of using the Show command on a model.

Model Coalinputs

Description: contains all the variables needed to be defined when the ~
coal is changed

Contains: [Decision Coals, Variable Coalchar_tab, Variable Raw_coal_cst,
Variable Ctrans, Variable Clean_cost, Variable Minerpbcost]

Author: js04

Needs: [Model Indices]

Example>show emisconsso2

Results of using the Show command on a variable.

Variable Emisconsso2

Units: lb/MBtu

Description: sulfur dioxide emission constraint, ~

describes the Rev. NSPS (1978) emission constraints

Definition: if potemso2 < 2 then 0.3 * potemso2 ~

else if potemso2 >= 2 AND potemso2 < 6 then 0.6 ~

else if potemso2 >= 6 AND potemso2 < 12 then 0.1 * potemso2 ~

else 1.2

Check: 0 <= emisconsso2 and emisconsso2 <= 15

*Now let's demonstrate the effect of verbosity. We'll use
the Why command to see how a variable depends upon
other variables.*

Example>verbosity

Verbosity :

= 2

Example>why cd

Cd : cost of as-fired coal (\$/wet ton) = Table(Cost_yr)

Description: cost of delivered coal on a wet ton basis. The term ~
multiplying raw_coal_cst is used to keep the cost per dry ton of coal ~
constant, since coal cleaning is likely to change the moisture content

Cd = Raw_coal_cst*(1-h2ofrac) / ~
(1-coalchar_table[coalindex=1,fuelchar_index = 'Moisture']) + CTrans + ~
Clean_cost + MinerPBcost

Coalindex :

= [1]

Fuelchar_ind:

= [HHV,

Carbon, Hydrogen, Oxygen, Sulfur, Nitrogen,

Ash, Moisture]

Coalchar_tab: Input coal Characteristics

=

Table(Coalindex, Fuelchar_ind)

H2ofrac : Moisture Content

(1b h2o/1b) = 0.123

Raw_coal_cst: raw coal cost

(\$/wet ton) = Table(Cost_yr)

Ctrans : Transportation Cost

(\$/wet ton) = Table(Cost_yr)

Clean_cost : Coal Cleaning Cost

(\$/wet ton) = 0

Minerpbcost : Miner Benefit Cost

(\$/wet ton) = 2

Example>verb:1

*With verbosity equal to 1 we get the results rather than the
general relationships.*

Figure 2-7: Contents of the File, Example.mod

{ From Demos User js04 , Demos Project EXAMPLE at Fri Apr 14 15:19:43}

{ System Variables with non-default values:}

Model Example

Description: Example on how to use the IECM

Contains: Model Wetscresp

Author: John Salmento

Date: Fri Apr 14 11:02:28

Savedate: Fri Apr 14 15:19:38

Close Example

Example>why cd

Cd : cost of as-fired coal (\$/wet ton) =
 Cost_yr 1989
 [35.64]

Description: cost of delivered coal on a wet ton basis. The term ~ multiplying raw_coal_cst is used to keep the cost per dry ton of coal ~ constant, since coal cleaning is likely to change the moisture content

$$Cd = \text{Raw_coal_cst} * (1 - h2ofrac) / \sim (1 - \text{coalchar_table}[\text{coalindex}=1, \text{fuelchar_index} = \text{'Moisture'}]) + \text{CTrans} + \sim \text{Clean_cost} + \text{MinerPBcost}$$

Coalindex : = [1]
 Fuelchar_ind: = [HHV,
 Carbon, Hydrogen, Oxygen, Sulfur, Nitrogen, Ash, Moisture]
 Coalchar_tab: Input coal Characteristics =
 Table(Coalindex, Fuelchar_ind)
 H2ofrac : Moisture Content (lb h2o/lb) = 0.123
 Raw_coal_cst: raw coal cost (\$/wet ton) =
 Cost_yr 1989
 [27.43]

Ctrans : Transportation Cost (\$/wet ton) =
 Cost_yr 1989
 [6.201]

Clean_cost : Coal Cleaning Cost (\$/wet ton) = 0
 Minerpbcost : Miner Benefit Cost (\$/wet ton) = 2

Example>verb:2

Verbosity is OK

2.6 Graphing and Probabilistic Analysis

Graphs and probabilistic analysis are perhaps the most important features of Demos. Using the Andrew windowing environment or the Macintosh allows high resolution graphs to be displayed on the screen easily. The probabilistic features of Demos and the IECM are discussed more in Chapter 3. There are six basic command to understand, which are described below.

Graph	Plots the deterministic values of one variable against another.
Ctl_global	This is an IECM decision variable which turns on or off the uncertainty functions in the IECM. It's default value is 'all off', which only allows the median value of all variables to be evaluated. Setting it to 'all on' turns on all the uncertainties. This variable is describe in more detail in Section 3.3.8.
Samplesize	Number of points used to estimate the pdf and cdf's. Larger samplesizes produces more accurate results, but requires longer computation time and uses more memory. For development work, values around 50 are typically used.
Pdf	Plots the probability distribution function of a variable. Also affected by the sample size.
Cdf	Plots the cumulative function of a variable. The accuracy of this plot is directly related to the variable, Samplesize.
Cdfnum	This is a function that is defined in the IECM. It prints a table which produces the numeric values of a cdf at 0%, 5%, 10%, 20%, 30%, 40%,

50%, 60%, 70%, 80%, 90%, 95%, and 100%. This function produces a table for those variables which are depended upon vectors.

Cdfperf

This does the same thing as Cdfnum; however, it produces a table for those variables which do not depend upon vectors.

The graph, cdf, and pdf commands produce their graphs in a separate window on the right side of the screen if the Andrew window manager is used. Figures 2-8, 2-10, and 2-12 show the picture of the screen after each command shown in the example below. The Andrew window manager automatically sizes the windows to take up any available space. The default number of graphs that are displayed on the right side is two. Demos overwrites the oldest graph if more that two graphs are requested. *In the Unix version, do not choose Zap as a menu selection in a graph window. This will cause the Demos program to crash.*

On the Macintosh version, a separate window pops up, which overlaps the current window. These windows can be moved to any place on the screen, or can be resized to any rectangular shape. Figures 2-9, 2-11, and 2-13 show the Macintosh screen after each of the commands below. After each window pops up, move it to a different part of the screen, so that they don't overlap. To close a graph window, click the little box in the upper left hand corner of the window. To get back to the Demos window, click on the Demos window and it will come to the foreground.

Let's do some simple parametric analysis to determine the affects of the plant size on pollution control costs. We've just changed the plant size from a scalar to a scalar.

Example>mwg:[250 300 350 400 450 500 550 600]

The variable, Elecpc shows the total revenue requirement in mills/kWh for all the pollution control equipment excluding the effects of coal cleaning if it is used. The text strings enclosed single quotes on the next line produce labels for the Y and X axis, respectively. More details about graphs can be found in reference [1].

Example>graph elecpc mwg 'Pollution Control Cost (Constant 1989\$) '~
Continue:'Gross Electrical Output (MW)'

Each graph produces a separate window. For the following three graphs, the full page screen pictures are shown on the following pages.

Pollution Control Cost (Constant 1989\$)

Example>mwg:600

Example>ctl_global:'all on'

Turns on the probability functions. Selected parameter uncertainties have been assigned as part of the data inputs for this example. Details are discussed later in this report.

Example>samplesize

Sample size :

= 20

Example>samplesize:50

This changes the sample size to 50.

Example>pdf elecpc

Probability density

Figure 2-8: Picture of Screen after Graph Elecpc Mwg Command

Figure 2-9: Picture of Screen after Graph Elecpc Mwg Command on the Macintosh

Figure 2-10: Picture of Screen after Pdf Elecpc Command

Figure 2-11: Picture of Screen after Pdf Elecpc Command on the Macintosh

Example>cdf elepc

Cumulative probability

Example>cdfnum(elepc)

Cost_yr
Cipct

1989

Since this is a probabilistic analysis, the exact values may not appear when you run the this tutorial.

0%	[	12.74]
5%	[	13.15]
10%	[	13.82]
20%	[	14.35]
30%	[	14.9]
40%	[	15.17]
50%	[	16.04]
60%	[	16.45]
70%	[	17.03]
80%	[	17.45]
90%	[	17.88]
95%	[	18.4]
100%	[	19.1]

Example>cdfperf(wet_cap)

Cdf_param
Cipct

Value

Cdfperf or cdfnum do not have to follow a cdf command.

0%	[	36.09]
5%	[	37.04]
10%	[	37.36]
20%	[	37.77]
30%	[	38.06]
40%	[	38.3]
50%	[	38.54]
60%	[	38.79]
70%	[	39.04]
80%	[	39.31]
90%	[	39.75]
95%	[	40.07]
100%	[	40.96]

Example>ctl_global:'all off'

This turns off all the probabilistic functions.

Figure 2-12: Picture of Screen after Cdf Elecpc Command

Figure 2-13: Picture of Screen after Cdf Elecpc Command on the Macintosh Version.

2.7 Changing Coals and Coal Cleaning

To change coals, six variables have to be redefined: Raw_coal_cst, Ctrans, Coalchar_table, Cc_table, Coals, and Coalindex. These are described below.

Raw_coal_cst	Cost of raw coal in current dollars per wet ton. The actual numeric value can be in any year's dollars as long as the definition also multiplies it by cost_index and divides it by the appropriate index for that year.
Ctrans	Transportation cost in current dollars per wet ton. It is also scaled by the cost index relative to the year initially specified.
CoalChar_table	The coal characteristics table. It requires at least one row of values. The default configuration is to have the first row represent a raw coal, and the other rows represent various levels of coal cleaning. (The coal cleaning module is described in more detail in Chapter 16.) Each column represents different coal characteristic: higher heating value (Btu/lb), carbon fraction, hydrogen fraction, oxygen fraction, sulfur fraction, nitrogen fraction, ash fraction, and moisture fraction.
Cc_table	The coal cleaning plant performance table. It contains information about the coal cleaning level, mass yield of the plant excluding the thermal driers, thermal drier yield, the fraction of cleaned coal dried, the amount of moisture evaporated, the moisture content of the refuse coal, the fraction of coal that was cleaned with froth flotation, and the retention time in the flotation cells. It has to have as many rows as CoalChar_table. The first row is for the raw coal, so all the values are zero, except for the mass yield of the plant and thermal drier.
Coals	This variable chooses the coal characteristics from the table Coalchar_table for the current run. The values of the elements can not exceed the total number of rows in coalindex. Usually this value is set to 1, (ie., the raw coal).
Coalindex	The index used in construction of the table Coalchar_table. It has to be a vector from [1] to the number of rows in Coalchar_table. For example, if Coalchar_table has five rows, then Coalindex would be defined as [1, 2, 3, 4, 5].

There are several ways of changing these characteristics (see the the Demos Manual [1]). The easiest way is to create a file with all the changes in it and load it with the read command. The file "epri.coal" in the coalfiles directory can be used as an example. This file can be loaded at any time; the new output results can be displayed by just retyping variable names.

Example>pausephoto

If a file is read into Demos while photo is on the contents of the file appears in the photo file, so it is better to pause photo.

Example>read coalfiles/epri.coal

##

On the Macintosh use either the Read Model command in the file menu or type read files:iec:coalfiles:epri.coal.

The definitions of all Variables and Functions are OK

Example>resumephoto

Example>coalindex

Coalindex :

= [1]

Example>show coalchar_table

Variable Coalchar_tab

Title: Input Coal Characteristics

Description: Illinois No 6. from Epri "Technical Assessment Guide"

Definition: Table(Coalindex,Fuelchar_index) (

10100. .575 .037 .058 .04 .009 .16 .12

)

Example>mid raw_coal_cst

Cost_yr 1989
[24.88]

Example>mid ctrans

Cost_yr 1989
[6.201]

Example>coals:2

An example of the error when Coals is set incorrectly.

Example>hhv

Hhv is the variable which extracts the higher heating value for the coal chosen by Coals out of Coalchar_tab.

Hhv : Higher Heating Value (btu/lb)

? Warning: *This first warning is telling you that coals is incorrectly defined. Type a return or yes for the question below.*

Decision Coals fails to satisfy Check: ((1<=Coals) AND (Coals<=Max(Coalindex)))

Would you like to stop to try and fix the problem? [Yes]:

Decision Coals

Title: Coals chosen

Description: Coal being used by model

Definition: 2

Check: 1 <= coals and coals <= max(coalindex)

This error occurs since Demos is still trying to evaluate Hhv. |

Argument number 1 of SysFunction Subscript must be a number or an array.

Error during evaluation of Hhv

Definition of Hhv is: if CCconfig = 'conventional' OR CCconfig = 'none' ~
then coalchar[Fuelchar_index = 'HHV'] ~
else if CCconfig = 'hgms' then Hhv_hgms ~
else if CCconfig = 'coal pyrite' then HHVclean ~
else -1

Since this problem is caused by coals being incorrectly defined, type no to all the questions that ask if you want to redefine other variables. This error can be fixed by redefining Coals to 1. On the Macintosh version the questions that Demos asks below appear in little windows. Just click on the no button.

Do you want to (re)define it? [Yes]: no

? Warning:

Attribute unusable

Example>coals:1

An important point to remember is that the photo session will turn off after this type of error, so it is important to check it. The following warning indicates that photo has stopped. Since we are about to read in another file, we are not going to restart photo.

Example>pausephoto

? Warning: Photo not on.

Changing coals or coal cleaning module characteristics is similar as far as Demos is concerned. Both steps require defining the same six variables listed above; however, coal cleaning also requires working with the Fortran coal cleaning model and the Demos coal cleaning module. Because of some of the limitations of Demos, only the economic part of the coal cleaning model is written in Demos. The performance part of the coal cleaning model is in Fortran and is described in more detail in Chapter 16. The economic module is in the file, "srcpetc/coalclean.petc." It can be loaded at any time, as any other module. It has its own input parameters and summary tables; however, all the input parameters are entered from a data file created by the Fortran model.

Running the coal cleaning module is thus a three step operation. First, the Fortran model is run, which creates a data file with all the input parameters for the economic module in Demos. Then the file created by the Fortran model is loaded into Demos. Finally, the coal cleaning economy module is loaded. Once the module is loaded various levels of coal cleaning are selected by setting the vector Coals to a new value. All the effects of cleaning the coal are correctly passed on to the rest of the IECM.

Washed coal data files from the Fortran module identical to other coal characteristic files except that they contain another input table, Cc_table. Cc_table contains the coal cleaning plant's performance information, which is needed by the Demos coal cleaning module to calculate washed coal prices. There is one row in Cc_table for every row in coalchar_table. Each row represents a different washed coal. For the example, it is assumed that the Fortran model was run to create the file "iec.coal". The contents of the file are shown in Figure 2-14. The text in italics on the right side of the page explains each variable or command. A more detailed explanation of the variables and their definitions is described in Chapter 16. The next steps are as follows:

```
Example>read coalfiles/iec.coal  On the Macintosh use either the Read Model
**                               command in the file menu or type read files:iec:coalfiles:iec.coal.
The definitions of all Variables and Functions are OK

                               On the Macintosh use either the Read Model command in
                               the file menu or type read files:iec:srcpetc:coalclean.coal.
Example>read srcpetc/coalclean.petc  Loads the coal cleaning module.
*****
The definitions of all Variables and Functions are OK

Example>photo  Let's restart photo. Type no to the first question and yes to the
                second question.
example.pho: File already exists. Do you wish to replace it?
[Type Y or N]: no
File already exists. Do you wish to append to it? [Type Y or N]: yes

Example>coals:2  This picks the 2nd row of coal characteristics which corresponds to
                 a 10% sulfur reduction by washing.

Example>tabwidth:20
```


Figure 2-14: Contents of iec.coal File Created by the Fortran Coal Cleaning Program.

```

Update                                     This is to allow the variables defined in this file to overwrite
                                           the same variables currently loaded into Demos.
Model Ieccoal                             This defines the model name.

                                           The line below is the cost of raw coal adjusted by the cost_index
Raw_coal_cst: 26.1000 * Cost_Index / 342.5

Coals: 1                                  This redefines coals to 1, to prevent potential errors from
                                           occurring, since coals could have equaled a value greater than 13 before
                                           this file was read.

                                           This indicates that there are 12 different washed coals to consider,
                                           plus the unwashed coal.
CoalIndex: Sequence(1, 13)

Coalchar_table: Table(Coalindex, Fuelchar_index) (
10194. .5695 .0370 .0719 .0436 .0112 .1438 .123
10354. .5784 .0383 .0766 .0398 .0114 .1241 .1314
10130. .5659 .0375 .0750 .0390 .0112 .1246 .1469
10293. .5750 .0388 .0798 .0352 .0113 .1045 .1553
10565. .5902 .0406 .0857 .0316 .0116 .0839 .1564
10385. .5802 .0384 .0768 .0400 .0114 .1281 .1250
10539. .5888 .0397 .0817 .0361 .0116 .1081 .1340
10838. .6055 .0416 .0879 .0324 .0119 .0868 .1338
11181. .6247 .0438 .0947 .0287 .0123 .0620 .1339
10352. .5783 .0383 .0766 .0398 .0114 .1276 .1280
10614. .5929 .0400 .0823 .0363 .0117 .1088 .1280
10914. .6097 .0419 .0885 .0327 .0120 .0871 .1280
11267. .6294 .0441 .0954 .0289 .0124 .0618 .1280
)

CC_table: Table(Coalindex, CC_Index) (
0 1.0000 1.0000 0.0000 0.0000 0.0000 0.0000 0.000
2 0.9634 1.0000 0.0000 0.0000 0.1380 0.0000 0.000
3 0.9699 1.0000 0.0000 0.0000 0.1880 0.0000 0.000
3 0.9320 1.0000 0.0000 0.0000 0.1694 0.0000 0.000
3 0.8751 1.0000 0.0000 0.0000 0.1558 0.0000 0.000
4 0.9703 0.9822 0.4124 0.1007 0.2031 0.0000 0.000
4 0.9338 0.9826 0.4093 0.1014 0.1816 0.0000 0.000
4 0.8856 0.9826 0.4316 0.1014 0.1638 0.0000 0.000
4 0.6943 0.9828 0.4126 0.1045 0.1654 0.0000 0.000
5 0.9722 0.9931 1.0000 0.0379 0.2008 0.0608 2.000
5 0.9357 0.9912 1.0000 0.0495 0.1805 0.0608 2.000
5 0.8882 0.9912 1.0000 0.0512 0.1645 0.0608 1.500
5 0.6996 0.9917 1.0000 0.0500 0.1669 0.0608 1.071
)

Variable MIBC                             MIBC usage for froth flotation circuit.
Units: lb/ton
Definition: 0.15

Variable Oil2                             #2 oil usage for froth flotation circuit.
Units: lb/ton
Definition: 0.15
Close ieccoal
endupdate

```

These are the properties of the raw coal and 12 washed levels.

The 12 washed coals correspond to several sulfur reduction levels for each of four prep plant configurations.

Example>mid fuelsummary

Fuelsummary is a command which creates a table of the fuel characteristics being used. For example, since Coals equals 2 the second row from Coalchar table is chosen. The difference between Coalchar table and Fuelsummary is that Coalchar table contains the information for all the washed coals, while Fuelsummary contains only the information chosen with Coals.

Coalparameter Fuelchar_ind	Value
HHV	[10.4K]
Carbon	[57.84]
Hydrogen	[3.83]
Oxygen	[7.66]
Sulfur	[3.98]
Nitrogen	[1.14]
Ash	[12.41]
Moisture	[13.14]

Example>mid ccsummary

Ccsummary is similar to fuelsummary, except that it contains the performance information of the coal cleaning plant for the coal chosen. The difference between Cc table and Ccsummary is that Cc table contains the information for all the washed coals, while Ccsummary contains only the information chosen by Coals.

Ccparameter Cc_index	Value
Cleaning Plant Level	[2]
Washer Mass Yield	[0.963]
Thermal Drier Yield	[1]
Coal Frac. to Drier	[0]
Moisture Evaporated	[0]
Moisture in Refuse	[0.138]
Coal Frac. to Flotat	[0]
Time in Flot. Cells	[0]

Example>tabwidth:6

Now let's determine which coal cleaning levels produce the lowest pollution control cost for a given % sulfur reduction. First, select all the coals from Coalchar table.

Example>coalindex

The variable Coals can have more than one value. Coalindex shows all the valid values for the Coals.

Coalindex :	1,	2,	3,	4,	5,	6,	
[	7,	8,	9,	10,	11,	12,	13]

Example>coals:sequence(1,13) *This chooses every washed coal for analysis, and changes Coals from a scalar to a vector.*

Example>round(sulfur_red)

This variable shows the sulfur reduction on an energy basis achieved by the coal cleaning plant for the washed coals. The function round is used to make the output neater. There are duplicate levels of sulfur reductions because different coal cleaning plant configurations were used to wash the coal. The data are not formatted as in the tables above, though this could easily be done with other Demos commands.

Coals	1,	2,	3,	4,	5,	6,	7,	8,	9,	
10,	11,	12,	13							
	[	0,	10,	10,	20,	30,	10,	20,	30,	40,
10,	20,	30,	40]							

Example>mid clean_cost
Cost_yr
Coals

*This shows the coal cleaning cost in dollars per wet ton for each
1989 coal selected.*

1	[	0]
2	[	3.051]
3	[	3.399]
4	[	4.709]
5	[	6.902]
6	[	4.847]
7	[	6.109]
8	[	8.011]
9	[	17.88]
10	[	4.687]
11	[	6.128]
12	[	7.973]
13	[	17.6]

Example>mid elecpc

Now let's see which value of coal produces the lowest pollution control cost. Elecpc is similar to Elecpc, except that it includes the effects of coal cleaning. The only requirement is that the variable Coals be a vector which includes 1. As can be seen, the raw coal has the lowest Cost. Results for this case show that pollution control cost savings at the power plant are not sufficient to offset the higher cost of cleaned coal, so that the unwashed coal is most economical for this situation.

Wetscresp>mid elecpc
Cost_yr
Coals

1	[	15.82]
2	[	16.49]
3	[	16.96]
4	[	16.74]
5	[	16.81]
6	[	17.4]
7	[	17.17]
8	[	17.02]
9	[	20.38]
10	[	17.35]
11	[	17.09]
12	[	16.92]
13	[	20.11]

2.8 Redefining Variables

Redefining variables is fairly easy to do; it is similar to changing the value of input parameters. These changes do not get saved to a file until the model is saved. Since Demos is non procedural, the new definitions' values will be propagated through the IECM correctly. The example below shows how to redefine variables using two different methods. The first method uses a command called "Whatif," which does not change the definition of a variable. The second method shows how to redefine a variable. It also shows some common problems with summary tables when variables are redefined. Remember that none of these methods will change the source files until the model is saved.

Example>coals:2

Example>show xwet

Variable Xwet

Title: Wet FGD Energy Penalty

Units: fraction

Description: Energy penalty for steam and electricity usage

Definition: $(fgd_reheat * 1K/hrsteam + fgdm * uffgdm) / Mwg$
uffgdm is an uncertainty variable. It is described in section 3.3.8.

Example>xwet *Displays the value of Xwet*

Xwet : Wet FGD Energy Penalty (fraction) = 0.041

Example>verb:1

Example>whatif xwet:0.02 *This only changes Xwet temporary (to see the effect of reducing the energy use).*

Examine the effect on which Variable? [Xwet]: final_cost

Final_cost : =

Cost_yr	=	1989					
Costs		TCC,	O&M,	UTIL,	TRR,	TRR,	TCC
Process							
Units [	\$M,	\$M/yr,	\$M/yr,	\$M/yr,	\$/MWh,	\$/kW]	
Base Plant [	582.5,	83.38,	-7.88,	135.7,	38.67,	1091]	
SCR [	39.53,	7.968,	1.261,	13.32,	3.794,	74.04]	
ESP [	24.52,	0.08,	0.353,	2.968,	0.846,	45.93]	
Wet FGD [	111.9,	13.8,	6.265,	31.63,	9.01,	209.5]	
Solid Wast [	29.67,	3.182,	0,	6.249,	1.78,	55.57]	
Total [	788.1,	108.4,	8e-16,	189.9,	54.1,	1476]	

Xwet is OK

would be changed to

Cost_ = 1989

Costs		TCC,	O&M,	UTIL,	TRR,	TRR,	TCC
Process							
Units [	\$M,	\$M/yr,	\$M/yr,	\$M/yr,	\$/MWh,	\$/kW]	
Base [	582.5,	83.38,	-4.66,	138.9,	38.67,	1066]	
SCR [	39.53,	7.968,	1.261,	13.32,	3.706,	72.33]	
ESP [	24.52,	0.08,	0.353,	2.968,	0.826,	44.87]	
Wet F [	111.9,	13.8,	3.051,	28.41,	7.907,	204.7]	
Solid [	29.67,	3.182,	0,	6.249,	1.739,	54.28]	
Total [	788.1,	108.4,	0,	189.9,	52.84,	1442]	

Xwet is OK

Example>tabwidth:20

Example>xwet:[0.03 0.07]

*Now let's change Xwet to a vector of constants
This is one way of performing traditional sensitivity analysis.*

Example>mid trrwet

*Shows the total revenue requirement of the scrubber as
a function of the energy penalty.*

Cost_yr	1989	
Xwet		
0.03 [		29.94]
0.07 [		36.04]

It is OK

Weteconchar is a output table of economic characteristics for the wet FGD module. It depends upon the vectors wetecon_param and wet_econ_ind. Since we created a new vector, Xwet, Demos will print out a table for each value of Xwet.

Example>mid weteconchar

```
Xwet = 0.03
Cost_yr = 1989
Wetecon_para Value
Wet_econ_ind
Material Handling M$ [ 9.56]
Train Cost, M$ [ 51.29]
Direct Capital, M$ [ 60.85]
Engineering Cost, M$ [ 4.26]
Architect Cost, M$ [ 1.217]
Construction Cost M$ [ 9.74]
Contractors Fees, M$ [ 3.043]
Contingencies, M$ [ 7.911]
Total Plant Cost, M$ [ 87.02]
Interest, M$ (curr) [ 8.228]
Total Plant Inv, M$ [ 95.24]
Startup Cost, M$ [ 13.39]
Royalties, M$ [ 0]
Working Capital, M$ [ 3.225]
Total Capital, M$ [ 111.9]
Indirect Cost Factor [ 0.838]
Total Capital, $/kW [ 206.9]
Reagent Cost, M$/yr [ 3.55]
Fixed Operate, M$/yr [ 10.25]
Total O&M Cost M$/yr [ 13.8]
Revenue wo Uti M$/yr [ 25.36]
Utilities, M$/yr [ 4.576]
Total Revenue, M$/yr [ 29.94]
Total Rev, mills/kWh [ 8.424]
```

```
Xwet = 0.07
Cost_yr = 1989
Wetecon_para Value
Wet_econ_ind
Material Handling M$ [ 9.56]
Train Cost, M$ [ 51.29]
Direct Capital, M$ [ 60.85]
Engineering Cost, M$ [ 4.26]
Architect Cost, M$ [ 1.217]
Construction Cost M$ [ 9.74]
Contractors Fees, M$ [ 3.043]
Contingencies, M$ [ 7.911]
Total Plant Cost, M$ [ 87.02]
Interest, M$ (curr) [ 8.228]
Total Plant Inv, M$ [ 95.24]
Startup Cost, M$ [ 13.39]
Royalties, M$ [ 0]
Working Capital, M$ [ 3.225]
Total Capital, M$ [ 111.9]
Indirect Cost Factor [ 0.838]
Total Capital, $/kW [ 216.5]
Reagent Cost, M$/yr [ 3.55]
Fixed Operate, M$/yr [ 10.25]
Total O&M Cost M$/yr [ 13.8]
Revenue wo Uti M$/yr [ 25.36]
Utilities, M$/yr [ 10.68]
Total Revenue, M$/yr [ 36.04]
Total Rev, mills/kWh [ 10.61]
```

It is OK

It is possible to get Demos to print out the values of Xwet beside each other. We'll use the Reform command to rearrange the order the Demos prints out the arrays. First, let's find out what arrays weteconchar depends upon by set Verbosity to 2 and typing its name.

Example>verb:2

Example>weteconchar

Weteconchar : (M \$) =
Table(Xwet, Cost_yr, Wet_econ_ind, Wetecon_para)

The Reform command needs the names of the variables after the word, table, in the line above. The Reform command is illustrated below, notice that the variables enclosed in parenthesis are rearranged. The Reform command tells Demos the order in which to print out the arrays of tables. The last item in the list is corresponds to the column labels, while the second to the last column corresponds to the row labels.

Example>reform(weteconchar,[cost_yr,wetecon_param,wet_econ_ind,xwet])

Cost_yr	1989	
Wetecon_para		Value
Xwet	0.03,	0.07
Wet_econ_ind		
Material Handling M\$ [	9.56,	9.56]
Train Cost, M\$ [	51.29,	51.29]
Direct Capital, M\$ [	60.85,	60.85]
Engineering Cost, M\$ [	4.26,	4.26]
Architect Cost, M\$ [	1.217,	1.217]
Construction Cost M\$ [	9.74,	9.74]
Contractors Fees, M\$ [	3.043,	3.043]
Contingencies, M\$ [	7.911,	7.911]
Total Plant Cost, M\$ [	87.02,	87.02]
Interest, M\$ (curr) [	8.228,	8.228]
Total Plant Inv, M\$ [	95.24,	95.24]
Startup Cost, M\$ [	13.39,	13.39]
Royalties, M\$ [	0,	0]
Working Capital, M\$ [	3.225,	3.225]
Total Capital, M\$ [	111.9,	111.9]
Indirect Cost Factor [	0.838,	0.838]
Total Capital, \$/kW [	206.9,	216.5]
Reagent Cost, M\$/yr [	3.55,	3.55]
Fixed Operate, M\$/yr [	10.25,	10.25]
Total O&M Cost M\$/yr [	13.8,	13.8]
Revenue wo Uti M\$/yr [	25.36,	25.36]
Utilities, M\$/yr [	4.576,	10.68]
Total Revenue, M\$/yr [	29.94,	36.04]
Total Rev, mills/kWh [	8.424,	10.61]

Now let's change Xwet back to the original definition.

Example>xwet:(fgd_reheat*1K/hrsteam + fgdm * uffgdm)/Mwg

Example>endphoto *Stopping photo now.*

2.9 Exiting and Saving the Model

This section shows how to save and exit Demos. The commands are save and bye. Save save the whole model into a file. Bye, first ask if you want to save this model, and then exits. These are shown below.

Example>bye

Incorporate the changes that you have made
in this session into the current Model? [yes]:no

Would you like to save the changes that you have made
in a version file ? [Yes]: no

Model Example not saved.

Goodbye js04

3. Initialization and IECM Structure

This chapter describes the structure, hierarchy and initialization files for the IECM. There are two initialization files "initiecm.mod" and "initiecm.petc." "Initiecm.mod" initializes the hierarchy and several performance variables which are needed for the model to run properly. "Initiecm.petc" initializes several economic variables and functions. Many times the term "module" is used with the name of a particular device, such as "wetfgd module." This term refers to all the variables, models, and files which describe the performance and economics of that device. So "wetfgd module" refers to both "wetfgd.mod" and "wetfgd.petc."

The IECM is written in the Demos language, which is described in the "Demos User's Manual" [1]. Demos is a nonprocedural language, which treats arrays, vectors, and matrices as if they were single value variables. The first feature means that variables can be defined in any order, while the second feature allows arithmetic operations to be performed upon arrays just like scalars. The IECM has been design to take advantage of these features. The IECM is designed around input and summary tables, with character strings and other input parameters being used to choose configurations. The main input table is Coalchar_table, which contains the as-fired coal characteristics. The main summary table is Final_cost, which contains the main costs for each of the processes in the configuration loaded.

The IECM is composed of many modules which are separated into files. These files are shown below and have been group by technology type. The files with the ".mod" extension are located in the "srccom" (abbreviation for SouRce Code, COMMon) directory and contain source code which is used by the IECM and other models under development at CMU. These files contain the main performance algorithms which determine the major mass flows in the power plant. The files with the ".petc" extension are located in the "srcpetc" (SouRce Code, PETC) directory and contain all the source code for the models specifically developed for DOE/PETC. This arrangement allows us to simplify the maintenance of the models developed in Demos.

Main Files	All these files are required for the proper execution of the IECM.
initiecm.mod	Initializes many variables, functions, indices, constant, and models which are used by the other modules; therefore, it has to be the first file loaded into Demos. This file also makes it easy for someone to examine the performance of an individual technology without loading the power plant module. To examine a single technology, such as an ESP, the user would load "initiecm.mod" and "esp.mod." Demos would prompt the user for all the parameters which have to be defined for the ESP module to execute properly. This will be discussed in more detail in Section 3.3.
plant.mod	Creates an uncontrolled coal-fired power plant. It contains information and algorithms of the coal characteristics, the basic power plant, and emissions constraints. The boiler efficiency and flow rates for the coal flue gas, bottom ash and ash entrained in the flue gas are calculated. This file contains the main input table, Coalchar_table.
initiecm.petc	This file defines the Chemical Engineering cost index, the AFUDC function, and initializes many economic variables which are needed by the "economics.petc" file. This file has to be loaded before any other ".petc" files. If a user wishes to examine the economics of an individual technology, they would have to load "initiecm.mod," "initiecm.petc," and the economics model of the technology. If the user wants to calculate the fixed charge and

levelization factor they would load "economics.petc."

plant.petc Contains algorithms for the economics of the base power plant, cost of all reagents, and a few performance algorithms specific to the IECM model. The cost of delivered coal, capital cost, O&M cost of the basic power plant are defined in this file.

economics.petc Contains the economic algorithms for the entire plant. It calculates the fixed charged factor and levelization factor based upon the EPRI TAG guidelines [4, 5]. The main summary tables, Final_cost and Final_yr, are defined in this file, and it defines functions for doing a year-by-year analysis as specified in the EPRI TAG.

Pre-combustion Technologies

The following pre-combustion technologies are optional. Only one of these modules may be loaded at a time. All of them can be loaded anytime after the plant module. All of these files can be used with any of the Combustion, Particulate, Post-Combustion SO₂/NO_x, or Solid Waste modules.

coalclean.petc Contains the economic algorithms for a conventional coal cleaning plant. There are four levels of coal cleaning to chose from: 2, 3, 4, and conventional froth flotation. This has to be used in conjunction with the FORTRAN coal cleaning model described in Chapter 16.

hgms.petc Contains the performance and economic algorithms for the High Gradient Magnetic Separation (HGMS) process. This has to be used in conjunction with the FORTRAN HGMS model describe in Chapter 17.

froth.petc Contains the algorithms for the Coal-Pyrite Flotation process.

Post-Combustion Particulate Technologies

These modules are required by other desulfurization or NO_x removal modules. They are mutually exclusive.

esp.mod Contains performance algorithms for a cold side electrostatic precipitator.

esp.petc Contains the economic algorithms for a cold side electrostatic precipitator. It must be loaded after esp.mod.

fabfil.mod Contains performance algorithms for a fabric filter. It requires one of the following files: fgaphotofabfil.mod, fgpt1tofabfil.mod, fgpt2tofabfil.mod, or fgpt3tofabfil.mod. These files are used to place the fabric filter in the correct location relative to the other technologies being used. Fgaphotofabfil.mod is required if a fabric filter is used with a wet scrubber, or copper oxide. Fgpt2tofabfil.mod is required if a fabric filter is used with a spray dryer or NOXSO. Fgpt3tofabfil.mod is required if a fabric filter is used with the Electron Beam.

fabfil.petc Contains economic algorithms for a fabric filter. It must be loaded after fabfil.mod.

Post-Combustion SO₂/NO_x

Technologies	These technologies are mutually exclusive except that SCR can be used with either wetfgd or dryfgd modules and the dryfgd module is required for the Electron Beam module.
scr.petc	Contains the economic and performance algorithms for the Selective Catalytic Reduction process. It can be used with either the wetfgd or dryfgd module.
wetfgd.mod	Contains the performance algorithms for a wet lime or limestone scrubber. There two choices for oxidation, forced or natural. It can be used with and without bypass. It can be loaded with either the ESP for fabric filter modules.
wetfgd.petc	Contains the economic algorithms for the wet scrubber. It must be loaded after wetfgd.mod.
dryfgd.mod	Contains the performance algorithms for a spray dryer. There are three choices for bypass: no bypass, bypass based on maximum scrubber efficiency, or bypass based on exit temperature. It needs the fabric filter module.
dryfgd.petc	Contains the economic algorithms for a spray dryer. It must be loaded after dryfgd.mod.
noxso.petc	Contains the performance and economic algorithms for the NOXSO process. This also needs a fabric filter.
cuo.petc	Contains the performance and economic algorithms for the Copper Oxide Process and requires the fabric filter module.
ebeam.petc	Contains the algorithms for the Electron Beam Process, which requires the spray dryer and fabric filter module.
Solid Waste Disposal Technologies	This module have to be loaded if any of the post-combustion particulate or SO ₂ /NO _x control technologies are loaded.
solwaste.mod	Contains the algorithms for the solid waste disposal system used by the flue gas desulfurization and particulate control devices. It calculates the total sludge flow rate going to either a landfill or pond.
solwaste.petc	Contains the economic algorithms for the solid waste disposal. This must be loaded after "solwaste.mod." It can calculate the capital and O&M costs for ponding or landfill.

To make loading a configuration easier, there are several files which automatically load the files required for certain common configurations. They contain the same commands that would be typed inside of Demos. The first command names the project for the specific configuration. The first file read is the "initiecm.mod" file, followed by all the other modules needed. Finally, the "economics.petc" file is loaded. A description of these files is given below. To reconfigure the IECM, that is to change the post-combustion technology loaded into Demos, it is best to quit Demos. Start it up and read in a new configuration. In the future, it should be possible to change the configuration of a model without being require to re-start Demos.

loadcuo.mod	Used to load the default Copper Oxide system. It loads the copper oxide, fabric filter, and solid waste modules. The highest model in the hierarchy is "cuoff."
loaddry.mod	Used to load the default spray dryer system. It loads the spray dryer, fabric filter, and solid waste modules. The highest model in the hierarchy is "dryff."
loadbeam.mod	Used to load the electron beam system (Research-Cottrell Configuration). It loads the spray dryer, fabric filter, electron beam, and solid waste modules. The highest model in the hierarchy is "drybeamff."
loadnoxso.mod	Used to load the default NOXSO system. It loads the NOXSO, fabric filter, and solid waste modules. The highest model in the hierarchy is "noxsoff."
loadscrdry.mod	Used to load the spray dryer system with SCR. It loads all the files in "loaddry.mod" plus the SCR module. The highest model in the hierarchy is "dryscrff."
loadscrwet.mod	Used to load the wet FGD system with the ESP and SCR modules. It loads all the files in loadwet.mod plus the SCR module. The highest model in the hierarchy is "wetscresp."
loadwet.mod	Used to load the default wet FGD system. This loads the ESP, wet scrubber, and solid waste modules. The highest model in the hierarchy is "wetesp."

3.1 Summary Tables and Conventions

The IECM has some conventions for naming certain variables, tables and models, which makes it easier to use the IECM interactively. These conventions are not absolute rules that have to be followed when modifying the model, but they will make it easier to find specific items. The first convention that is followed is that all abatement technology modules have three submodels: input, performance and economic. In addition they may have three other submodels: link, uncertainty, and functions. An abatement technology may define additional submodels, but these six submodels have a specific function. Each of the following models has a specific suffix (except for uncertainty), which is attached to the abbreviation of the abatement technology. The suffices for these models are Functions, Inputs, Link, Perf, and Econ. The uncertainty model uses the prefix "Unc_." For example, the wet FGD module creates the following models: Wetmodelinputs, Wetlinks, Wetperf, Wetecon, and Unc_wet. This will be explained in more detail in the Model Hierarchy Section, 3.2.

All the abatement modules have three summary tables: input, performance, and economic. The input table contains the values for all the input parameters for a given technology. For example, the basic power plant model has a table PlantInputs. All the abatement technology modules use the name of the module and add the suffix "Inputs" for the name of the input table. The performance table contains the main performance parameters of the abatement module. For example, the basic power plant model has a table PlantChar. The name for this table is the name of the abatement module with the suffix "Char." The economic table contains the values for the economic parameters and uses the suffix "EconChar." For example, the basic power plant module has a table, PlantEconChar.

Every abatement module has the following variables listed below and they all have the

following format. The format is a generic variable name plus the name of the module. An example of the format appears below in parenthesis after each generic variable description. The economic and uncertainty variables are described in more detail in Chapter 5 and Section 3.3.8, respectively.

Ctl_	Uncertainty control variable which turns on the probabilistic variables for this module. (Ctl_wet)
Unctest	Uncertainty test variable. (Wetuncctest)
X	Energy penalty expressed as a fraction of the gross plant capacity. (Xwet)
ICF	Indirect charge factor of the capital cost. (ICFwet)
TDC	Direct capital cost of the abatement technology in millions of dollars. (TDCwet)
TCC	Total capital cost of the abatement technology in millions of dollars. (TCCwet)
TCCout	Total capital cost of the abatement technology in dollars per kilowatt. (TCCwetout)
TOandM	Total annual operating and maintenance cost of the abatement technology in millions of dollars per year. (TOandMwet)
TRRWOUtil	Total revenue requirement without utilities in millions of dollars per year. (TRRwetWOUtil)
UTIL	Cost of electricity and steam utility consumption in millions of dollars per year. (UTILwet)
TRR	Total revenue requirement including utilities in millions of dollars per year. (TRRwet)
ELEC	Total revenue requirement including utilities in mills per kilowatt hour. (ELECwet)

3.2 Model Hierarchy and Layout of IECM

As mentioned earlier in the tutorial, the IECM is arranged as a hierarchy of models. This hierarchy organizes the IECM so that input and output tables and variables are located easily. The file "initiecm.mod" sets up the basic hierarchy which is shown in Figure 3-1. As other files are loaded they move their input models into Allinputs and create other models on the same level as Allinputs, Plant, and Initiecm. Figure 2-6 shows the upper four levels of the hierarchy after the wet fgd, SCR, ESP, and solid waste modules are loaded into Demos.

The input model contains all the variables which are input parameters for the abatement technology. This model is placed into the Allinputs model as it is being loaded into Demos. This allows a user to find all the input parameters by using the show command on the Allinputs model. The input model of an abatement technology may also contain an uncertainty model. This model has the prefix "Unc_" attached to the abbreviation of the abatement technology. For example the wet FGD module creates the model, Unc_wet. The uncertainty model contains input uncertainty distributions which are being used in this module.

The link module is used to link the input and exiting flue gas variables of the abatement

Figure 3-1: Hierarchy Setup by Initiecm.mod

technologies with the main plant module. For example, all the abatement technology have a variable for the flue gas flow rate entering and exiting the process. These variables are redefined by the link module to correspond with the proper flue gas variables defined in the "plant.mod" file. The performance model contains all the performance variables, while the economic model contains the economic variables.

Figure 3-2 shows the variables which describe the flue gas at different points in the power plant. All these variables are defined in the model, Fluegas, in "plant.mod." These are the variables that the link models uses to connect the abatement technologies to the power plant. For example, the wet fgd module gets the composition of the flue gas entering the scrubber from Gasfg1 and redefines Gasfg2 as the gas exiting the scrubber. So the wet scrubber fits in between Gasfg1 and Gasfg2. Figure 3-3 shows the placement of SCR, ESP, and wet fgd modules. Figures 3-4 - 3-7 show the placement of the different technologies available in the IECM. The nomenclature used in naming these variables is defined below.

gas	Variable which contains the composition of the flue gas. The units are lb-mole/hr if the suffix "_coal" is not present. The variables without the suffix "_coal" are used by the abatement technologies to get the composition of the flue gas entering the plant.
temp_	Variable which contains the temperature of the flue gas in degrees Rankine.
ash	Variable which contains the composition of solid material entrained in the flue gas in units of tons/hr.
_coal	Variable which contains the composition of the flue gas in units of lb-mole/lb of fuel. These variables are used to determine the furnace efficiency for an uncontrolled power plant.
fdfan	Entering the forced draft fan
leak	Leaking across the air preheater
apho	Exiting the air preheater
furn	After combustion
econo	Exiting the economizer
fgaphi	Entering the air preheater on the flue gas side
fgapho	Exiting the air preheater on the flue gas side
fg1	First point down stream of the air preheater
fg2	Second point down stream of the air preheater
fg3	Third point down stream of the air preheater
stack	Gas entering the stack

3.3 Initiecm.mod File

The main purpose of this file is to initialize the hierarchy and define indices, functions, constants, and other variables which are used by the other modules in the IECM. This file does not define the highest level models, since that is usually defined by the loadfiles or by the user why they first start Demos. The models and variables will be described in the order of that appears in the file, except for the models Allinputs and Plant. These models are described below

Figure 3-2: Flue Gas Schematic Defined by Plant.mod

Figure 3-3: Flue Gas Schematic for a Wet FGD and SCR Configuration

Figure 3-4: Flue Gas Schematic for a Spray Dryer and SCR Configuration

Figure 3-5: Flue Gas Schematic for the Copper Oxide Configuration

Figure 3-6: Flue Gas Schematic for the NOXSO Configuration

Figure 3-7: Flue Gas Schematic for the Electron Beam Configuration

since they are just initialized in "initiecm.mod."

This file has to be the first file loaded into Demos. It contains many objects which are used by the other modules. These objects have been placed in a separate file, so that a single technology can be loaded into Demos. For example, if only the ESP performance model needs to be examined, then only the "initiecm.mod" and "esp.mod" files have to be loaded into Demos. Demos will automatically prompt the user for all the input variables which need to be defined for the model to execute properly. If the economics of the ESP is needed, then the files "initiecm.petc" and "esp.petc" need to be loaded. Demos will prompt the user for the additional input variables. If the algorithms for the fixed charge, levelization factor or a year by year analysis are desired, then the file "economics.petc" needs to be loaded. This has the advantage of reducing the number of objects in Demos which, save memory and increases the execution time.

Allinputs	This model serves as a place to hold all the input models for every abatement technology model. It is defined after Initiecm, but initially it does not contain anything. Input models from other files are placed in Allinputs by the open command. A user can then find any input model for a technology by typing, show Allinputs , which will list all the input models. Then the user can use the show command on these models to find any desired variable.
Plant	This model is defined to initialize the hierarchy of the power plant. The submodels initially are empty and are later filled with variables from the plant module. The submodels for Plant will be described in more detail in Chapter 4.

3.3.1 Initiecm Model

This model contains the models Indices, Configuration, Iecmfunction, Iecmconstant, and Nodevices. It is used to organize the other initialization models. These models will be describe below in the order that they are created in "Initiecm.mod."

3.3.2 Indices Model

The Indices model contain all the main indices used by the IECM. It is important to remember that character string values are case sensitive. If any index is incorrectly defined Demos will respond with an error message that a variable failed its check or that it is not defined correctly. The file "initiecm.petc" will open this model to add other indices which are used for the economics. These indices will be described in Section 3.4.

Coalindex	Row index for the tables coalchar_table, and cc_table. This variable is a vector which has a minimum value of [1], and each row in coalchar_table must have a entry in coalindex. If two rows are in coalchar_table then its value would be [1, 2]. If the FORTRAN coal cleaning model is used then this variable is redefined by the output file from the FORTRAN coal cleaning model.
Fuelchar_index	Index which labels the columns for the "as-received" fuel characteristics table, fuelcha. The labels are the higher heating value, carbon, hydrogen, oxygen, sulfur, nitrogen, ash, and moisture.

Cc_index	Index which labels the columns for the coal cleaning performance table, cc_char. The labels are the coal cleaning plant level, mass yield for the washing equipment, mass yield of the thermal drier, mass fraction of coal entering the thermal dryer, mass fraction of moisture evaporated, moisture content of the refuse stream, mass fraction of coal entering the froth flotation vessels, and the residence time of the coal in the froth flotation vessels.
Gascomp	Index which labels any air or flue gas variable. The labels are nitrogen, oxygen, water, carbon dioxide, carbon monoxide, sulfur dioxide, sulfur trioxide, nitrogen oxide, and nitrogen dioxide.
SolidsLabel	Index which labels any ash or FGD solid waste variable. The labels are ash from coal, calcium oxide, calcium carbonate, hydrated calcium sulfite, hydrated calcium sulfate, calcium sulfate, miscellaneous compounds, and water.

3.3.3 Configuration Model

This model contains variables which describe the current configuration that has been loaded. It is used primarily by the solid waste module and the Final_cost variable to pick the appropriate variables or algorithms. These variables are redefined by the abatement technology modules, so they should not be defined by the user. The table, configtable, creates a nice table of the current configuration.

So2Config	Variable with the name of the sulfur dioxide control technology being used. The valid entries presently are 'No SO2 Technology', 'Wet FGD', 'Spray Dryer', 'NOXSO', 'Copper Oxide', and 'Electron Beam'.
NOxConfig	Variable with the name of the nitrogen oxide control technology being used. The valid entries are 'No NOx Technology', ' ', and 'SCR.' If a combined SO ₂ /NO _x technology is used then this variable is set to ' '.
TspConfig	Variable with the name of the particulate control technology being used. The valid entries are 'No TSP Technology', 'ESP', and 'Fabric Filter'.
CCConfig	Variable with the name of the coal cleaning technology being used. The valid entries presently are 'No Coal Cleaning', 'Conv. Coal Cleaning', 'HGMS', 'Coal Pyrite'.
Control_index	Row index for the configuration table, configtable. This is just an array of spaces, so that the configuration table can be created.
Parametervalue	Column index for configtable. This is just a single dimensional array with the string, value, so that the configuration table can be created. In order for the configuration table to print properly, parametervalue must follow control_index.
Configtable	Table which displays the values for all the configuration variables listed above.

3.3.4 Iecmfunction Model

This model contains five models of user-defined functions, which are Miscfunctions, Gasfunctions, Ashfunctions, Uncertfunctions, and Enthalpy. Miscfunctions contains functions for the saturation pressure and temperature of steam, and for converting values from lbs/hr to

lbs/Mbtu of fuel. Gasfunctions contain functions and variables pertaining to the flue gas. Ashfunctions contains functions and variables pertaining to any ash or FGD waste stream. UncertFunc contains user-defined uncertainty distribution functions and other functions for manipulating probabilistic variables. Enthalpy contains functions for determining the enthalpy of difference species. The Iecmfunction model will contain the model Econfunctions after "initiecm.petc" is loaded. The content of the model is described in Section 3.4 and Chapter 5.

3.3.5 Miscfunction Model

This model contains functions for the saturation pressure and temperature of steam, and for converting values from lbs/hr to lbs/Mbtu of fuel.

Satpressteam(pres:numeric)

Function which determines the saturation pressure of steam in lbs/in^2 given a temperature in degrees Fahrenheit between 60 and 190. From the ASHRAE Handbook, 1981 Fundamentals.

Sattempsteam(temp:numeric)

Function which determines the saturation temperature of steam in degrees Fahrenheit given a pressure in lbs/in^2 between 0.25635 and 9.3495. From the ASHRAE Handbook, 1981 Fundamentals.

Quadeqn(a,b,c:numeric)

Function which determines the "positive" root of a quadratic equation of the form $ax^2 + bx + c$.

Quadeqneg(a,b,c:numeric)

Function which determines the "negative" root of a quadratic equation of the form $ax^2 + bx + c$.

Mbtu(input: numeric)

Function which divides the input value by the energy flow rate of the fuel in millions of Btu per hour. This function is useful to convert molar or mass flow rates of pollutants into per lbs per million Btu or to get the heat rate given the electrical energy out.

3.3.6 Gasfunctions Model

This model contains functions and variables which are used to analyze and modify any gas stream index by Gascomp. The term Gascomp stream is used to indicate that a variable is indexed by Gascomp. All Gascomp variables have units of lbmole/hr or lbmole/lb-fuel. It contains Remwatergas, Sumgas, Sumsox, Sumnox, Avgmolewt, acfm, Gasent, Energy, Energydiff, Cpgas, Btuf, Wgt_water, Fraccomp, Gas_param, and Gastable. For Gasent, Energy, Energydiff, Cpgas, and Btuf, the enthalpy is set to zero at 77 degrees F. The maximum temperature for which the correlations are valid is 3240 degrees Rankine. This corresponds to the highest valid temperature for sulfur dioxide.

Rewatergas

This variable removes the water vapor fraction of a Gascomp stream. All values are 1.0, except for the third element which is zero. Therefore, if this variable is multiplied by any Gascomp stream, the result is a stream except whose water composition is zero. This is useful to get the dry composition of a gas stream.

- Sumgas(gas:arraytype)**
Sums the components of a Gascomp stream.
- Sumsox(gas:arraytype)**
Sums the SO₂ and SO₃ components of a Gascomp stream.
- Sumnox(gas:arraytype)**
Sums the NO and NO₂ components of a Gascomp stream.
- Avgmolewgt(gas:arraytype)**
Determines the average molecular weight of a Gascomp stream.
- Acfm(gas:arraytype;temperature,pressure:numeric)**
Calculates the actual volumetric flow rate in cubic feet per minute of a Gascomp stream given the temperature in degrees Rankine, and the gauge pressure in inches of water. It is based on the ideal gas law.
- Gasent(tmpf:numeric)**
Determines the enthalpy in Btu/lbmole of the components in a Gascomp stream given a temperature in degrees Rankine. Algorithms based on the polynomial correlations for specific heat taken from Barin et al. [6, 7]. (Our June 1988 Quarterly Report to PETC discusses the integration and unit conversion of the polynomial correlations.)
- Energy(gas:arraytype;temp1:numeric)**
Determines the total energy flow rate in Btu per hour of a Gascomp stream given a temperature in degrees Rankine.
- Energydiff(gas:arraytype;temp1,temp2:numeric)**
Determines the change in the energy flow rate in Btu per hour of a Gascomp stream between two temperatures in degrees Rankine.
- CpGas(gas:arraytype;temp1,temp2:numeric)**
Determines the average specific heat in Btu/(lbmole F) of a Gascomp stream between two temperatures in degrees Rankine.
- Btuf(gas:arraytype; temp1,temp2:numeric)**
Determines the average energy absorbed or rejected per degree Rankine given a Gascomp stream between two temperatures in degrees Rankine.
- Wgt_water(gas:arraytype)**
Determines the mass fraction of water in a Gascomp stream in units of pound of water per pound of dry gas.
- FracComp(gas:arraytype;comp)**
Determines the molar fraction of the gas component, comp. Comp has to be either 'N2', 'O2', 'H2O', 'CO2', 'CO', 'SO2', 'SO3', 'NO', or 'NO2'. The single quotes are significant.
- Gas_param**
Index used by the Gastable function.
- Gastable(gas:arraytype)**
Creates a table of a Gascomp stream. The rows will be index by Gascomp. This table may not work correctly if new variables or exiting variables which were scalar have been defined to be arrays. The **reform** command can be used if this function does not produces the desired result.

3.3.7 Solidsfunctions Model

This model contains functions and variables which are used to analyze and modify any solids stream index by SolidsLabel. The term "SolidsLabel" stream is used to indicate that a variable is indexed by Solidscomp. All SolidsLabel variables have units of tons/hr. It contains Onlyash, Remwatersolids, Onlywaterinsolids, Sumsolids, Solids_param, and Solidstable.

- Onlyash** This variable re-normalizes a stream by removing everything but the coal ash from an SolidsLabel stream. It contains all zeros, except for the first element which is a one. Therefore, if this variable is multiplied by any SolidsLabel stream, the result contains only the ash component.
- Rewatersolids** This variable removes the water component of an SolidsLabel stream. It contains all ones, except for the last element which is a zero. Therefore, if this variable is multiplied by any SolidsLabel stream, then the result is normalized so that the water component is zero. This is useful to get the dry composition of a SolidsLabel stream.
- Onlywaterinsolids** This variable gives the water component of an SolidsLabel stream. It contains all zeros, except for the last element which is a one. Therefore, if this variable is multiplied by any SolidsLabel stream, the result only contains the water component.
- Sumsolids(Solids:arraytype)**
Determines the total mass flow rate of the SolidsLabel stream.
- Solids_param** Index used by the Solidstable function.
- Solidstable(Solids:arraytype)**
Creates a table of a SolidsLabel stream. The rows will be indexed by SolidsLabel. This table may not work correctly if new variables or exiting variables which were scalar have been defined to be arrays. The reform command can be used if this function does not produces the desired result.

3.3.8 Uncertfunctions Model

Uncertfunctions contains user-defined uncertainty distribution functions and other functions for manipulating probabilistic variables. The IECM defines five probability distribution functions which are useful in probability analysis: Half_normal, NegHalf_normal, Bernoulli, Wedge, and Triangular. This model also contains Cipct, Probcuts, Cdfnum, Cdf_param, Cdfperf, Changemean, and Changestdev. These are defined as follows.

- Half_normal(mean,sd)**
Calculates the positive half normal distribution given the mean and standard deviation. It returns the mean when evaluated deterministically.
- NegHalf_normal(mean,sd)**
Calculates the negative half normal distribution given the mean and standard deviation. It returns the mean when evaluated deterministically.
- Bernoulli(p:determ)**
Calculates the discrete binary distribution, used by the Triang function. It returns zero if p is less than or equal to 0.5 and 1 if p is greater than 0.5 when it is evaluated deterministically.
- Wedge(a,b:prob)** Calculates a wedge shaped density function increasing from a to b. It returns

the median when evaluated deterministically.

Triang(a,b,c: scalar)

Calculates a triangular distribution bound by a and c with a mode at b. Currently, when this function is evaluated deterministically it returns the a vector with the same number of values as Samplesize. This will be fixed in future versions of Demos.

Cipct

Index which labels the probability cut points of Probcuts. This index is indirectly used by Cdfnum and Cdfperf.

Probcuts

Index used by Cdfnum and Cdfperf, which contains the percentiles to be extracted from a probabilistic variable.

Cdfnum(Pvariable:prob)

Creates a table of the percentile values specified in Probcuts for variables which are index by any other variables. This is typically used with variables index by Cost_yr.

Cdf_param

Index used by the Cdfperf function.

Cdfperf(Pvariable:prob)

Creates a table of the percentiles values specified in Probcuts for variables which are not indexed by any other variables.

Changemean(mean,stdev:numeric)

Calculates the geometric mean (median) given the mean and the standard deviation of a lognormal distribution.

Changestdev(mean,stdev:numeric)

Calculates the geometric standard deviation given mean and the standard deviation of a lognormal distribution.

All the uncertainty variables used in a particular module are grouped into a model which has the prefix "Unc_" attached to the abbreviation of the module. The model has been designed so that the uncertainty variables can be activated all at once or by individual technologies. This is accomplished with a global control variable, Ctl_Global, and the uncertainty control and test variables of each module. The uncertainty control variables have names beginning with "Ctl_" attached to the module abbreviation. For example, the control uncertainty variable for the wet FGD module is Ctl_Wetfgd. To activate all the uncertainty variables, set Ctl_Global to 'all on' (Note: These values are case sensitive and the single quote mark is significant.). To deactivate all of them, set Ctl_Global to 'all off.' To only use uncertainty variables for certain abatement modules, set Ctl_Global to 'on,' and the uncertainty control variable of the specific abatement module to 'on.' The uncertainty test variable has the suffix, "unctest." This variable has the value of 1 for probabilistic analysis and a value of 0 for deterministic analysis. The uncertainty test variable makes it easier to write the definition of the uncertainty variables.

Every variable with a dimensional probability distribution has an uncertainty variable with the same name plus a prefix, "Uf." For example, the variable, xwet, has an uncertainty variable called ufxwet. When Ctl_Global = 'all on,' (or Ctl_Global = 'on,' and Ctl_wetfgd = 'on'), then ufxwet has a normal distribution with a mean of one and a default standard deviation of 10%; otherwise, it has a value of one. The definition of ufxwet, Wetuncctest, xwet are shown below.

Variable Xwet

Title: Wet FGD Energy penalty

Units: fraction

Description: Energy penalty for steam and electricity usage

Definition: (fgd_reheat + xfgdm) * ufxwet

Variable Wetunctest

Definition: if Ctl_global = 'all on' OR (Ctl_global = 'on' AND
Ctl_wet = 'on') then 1 else 0

Variable ufxwet

Definition: if wetunctest then normal(1,.1) else 1

A variable can also have a non-dimensional uncertainty distribution which completely replaces the deterministic value. For this case, a separate uncertainty variable is not defined; instead, the uncertainty distribution is defined inside the definition for the variable. The uncertainty control variables will still work if they are defined as shown in the example below. The example below shows the definition for Xwet for a non dimensional uniform distribution. In this example, the probabilistic value is specified by a uniform distribution and is not affected by the performance of the wet FGD.

Variable Xwet

Title: Wet FGD Energy penalty

Units: fraction

Description: Energy penalty for steam and electricity usage

Definition: if wetunctest then uniform(.01,.02) else (fgd_reheat + xfgdm)

The variables which have default uncertainties in this version of the IECM are not the only variables which can be uncertain, nor are they required to be uncertain. Almost any variable in the IECM can be uncertain. All one has to do is create a new dimensional uncertainty distribution and multiply it by the variable to be uncertain. Or, a non-dimensional uncertainty distribution can be added to any variable that currently exists. It is recommended that the user follow the conventions described above. It is also possible to delete the uncertainty distributions currently defined in the IECM by deleting the uncertainty variables from the source code files.

3.3.9 Enthalpy Model

This model contains the enthalpy functions for the following compounds: CH₄, CO, CO₂, COS, Cu, CuO, CuSO₄, H₂, H₂O, H₂S, N₂, NH₃, NO, NO₂, O₂, S₂, SO₂, SO₃, and Al₂O₃. The names of the enthalpy function for a specific compound is just the compound name with the prefix, H. All functions require the input temperature to be in degrees Fahrenheit and return the enthalpy in units of Btu/lbmole, except for Hal2o3 which returns Btu/lb. Algorithms based on the polynomial correlations for specific heat are taken from Barin et al. [6, 7]. Our June 1988 Quarterly Report to PETC discusses the integration and unit conversion of the polynomial correlations. All enthalpies are set to zero at 77 degrees Fahrenheit. The maximum temperature in degrees Fahrenheit for the correlation for the compounds listed above are 3140, 4040, 4040, 2780, 1983, 1986, 1480, 4940, 4040, 2780, 4040, 3140, 4940, 4940, 4940, 3140, 2780, 3140, and 2780, respectively.

3.3.10 Iecmconstants Model

This model contains two variables which are used by other modules in the IECM. These are Incheswtopsi and Mole_wgt_gas. Incheswtopsi is a constant which converts pressure from inches of water to pounds per square inch. Mole_wgt_gas is an array of constants which are the

molecular weights of the compounds in Gascomp. This array converts the molar flow rate of Gascomp streams into mass flow rates when multiplied by a Gascomp stream.

3.3.11 Nodevices Model

This model defines as zero many of the variables needed by other abatement modules. These variables then are redefined by the abatement modules if they are loaded into the IECM. The primary purpose of this model is to prevent Demos from complaining that certain variables are not defined when a module is not loaded. For example the variable, Mwnet, must have the energy penalty variables for all abatement technologies defined in order to evaluate properly. This model groups all variables by the technology that would define those variables if they were loaded into Demos. The prefix "No" is attached to the abbreviation of each abatement technology. For example, the model Noesp contains the variables which would be defined by the ESP module if it were loaded. The file "initiecm.petc" opens some of the models created in Nodevices to add other variables which are needed by the economics model. These other models will be discussed in the next section.

3.4 Initiecm.petc File

The main purpose of this file is to define the cost scaling variables and to initialize variables which are used by the economics module. This file has to be loaded into Demos before any other ".petc" files. With "initiecm.mod" plus this file, the economics of a single technology can be examined without loading "plant.mod" or "plant.petc." Demos will prompt the user for the additional input variables needed for the model to execute properly. If the algorithms for the fixed charge factor, levelization factor, or year-by-year analysis are desired, the file "economics.petc" needs to be loaded. As mentioned previously, this has the advantage of reducing the number of objects in Demos, which saves memory and decreases the execution time. The hierarchy, shown in Figure 3-8, changes slightly when "Initiecm.petc" is loaded since it creates the model, Econfunctions, inside of Iecmfunctions, and also creates four models (Nocuo, Noebeam, Nonoxso, Noscr) inside of Nodevices.

3.4.1 Indices Model

This section describes the variables which are defined to scale the capital and operating costs to different year's dollars using the mid-year Chemical Engineering Plant index. New years can be added by adding a new value to the arrays in Eng_indices and Plant_indices and increasing the Years array by number of values added.

Years	Index with the range for the Chemical Engineering Cost indices. Currently set to an array from 1977 to 1989.
Cost_yr	Vector used to select the year in which dollars are reported. The default value is [1988]. It must be a vector, but it does not have to be an integer value. If a non-integer value is used, Ppcost_index and Cost_index will interpolate between the closest two integers. This estimates the end or beginning of year values for the Chemical Engineering Cost index.
Eng_indices	Values of the Chemical Engineering cost index for the years specified in the index, Years. Number of entries in this index must match the number of years in the index, Years. This index is used in by all the economic models,

Figure 3-8: Hierarchy Setup by Initiecm.mod and Initiecm.petc

except for the capital cost of power plants before 1984.

Plant_indices	Values of the Handy-Whitman electric utility cost index for the years specified in the index, Years. Number of entries in this index must match the number of years in the index, Years. This index is used to scale the power plant capital costs.
Ppcost_index	Scalar with the Handy-Whitman cost index for the year specified by Cost_yr.
Cost_index	Scalar with the standard Chemical Engineering cost index for the year specified by Cost_yr.

3.4.2 Econfunctions Model

This model contains various economic functions and is located in the Iecmfunction model. These functions are created in this file so that the AFUDC function can be defined, which is used by all the abatement technologies to calculate the AFUDC cost. Other functions are loaded into this model via the "economics.petc" file.

afudc(year,interestrate,infla)

Calculates the Allowance for Funds Used During Construction, (AFUDC), based on the length of the project in years, the interest rate, and the inflation rate. The length of the project must be an integer number of years, between 1 and 10. The algorithm assumes that the money is spent uniformly at discrete mid-year intervals.

3.4.3 Nodevices Model and Plant Model

This opens the Nodevices model and several submodels to define several variables which are needed by the economics module. It also creates several new models for selected environmental control technologies. These are Copper Oxide, Electron Beam, NOXSO, and SCR. The plant model is also open to define several economic variables which are needed by the economics module.

4. Basic Power Plant Module

This chapter describes the files, models, and variables for the basic plant module. The analytical documentation can be found in "Modelling and Assessment of Advanced Processes for Integrated Environmental Control of Coal-Fired Power Plants," by Rubin, et. al. [2]. Many of the variables described below have a one to one correspondence with the nomenclature used the analytical documentation. Where this is the case, the nomenclature and chapter number appear in the description of the variable, surround by parenthesis.

The basic power plant module consists of two files, "plant.mod" and "plant.petc." "Plant.mod" creates an uncontrolled coal-fired power plant. "Plant.petc" contains the economics for the uncontrolled coal-fired power plant and modifies some of the plant performance parameters. Each file is described in the following sections.

Every input parameter in the IECM is checked to see that it falls within a specified range. The IECM use the check attribute in Demos for input and other variables to ensure that their values are within a specified range. If the value of a variable is outside its range, Demos will issue a message stating that the variable failed to satisfy its check, and let the user decided to ignore or correct the problem. An example of this was shown in the Tutorial Chapter. Many times the IECM also uses the range as a method to catch typographical errors. For example, the steam cycle heat rate could theoretically be lower than 6000; however, it is not likely. So the value 6000 is used to ensure that a typographical error has not occurred. The values of the range are not constants and can be changed by the user.

4.1 Plant.Mod File

As mentioned previously, "plant.mod" creates an uncontrolled coal-fired power plant. The main parameters that "plant.mod" calculates are the heat rates, net capacity, furnace efficiency, mass flow rate of coal, air into the furnace, flue gas exiting the furnace, bottomash, flyash, and the unregulated emissions of SO₂, NO_x, and TSP. It creates four input parameters models (Basecontrols, Coal_inputs, Baseplinputs, and Airemissions) which are put into Allinputs. Finally, it defines variables for the models in Plantperf, which was created in "Initiecm.mod." Figure 4-1 shows the hierarchy when just the "initiecm.mod" and "plant.mod" files have been loaded into Demos. The models and variables defined in "plant.mod" are defined in the order they appear in the file.

4.1.1 Basecontrols Model

This model contains decision variables which take character strings as values. They are used to change some of the options in a power plant, such as the furnace firing method, how the FGD sludge is disposed, or whether the fly ash is mixed with the FGD sludge or sent to the bottom ash pond. The variable, Basectlinputs, creates a table of the values of these variables. This model contains Furnacectl, Disposal, Mixedctl, Ctl_global, Basectllabels, Basectl_param, and Basectlinputs.

Furnacectl	Determines the firing method for the furnace. The valid options are 'cyclone', 'tangential', and 'wall.' The default is 'tangential.'
Disposal	Control parameter used to select the waste disposal method. The valid options are 'ponding' and 'landfill'. The default is 'landfill.' Disposal

Figure 4-1: Hierarchy of IECM created by Plant.mod after Initiecm.mod has loaded into Demos

	should only be ponding if So2config = 'NO SO2 Technology' or 'Wet FGD.'
Mixedctl	Controls if the flyash is mixed with the FGD waste. For a Spray Dryer, and Electron Beam this variable should equal 'yes,' since the FGD waste and flyash are collected together in the particulate device. The default is 'yes.'
Ctl_Global	The global uncertainty control variable. Setting to 'all on' turns on all uncertainty variables, to 'all off' turns off all uncertainty variables to 'on' turns on only those uncertainty variables, whose module control variable is turned on. The default is 'all off.'
Basectllabels	Row labels for Basectlinputs.
Basect_param	Column label for Basectlinputs. This has to follow Basectllabels in order for Basectlinputs to print out correctly.
Basectlinputs	Summary table of the decision variables defined in this model.

4.1.2 CoalInputs Model

This model contains all contains the main variables related to the coal. The model contains the main input table, Coalchar_table, and the variable used to access the data in this table, Coals. It also serves as an interface between the different coal cleaning technologies and the rest of the IECM. For conventional coal cleaning, the FORTRAN coal cleaning program creates a file with the variables described below and other variables described in Section 4.2.1. The HGMS and Coal Pyrite Flotation modules redefine Coalchar_table to be an equation which creates a table of the raw and cleaned coal characteristics. New variables for the raw coal characteristics are created in the HGSM and Coal Pyrite Flotation modules. This will be described in more detail in Chapters 17 and 18. Other variables are put into this model by the file "plant.petc." These are described in Section 4.2.1.

Coals	Variable used to select the coal or coals from Coalchar_table. Must have a value between 1 and the maximum value of coalindex. It is possible to change Coals into a vector of values so that the results of multiple cleaning levels can be easily seen. For example, Coals can equal [1 2 3]. The default is 1, with a valid range of 1 to the maximum value in Coalindex.
Coalchar_table	This is the input table which contains the characteristics of the coal being used in the IECM. Each row contains the "as-received" composition of the input coal cleaned to different levels. The first column is the higher heating value in Btu/lb, the second column is the carbon content, the third column is the hydrogen content, the fourth is the oxygen content, the fifth is the sulfur content, the sixth is the nitrogen content, the seventh is the ash content, and the last column is the moisture content. This table is redefined if the HGMS or Coal Pyrite Flotation module are used. The default is table(Coalindex,fuelchar_index)(10194. .5695 .0370 .0719 .0436 .0112 .1438 .1230), which is a representative Illinois #6 coal. The range checks for the coal composition occurs in Fuel_char.

4.1.3 Basepltinputs Model

The Basepltinputs model has the input parameters for the base power plant. It contains Mwg, Hrsteam, Cf, Exair_boiler, Exair_airheater, Orgtemp_furn, Orgtemp_econo, Orgtemp_fgapho, Xpulverizers, Xsteampumps, Xfans, xtowers, Xmisc, Unacc_loss, Pltinputlabel, pltvalue, and Plantinputs.

Mwg	Gross electrical capacity in MW at the generator. The default is 540 MW with a valid range between 100 and 1000. (MW ₂ , 6)
Hrsteam	Steam cycle heat rate in Btu/kWh. The default is 7880, with a valid range between 6000 and 10000. (CF, 6)
Cf	Capacity factor as a fraction of the power plant. The default is 0.65 with a valid range between 0.15 and 1.00. (CF, 6)
Exair_boiler	Excess air as a fraction of the stoichiometric air used for combustion. The default is 0.15 for cyclone firing and 0.2 for wall or tangential firing. It has a valid range between 0.1 and 0.3.
Exair_airheater	Air that leaks at the air preheater as a fraction of the stoichiometric air used for combustion. The default is 0.19 with a valid range between 0 and 0.4.
Orgtemp_furn	Flue gas temperature in degree Fahrenheit in the furnace. The default is 2500 with a valid range between 1600 and 3600.
Orgtemp_econo	Flue gas temperature in degree Fahrenheit after the economizer. The default is 700 with a valid range between 600 and 800.
Orgtemp_fgapho	Flue gas temperature in degree Fahrenheit after the air preheater. The default is 300 with a valid range between 250 and 400.
Xpulverizers	Energy penalty for the pulverizers as a fraction of the gross electrical capacity. The default is 0.006 with valid range between 0.001 and 0.01.
Xsteampumps	Energy penalty for the pumps in the steam cycle as a fraction of the gross electrical capacity. The default is 0.0065 with valid range between 0.005 and 0.007.
Xfans	Energy penalty for the forced draft and primary air fans as a fraction of the gross electrical capacity. The default is 0.015 with valid range between 0.007 and 0.04.
Xtowers	Energy penalty for the cooling system as a fraction of the gross electrical capacity. The default is 0.018 with valid range between 0.004 and 0.02.
Xmisc	Energy penalty for miscellaneous equipment as a fraction of the gross electrical capacity. The default is 0.013 with valid range between 0 and 0.02.
Xunacc_loss	Unaccounted for losses in the furnace as a fraction of the energy entering the furnace in the fuel. The default is 0.005 with a valid range between 0 and 0.01.
Pltinputlabel	Row labels for the Plantinputs table.
pltvalue	Column label for the Plantinputs table, which must follow Pltinputlabel.
Plantinputs	Summary table of the input parameters defined in this model.

4.1.3.1 Unc_plant Model

This model contains the dimensional uncertainty distributions for the Plantinputs model. It contains Ctl_plant, Pltunc test, Ufcf, Ufhr, Ufexair_boiler, and Ufexair_airheater.

Ctl_plant	Uncertainty control variable for the uncertain variables in plantinputs. The default value is 'on,' but it can also be 'off.'
Pltunc test	Uncertainty test variable which determines whether the uncertain variables in this model are probabilistic or deterministic. This is an equation so it should not be changed.
Ufcf	Dimensional uncertainty variable for the capacity factor. The default is a normal with mean of 1 and a standard deviation of 0.07.
ufhr	Dimensional uncertainty variable for the steam cycle heat rate. The default is a negative half normal with mode of 1 and a standard deviation of 0.018.
Ufexair_boiler	Dimensional uncertainty variable for the excess air in the boiler. The default is a normal with mean of 1 and a standard deviation of 0.025.
Ufexair_airheater	Dimensional uncertainty variable for the leakage across the air preheater. The default is a normal with mean of 1 and a standard deviation of 0.025.

4.1.4 Airemissions Model

This model has the input parameters for the ambient air conditions, emission constraints, and emissions factors for the power plant. It contains Tai_air, Pai_air, Humdt, Noxef, Fao, So2ashret, Fso, Fracco, Cinrefuse, Fracc, Frasco2, Fracno, Emissconso2, Emissconsnox, Emissconsash, Air_emislabels, Airvalue, and Airemisinputs.

Tai_air	Ambient air temperature in degree Fahrenheit. The default is 80 with a valid range of 0 to 110.
Pai_air	Ambient air pressure in pounds per square inch. The default is 14.696 with a valid range of 12 to 15.
Humdt	Specific humidity of the ambient air in pounds of water per pound of dry air. The default is 0.018 with a valid range of 0 to 0.03. (ω , 6)
Noxef	Nitrogen oxide emission factor in pounds equivalent NO_2 per ton of coal. This factor is used to determine the amount of nitrogen oxides that are formed in the boiler. For bituminous or subbituminous coal the default is 37 for cyclone-fired, 21 for wall-fired, and 15 for tangential-fired furnaces. For lignite, it is 17 for cyclone-fired, 12 for wall-fired, and 8 for tangential-fired. The valid range is of 5 to 40. ($\text{NO}_{x_{\text{ef}}}$, 6)
Fao	Fraction of ash entering the flue gas stream; the rest is bottom ash. The default is 0.25 for cyclones. For the other firing methods, it is 0.75 for bituminous, 0.7 for subbituminous, and 0.65 for lignite. The valid range is 0.2 to 0.8.
So2AshRet	Fraction of sulfur in the bottom ash. The default is 0.025 for bituminous, 0.125 for subbituminous, and 0.25 for lignite. The valid range is 0 to 0.5
Fso	Fraction of sulfur entering the flue gas. The default is 1 minus So2ashret with a valid range of 0.5 to 1
Fracco	Fraction of carbon in coal which is converted to carbon monoxide. The

	default is 0 with a valid range of 0 to 0.05.
Cinrefuse	Fraction of unburned carbon in bottom and fly ash. The default is 0 with a valid range of 0 to 0.01.
FracC	Total fraction of unburned carbon. The default is an equation based on the unburned carbon the the ash, the ash and carbon content of the coal. The default is 0 with a valid range of 0 to 0.05.
FracSo2	Fraction of sulfur in the flue gas which forms SO ₂ . The default is 0.97 with a valid range of 0.95 to 1. (f _g , 6)
Fracno	Fraction of total nitrogen oxides which form NO. The default is 0.95, with a valid range of 0.95 to 1.
Emisconsso2	Sulfur dioxide emission constraint defined by the 1979 RNSPS in pounds per million Btu. This depends on the potential emissions of the raw coal not including any sulfur retention effects. The valid range is of 0 to 15. (EC _s , 6)
Emisconsnox	NO _x emission constraint in pounds per million Btu. The default is 0.08; however, an algorithm for the current standard is in the description. The valid range is of 0 to 1. (EC _n , 6)
Emisconsash	Total suspended particulate (TSP) standard in pounds per million Btu. The default is 0.03 with a valid range of 0 to 1. (EC _a , 6)
Air_emislabels	Row labels for Airemisinputs.
Airvalue	Column label for Airemisinputs, which must follow Air_emislabels in order for Airemisinputs to print out correctly.
Airemisinputs	Summary table of the input parameters defined in this model.

4.1.5 Plant Model

This model sets up an uncontrolled coal fired power plant. The file "plant.mod" contains just the performance models for the base power plant. Additional models are entered by the file "plant.petc."

4.1.6 Plantperf Model

This model determines the coal consumption, net heat rate, and net capacity. It contains Hrcycle, Mfuel, Hr, Mwwopece, Mwnet, Hrnet, Hoursannual, Mwhannual, Plant_index, Plant_param, Plantchar. It also contains following models Fuel_char, Efficiency, Airheater, Fluegas, Tempfluegas, Ashfluegas, Fgsummary, and Emissions. These models calculate the flow rates of coal, bottom ash, fly ash, combustion air, and flue gas, and determine the performance of the furnace, air preheater, and overall plant.

Hrcycle	Gross cycle heat rate in Btu/kWh. This includes the boiler efficiency.
Mfuel	Coal consumption in tons per hour at a 100 percent capacity factor.
Hr	Net heat rate in Btu/kWh for an uncontrolled coal-fired power plant. This does not include the energy penalties from environmental control equipment.
Mwwopece	Net electrical capacity in MW for an uncontrolled coal-fired power plant. This includes the internal energy penalties for the pulverizers, steam pumps,

forced draft fans, primary air fans, cooling system, and miscellaneous equipment but it does not include the energy penalties for abatement technologies..

Mwnet	Net electrical capacity in MW for the controlled power plant. This does include the energy penalties for the environmental control equipment.
Hrnet	Net heat rate in Btu/kWh for the controlled power plant. This does include the energy penalties for the environmental control equipment.
Hoursannual	Annual hours of operation.
Mwhannual	Net megawatt hours of the controlled power plant.
Plant_Index	Row index for PlantChar.
Plant_param	Column index for Plantchar, which must be defined after Plant_index for Plantchar to print properly.
PlantChar	Summary table for the variables defined in this model.

4.1.7 Fuel_char Model

This model extracts the coal characteristics from Coalchar_table for the coal chosen and determines the rank of that coal. It contains Fuelchar, Fuelparameter, Fuelsummary, Hhv, Cfrac, Hfrac, Ofrac, Sfrac, Ashfrac, H2ofrac, H2ofR, Hhvmmf, and Coalrank.

Fuelchar	Array which extracts the coal characteristics from Coalchar_table for the coal chosen by the variable Coals. If Coals = 1, then Fuelchar contains the first row of Coalchar_table. If Coals = [1 2], then Fuelchar contains the first and second rows. This array is index by Fuelchar_index.
Fuelparameter	Column label for Fuelsummary.
Fuelsummary	Table of the coal characteristics in Fuelchar.
Hhv	Higher heating of the coal being burned, with a valid range of 5000 to 15000.
Cfrac	Carbon fraction of the coal being burned, with a valid range of 0.3 to 0.8.
Hfrac	Hydrogen fraction of the coal being burned, with a valid range of 0.01 to 0.05.
Ofrac	Oxygen fraction of the coal being burned, with a valid range of 0.01 to 0.12.
Sfrac	Sulfur fraction of the coal being burned, with a valid range of 0.001 to 0.05.
Nfrac	Nitrogen fraction of the coal being burned, with a valid range of 0.001 to 0.02.
Ashfrac	Ash fraction of the coal being burned, with a valid range of 0.04 to 0.20.
H2ofrac	Moisture fraction of the coal being burned, with a valid range of 0.04 to 0.4.
Hhvmmf	Higher heating value of the raw coal on a mineral and moisture-free basis. It is used to calculate the rank of the coal based on the Parr formula [8].
CoalRank	Rank of the coal based on the ASTM criteria [8]. Bituminous equals 1, subbituminous equals 2, and lignite equals 3.

4.1.8 Efficiency Model

This model determines the furnace efficiency based on the algorithms from Babcock and Wilcox [8]. This model determines the dry flue gas, moisture, carbon, and radiation losses to calculate the furnace efficiency. It uses the functions in the model Gasfunctions. The flue gas variables which are used in this model are in units of lbmole/lb of coal. This allows the efficiency to be calculated without knowing the amount of coal entering the furnace. It contains Dryflue_loss, H2o_loss, Carbon_loss, Radiation_loss, Boiler_eff, Boiler_labels, Boilervalue, and Boilerchar.

Dryflue_loss	Energy loss from the dry flue gas exiting the air preheater based on the uncorrected flue gas temperature, which is calculated in the Airheater model.
H2o_loss	Energy loss from the moisture in the air and coal and the hydrogen in the fuel, assuming a latent heat of evaporation of 1040 Btu/lb.
Carbon_loss	Energy loss due to incomplete combustion, assuming a carbon higher heating value of 14,100 Btu/lb and an energy loss from carbon monoxide of 9755 Btu/lb.
Radiation_loss	Energy loss due to radiation from the boiler, based upon a curve fit of Figure 5, Page 6-9 in Combustion Engineering [9].
Boiler_eff	The efficiency of the boiler (furnace) which is 1.0 minus all the losses defined above plus the unaccounted for losses defined in the AirEmissions Model.
Boiler_labels	Row labels for Boilerchar.
Boilervalue	Column labels for Boilerchar, which must follow Boiler_labels for Boilerchar to print correctly.
Boilerchar	Summary table of the variables defined in this model, plus other variables which are necessary to calculate the boiler efficiency.

4.1.9 Airheater Model

The Airheater model is main purpose is to calculate the uncorrected flue gas temperature and the size of the air preheater. The uncorrected temperature is based on the specified exit temperature of the flue gas, the ambient air temperature, and the leakage across the air preheater. The model uses the "_coal" flue gas variables so that the performance for the uncontrolled power plant can be determined. This allows comparisons to be made with abatement technology that is upstream of the air preheater. This model divides the heat transfer and leakage across the air preheater into two parts (see Figure 4-2). The heat transfer occurs in a counter flow heat exchanger, while the leakage occurs downstream of the air preheater. This model contains Taph_uncorrected, Q_Air_orig, Orgtemp_apho, Ua_orig_air, Aph_index, Aphorig_param, Aphorigchar.

Taph_uncorrected	Temperature in degrees Fahrenheit of the flue gas before the leakage air is mixed with the flue gas for the original or uncontrolled power plant. This equals the reciprocal of the energy released per degree for the flue gas entering the air preheater times the energy absorbed by the leakage air plus the exit temperature of the flue gas.
Q_air_orig	Energy flow rate in Btu/hr across the air preheater for the uncontrolled power plant. This equals the energy of the flue gas and air entering the air

Figure 4-2: Schematic Diagram of the Air Preheater

	preheater minus the energy of the total flue gas exiting the air preheater.
Orgtemp_apho	Furnace air inlet temperature in degrees Fahrenheit. This equals the reciprocal of the energy absorbed per degree for the air entering the furnace time the energy transfer across the air preheater plus the inlet air temperature.
Ua_orig_air	Overall heat transfer coefficient times the area. This equals the energy transfer across the air preheater divided by the log mean temperature difference
Aph_index	Row labels for Aphorigchar
Aphorig_param	Column label for Aphorigchar, which must follow Aphorig_param so that Aphorigchar prints out correctly.
Aphorigchar	Summary table of the variables in this model and other variables which affect the performance of the air preheater.

4.1.10 Fluegas Model

This model determines the flow rates for the combustion air, leakage across the air preheater, and the flue gas. This model assumes that the composition of dry air is 21% oxygen and 79% nitrogen. The composition of the flue gas is based upon the emissions factors specified in the Airemissions Model. The variables with the suffix "_coal" are in units of lbmole/lb-fuel and are reserved for calculating the performance of the boiler efficiency and air preheater. The definition of these variables should not be changed by the other modules. The variables without the suffix are in units of lbmole/hr and can be redefined by other modules. Figure 3-2 shows the locations in the ductwork that correspond to the variables in this model. All the variables in this model, except O2fracair and O2needed, are arrays which are index by Gascomp. This model contains O2fracair, O2needed, Aircompgas, Gasapho_fuel, Gasapho, Gasfurn_fuel, Gasfurn, Gasecono_fuel, Gasecono, Gasfgaphi_fuel, Gasfgaphi, Gasleak_fuel, Gasleak, Gasfdfan_fuel, Gasfdfan, Gasfgapho_fuel, Gasfgapho, Gasfg1_fuel, Gasfg1, Gasfg2_fuel, Gasfg2, Gasfg3_fuel, Gasfg3, Gasstack_fuel, and Gasstack.

This model also keeps track of the changes in the flue gas at different points in the plant as other modules are loaded. These modules may redefine the flue gas variables in lbmole/hr, (ie the variables without the suffix "_coal." For example, the wet FGD module gets the flue gas composition entering the scrubber from Gasfg1 and redefines Gasfg2 and Gasfg3, which can be seen in Figure 3-3. However, these variables stay in the Fluegas model, so a user can trace how the composition of the flue gas changes as it passes through the plant.

O2fracair	Oxygen content as a fraction of moist air, based on a dry oxygen content of 21% and the moisture content of the ambient air, Humdt.
O2needed	Atmospheric oxygen in lbmole/lb-fuel required for stoichiometric combustion of the carbon, hydrogen, and sulfur minus the oxygen in the fuel.
Aircompgas	Composition of combustion air as a fraction, based on O2fracair, a dry nitrogen content of 79%, and the moisture content of the ambient air, Humdt. The significant difference between this variable and O2fracair is that this is an array index by Gascomp.
Gasapho_fuel	Combustion air into the furnace in units of lbmole/lb-fuel. The composition

	of air, Aircompgas, multiplied by the total amount of air, $(1 + \text{Exair_boiler}) * \text{O2needed} / \text{O2fracair}$, gives the total flow rate of nitrogen, oxygen, and water in the air.
Gasapho	Combustion air in lbmole/hr, which equals Gasapho_fuel times the coal flow rate.
Gasfurn_fuel	Flue gas after complete combustion of the coal in units of lbmole/lb-fuel. This is equals the combustion air into the furnace, Gasfurn_fuel, plus the result of burning the coal.
Gasfurn	Flue gas after complete combustion of the coal in units of lbmole/hr, which equals Gasfurn_fuel times the coal flow rate.
Gasecono_fuel	Flue gas exiting the economizer in lbmole/lb-fuel, which equals Gasfurn_fuel.
Gasecono	Flue gas exiting the economizer in lbmole/hr, which equals Gasfurn.
Gasgaphi_fuel	Flue gas entering the air preheater in lbmole/lb-fuel, which equals Gasecono_fuel.
Gasgaphi	Flue gas entering the air preheater in lbmole/hr, which equals Gasecono.
Gasleak_fuel	Air leaking across the air preheater in lbmole/lb-fuel, which is the air composition multiplied by the total air leaking, $\text{Exair_airheater} * \text{O2needed} / \text{O2fracair}$.
Gasleak	Air leaking across the air preheater in lbmole/hr, which equals Gasleak_fuel times the coal flow rate.
Gasdfan_fuel	Air entering the forced draft and primary air fans in lbmole/lb-fuel, which equals Gasleak_fuel plus Gasapho_fuel.
Gasdfan	Air entering the forced draft and primary air fans in lbmole/hr, which equals Gasleak plus Gasapho.
Gasfgapho_fuel	Flue gas exiting the air preheater in lbmole/lb-fuel, which equals Gasgaphi_fuel plus Gasleak_fuel.
Gasfgapho	Flue gas exiting the air preheater in lbmole/hr, which equals Gasgaphi plus Gasleak.
Gasfg1_fuel	Flue gas at point 1 in ductwork in lbmole/lb-fuel, which equals Gasfgapho_fuel.
Gasfg1	Flue gas at point 1 in ductwork in lbmole/hr, which equals Gasfgapho.
Gasfg2_fuel	Flue gas at point 2 in ductwork in lbmole/lb-fuel, which equals Gas1_fuel.
Gasfg2	Flue gas at point 2 in ductwork in lbmole/hr, which equals Gas1.
Gasfg3_fuel	Flue gas at point 3 in ductwork in lbmole/lb-fuel, which equals Gas2_fuel.
Gasfg3	Flue gas at point 3 in ductwork in lbmole/hr, which equals Gas2.
Gasstack_fuel	Flue gas entering stack in lbmole/lb-fuel, which equals Gas3_fuel.
Gasstack	Flue gas entering stack in lbmole/hr, which equals Gas3.

4.1.11 Tempfluegas Model

This model keeps track of the flue gas temperature in degrees Rankine at different points in the power plant. As other modules are loaded they may redefine these variables, but they remain in the Tempfluegas model. There is a temperature variable for every flue gas variable in Fluegas. This model contains Temp_fdfan, Temp_apho, Temp_furn, Temp_econo, Temp_fgaphi, Temp_fgapho, Temp_gasleak, Temp_fg1, Temp_fg2, Temp_fg3, and Temp_stack.

Temp_fdfan	Combustion air temperature entering the forced draft and primary air fans in degrees Rankine, which equals $T_{ai_air} + 460$.
Temp_apho	Combustion air temperature exiting the air preheater in degrees Rankine, which equals the $Orgtemp_apho + 460$.
Temp_furn	Flue gas temperature in furnace in degrees Rankine, which equals $Orgtemp_furn + 460$.
Temp_econo	Flue gas temperature exiting the economizer in degrees Rankine, which equals $Orgtemp_econo + 460$.
Temp_fgaphi	Flue gas temperature entering the air preheater in degrees Rankine, which equals Temp_econo.
Temp_fgapho	Flue gas temperature exiting the air preheater in degrees Rankine, which equals $Orgtemp_fgapho + 460$.
Temp_leak	Average temperature of the air leaking across the air preheater, which equals the sum of Temp_fdfan + Temp_fgapho divided by 2.
Temp_fg1	Flue gas temperature at point 1 in the ductwork in degrees Rankine, which equals Tempfgapho.
Temp_fg2	Flue gas temperature at point 2 in the ductwork in degrees Rankine, which equals Tempfg1.
Temp_fg3	Flue gas temperature at point 3 in the ductwork in degrees Rankine, which equals Tempfg2.
Temp_stack	Flue gas temperature entering the stack in degrees Rankine, which equals Tempfg3.

4.1.12 Solidsfluegas Model

This model calculates the mass flow rates of the bottom ash and flyash, and tracks the entrained solids in the fluegas as it passes through the ductwork in the power plant. All the variables in the model are in units of tons/hr and indexed by Solidslabel. This allows the IECM to distinguish between the ash from the coal, excess reagents, sulfur byproducts, moisture, and miscellaneous compounds in the solids streams. This model contains Solidsfurn, Bottomash, Solidsecono, Solidsfgaphi, Solidsfgapho, Solidsfg1, Solidsfg2, Solidsfg3, and Solidsstack.

Solidsfurn	Total solids in the furnace in tons/hr.
BottomAsh	Bottom ash (and solids) exiting the bottom of the furnace in tons/hr, which equals Solidsfurn times one minus the overhead ash fraction.
Solidsecono	Entrained solids flow rate exiting the economizer in tons/hr which equals Solidsfurn times the overhead ash fraction.
Solidsfgaphi	Entrained solids flow rate entering the air preheater in tons/hr, which equals

	Solidsecono.
Solidsfgapho	Entrained solids flow rate entering the air preheater in tons/hr, which equals Solidsfgapho.
Solidsfg1	Entrained solids flow rate at point 1 in the ductwork, which equals Solidsfgapho.
Solidsfg2	Entrained solids flow rate at point 2 in the ductwork, which equals Solidsfg1.
Solidsfg3	Entrained solids flow rate at point 3 in the ductwork, which equals Solidsfg2.
Solidsstack	Entrained solids flow rate entering the stack, which equals Solidsfg3.

4.1.13 Fgsummary Model

This model creates four summary tables of the flue gas as it passes through the power plant. These tables show the flue gas composition, total molar flow rate, volumetric flow rate, the temperature, entrained solids composition, SO_2 , and NO_x . These tables allow the user to see how the flue gas and its components change as it passes through the power plant. The variables Fg_point1, Fg_point2, and Fg_point3 are character strings which are used in the labels of the summary tables in this model. In this model, their definitions are 'Flue Gas at 1', 'Flue Gas at 2', and 'Flue Gas at 3,' but they are changed as other modules are loaded into Demos. These new definitions are character strings which describe the location of the flue gas relative to the equipment which have been loaded into Demos. For example, when the wet FGD module is loaded into Demos it changes the definition of Fg_point2 and Fg_point3 to 'Exiting Scrubber' and 'Exiting Reheater,' respectively. This model contains Fg_point1, Fg_point2, Fg_point3, Fluegas_index, Fluegascomp, Gasarray, Temparray, Solidsarray, and Fluegaschar.

Fg_point1	Variable which describes the flue gas at point 1 relative to the other abatement technologies, and used as a label in the summary tables in this model.
Fg_point2	Variable which describes the flue gas at point 2 relative to the other abatement technologies, and used as a label in the summary tables in this model.
Fg_point3	Variable which describes the flue gas at point 3 relative to the other abatement technologies, and used as a label in the summary tables in this model.
Fluegas_index	Row labels for the summary tables in this model, which use Fg_point1, Fg_point2, and Fg_point3.
Fluegascomp	Column labels for the summary table Fluegaschar, which must follow Fluegas_index so that Fluegaschar prints out properly.
Gasarray	Summary table of the flue gas components in lbmole/hr.
Temparray	Summary table of the flue gas temperature in degrees Fahrenheit.
Solidsarray	Summary table of the entrained solids in the flue gas in tons/hr.
Fluegaschar	Summary table which contains the total flue gas flow rate in moles/hr, flue gas volumetric flow rate in acfm, temperature in degrees Fahrenheit, total mass flow rate of entrained solids in tons/hr, SO_2 flow rate in lbs/MBtu, and NO_x flow rate in lbs/MBtu in columns 1 through 6, respectively.

4.1.14 Emissions Model

This model calculates the uncontrolled power plant emissions and removal efficiencies required to meet the standards for SO_2 , NO_x , and particulates. The default sulfur dioxide and TSP standards are the NSPS values, while the NO_x limit is assumed to be approximately an order of magnitude more stringent than the current standard so that NO_x controls are required. It contains Potemso2, Erso2, Ernox, Erash, Fgdremeff, Noxremeff, Tspremeff, Emis_index, Emis_param, and Emischar.

Potemso2	The potential emissions of SO_2 of the raw coal in pounds per million Btu. This does not take into account any coal cleaning or sulfur retention in the ash..
ErSO2	The emissions out of the boiler including the effects of coal cleaning and sulfur retention in the ash. (PE_s , 6)
Ernox	The equivalent NO_x emissions out of the boiler in pounds per million Btu. (PE_{NO_x} , 6)
Erash	The ash emissions out of the boiler in pounds per million Btu. (EC_a , 6)
Fgdremeff	The sulfur dioxide removal efficiency required by any flue gas desulfurization device. (η_{SO_2})
NOxremeff	The NO_x removal efficiency required. (η_{NO_x} , 6)
Tspremeff	The TSP removal efficiency required. (η_{TSP} , 6)
Emis_index	Row label for EmisChar.
Emis_Param	Column label for Emischar, which must follow Emis_index for Emischar to print out properly.
EmisChar	Summary table of emission constraints and removal efficiencies.

4.2 Plant.petc File

This file contains all the economic variables for the base power plant, and modifies some of the performance variables to account for other models developed for PETC. This file opens Coalinputs, Baseplinputs, Unc_plant, Fuel_char, and Plantperf. It also creates a new input model, Reagents, which has the cost for various reagents used by the different abatement technologies. Finally it creates the uncontrolled power plant economic model, PlantEcon. The hierarchy of the IECM created by "Plant.petc" after "Initiecm.mod" and "Initiecm.petc" have been loaded into Demos is shown in Figure 4-3.

4.2.1 Coalinputs Model

This file opens the Coalinputs model to define the four variables needed to calculate the delivered cost of coal, Raw_coal_cst, Ctrans, Clean_cost, and MinerPBcost.

Raw_coal_cst	Raw coal cost on a moist basis. The default value was estimated using the EPRI TAG [4] 1985 value of 1.28 \$/MBtu for the mine-mouth cost of a high sulfur midwest coal. This converts to 26.10 \$/ton for the default coal, assuming no escalation. The valid range is 5 to 75 \$/ton.
Ctrans	Transportation cost on a wet basis. The default value was estimated using the EPRI TAG [4] 1985 value of 0.29 \$/MBtu for the transportation cost of a

Figure 4-3: Hierarchy of IECM created by Plant.petc after Initiecm.mod and Initiecm.petc has been Loaded into Demos

	midwest coal delivered to the ENC2 region. This converts to 5.90 88\$/ton for the default coal, assuming no escalation. The valid range is 0 to 30\$/ton.
Clean_cost	Coal cleaning cost on a wet basis. The default is zero but it is redefined by the coal cleaning modules if they are used. The valid range is 0 to 40 \$/ton.
MinerPBcost	Miners' Pension and benefit fund cost. As estimated in "Economic Evaluation of FGD Systems," [10] (page 13-13) this cost is \$2/ton for bituminous coal on a wet basis. The valid range is 0 to 5 \$/ton.

4.2.2 Basepltinputs Model

This model contains the economic input parameters for the base power plant and redefines Plantinputs to display these new variables. It contains Icfbp, Oplaborrate, and Lc.

Icfbp	Indirect charge factor for the base power plant capital cost. The default is 0.5, with a valid range of 0.25 to 1.0.
Oplaborrate	Operating labor cost for the entire power plant. This was estimated using the EPRI TAG [4] 1985 value of \$19.70/man-hour and scaling it to 1988 with the Chemical Engineering Cost index. The default is \$20.74/man-hour, with a valid range of 10 to 30.
Lc	Land cost in \$ per acre. This was estimated using the EPRI TAG [4] 1985 value of \$6,500/acre and scaling it to 1988 with the Chemical Engineering Cost index. The default is \$6,840/acre, with a valid range of 2,500 to 10,000.
PltInputLabel	The column labels for Plantinputs is redefined to include Icfbp and Oplaborrate.
Plantinputs	The power plant input table is redefined to include Icfbp and Oplaborrate.

4.2.2.1 Unc_plant Model

An uncertainty variable for the land cost, Lc, is added to Unc_plant. It contains Uflc.

Uflc	Uncertainty variable for the land cost. The default is a normal distribution with a mean of 1 and a standard deviation of 0.15.
------	---

4.2.3 Reagents Model

This model contains all the reagent costs used in the IECM. This ensures that the same costs for reagents are used for different abatement technologies. All reagent costs are in 1988 dollars. It contains Lime_cost, Limestone_cost, Ammonia_cost, Oxygen_cost, Methane_cost, Oil6_cost, Sulfacid_cost, Sulfur_cost, Co2_cost, Reagentindex, and Reagentinputs.

Lime_cost	Lime cost, which was estimated from the EPRI TAG [4] 1985 value. The default is \$65/ton, with a valid range of 40 to 90. This has not been escalated since new data indicates that the cost has remained relatively constant.
Limestone_cost	Limestone cost, which was estimated from the EPRI TAG [4] 1985 value. This has not been escalated since new data indicates that the cost has remained relatively constant. The default is \$15/ton, with a valid range of 7 to 23.

Ammonia_cost	Ammonia cost, which was estimated from the EPRI TAG [4] 1985 value. This has not been escalated since new data indicates that the cost has remained relatively constant. The default is \$150/ton, with a valid range of 100 to 25.
Oxygen_Cost	Liquid oxygen cost, which was estimated from Economic Evaluation of FGD Systems, Vol 1 [10]. This value has been scaled to 1988 dollars with the Chemical Engineering cost index. The default is \$115/ton, with a valid range of 85 to 145.
Methane_cost	Methane cost, which was estimated from the EPRI TAG [4] 1985 value. This has not been escalated since new data indicates that the cost has remained relatively constant. The default value is \$4.5/mscf, with a valid range of 3.5 to 5.5.
Oil6_cost	Number 6 fuel oil cost, which was estimated from the EPRI TAG [4] 1985 value. This has not been escalated since new data indicates that the cost has remained relatively constant. The default value is \$4.5/MBtu, with a valid range of 3.5 to 5.5.
Sulfacid_cost	Sulfuric acid cost. The default value was estimated using the PETC Flue Gas Cleanup Technologies By-Products report's [11] 1987 value of \$55 per metric ton. This converts to 53 88\$/ton using the Chemical Engineering Index. The valid range is 25 to 75.
Sulfur_cost	Sulfur cost. The default value was estimated using the PETC Flue Gas Cleanup Technologies By-Products report's [11] 1987 value of \$130 per metric ton. This converts to 125 88\$/ton using the Chemical Engineering Index. The valid range is 100 to 200.
Co2_cost	Carbon dioxide cost, which was estimated from EPRI Economic Evaluation of FGD Systems [10] and scaled to 1988 with the Chemical Engineering Cost Index. The default is \$115/ton, with a valid range of 50 to 80.
Reagentindex	Label index for Reagentinputs.
Reagentinputs	Summary table of all the input variables defined in this model. Since all the variables are index by Cost_yr, the reform command is used to reformat the table.

4.2.3.1 Unc_reagents

This model contains the uncertainty distributions for the reagents' cost defined in Reagents. Unlike other uncertainty models, this model does not have an uncertainty control variable, since these uncertainty distribution are defined and controlled by the abatement technologies when they are loaded into Demos. The definitions for all these variables is one, since other abatement technologies will define these variables. This model contains Uflime_cost, Uflimestone_cost, Ufammonia_cost, Ufoxygen, Ufmethane_cost, Ufoil6_cost, Ufsulfacid_cost, Ufsulfur_cost, and UfCo2_cost.

4.2.4 Fuel_char Model

This model calculates the delivered cost of coal and creates and a table summarizing the result. It contains Cd, Coallabels, Cd_param, and Cdchar.

Cd	The cost of as-fired coal at the power plant. It is the sum of Raw_coal_cst, Ctrans, Clean_cost, and MinerPBcost. Since cleaning coal changes the moisture content of the coal, this model adjusts the raw coal cost for the moisture content of the delivered coal, so that the cost per ton on a dry basis is constant.
Coallabels	Row labels for Cdchar.
Cd_param	Column label for Cdchar, which must follow Coallabels so that Cdchar prints out properly.
Cdchar	Summary table which displays the components of the delivered cost of coal.

4.2.5 Plantperf Model

This model redefines the net capacity, Mwnet, and the summary table, Plantchar, to account for the technologies developed for PETC. It also defines two new variables, Mwgnew and Mwhgrossannual, to account for the energy credit from the Copper Oxide and NOXSO processes. How the IECM handles energy credits and penalties is defined in the the next Subsection 4.2.6. This model contains Energycredit, Mwnet, Mcoalcredit, Mwgnew, Mwhgrossannual, Plant_index, and Plantchar.

Energycredit	Additional electricity in MW generated from the heat added to the boiler or the steam cycle from the abatement technologies. The default value is zero, but it is redefined if the copper oxide or NOXSO modules are loaded into Demos. This variable accounts for the efficiency of the boiler, the steam cycle heat rate, and the internal electricity consumption of the uncontrolled power plant.
Mwnet	Net electrical capacity, MW, which is redefined to include process energy penalties and Energycredit. Notice that all the energy penalties are based on the gross electrical capacity, Mwg, excluding any the energy credit.
Mcoalcredit	Equivalent tons per hour of coal which would be required to produce the capacity specified in Energycredit.
Mwgnew	Electrical capacity, MW, including the capacity specified by Energycredit, but not including the energy penalties of the abatement technologies.
Mwhgrossannual	Electricity produced in a year in MWh including the capacity specified by Energycredit, but not including the energy penalties of the abatement technologies.
Plant_index	Redefinition of the row labels for Plantchar, to include the coal credit, energy credit, new gross capacity, and electricity produced.
PlantChar	Redefinition of the summary to include the variables defined in this model.

4.2.6 Plantecon Model

This model determines the capital and operating cost of the uncontrolled power plant to determine the cost of electricity, Electricity_cost, charged to the abatement technologies. The IECM determines the cost of electricity by dividing the total revenue requirement of the uncontrolled plant by the megawatt-hours produced in a year. The total revenue requirement is the sum of annualized capital, operating and maintenance costs including the value of any energy

credit. The megawatt hours produced is equal to the gross capacity minus the internal energy consumption, steam pumps, pulverizers, FD fans, primary fans, and miscellaneous, plus any energy credit. The IECM handles energy credits by assuming that the uncontrolled plant builds and operates a boiler, steam cycle, turbine, and generator that is larger by the increment, EnergyCredit. Therefore, it incurs additional capital costs, Delttddcbp, and operating cost, Delttoandmbp, for this larger system. The process which created the energy credit gets "credit," Coalcredit, for "selling" the energy to the plant.

The IECM handles energy penalties in a similar fashion except that the abatement technologies get charged for the electricity and steam they use, while the uncontrolled power plant gets a credit for the sale of electricity. The net result of the accounting is that the cost of electricity charged to the abatement technologies does not change with different configurations. It depends only on the plant size, steam cycle heat rate, internal energy consumption, coal characteristics, coal cost, levelized fixed charge factor, and levelization factor. A conceptual model for this arrangement is that the uncontrolled power plant pays for the capital and operating cost of producing electricity. It pays for the coal consumed and any energy it receives from the abatement technologies, and gets paid by the abatement technologies for the electricity it provides to them. The abatement technologies are responsible for building and operating the equipment needed to clean the flue gas. PlantEcon contains Coalcredit, Delttoandmbp, Toandmbp, Tdcbporig, Delttddcbp, Tdcbp, Tccbp, Tccbpout, Trrbpwoutil, Electricity_cost, Utilbp, Trrbp, Elecbrp, Bp_econ_ind, and Planteconchar.

Coalcredit	Monetary equivalent in M\$/yr for the energy produced by an abatement technology. This is the cost of the equivalent amount of coal needed to produce the energy credit. It is charged to the uncontrolled power plant and credited to the abatement technology.
Delttoandmbp	Non-fuel operating and maintenance cost in M\$/yr of the energy produced by the abatement technology. This is charged to the uncontrolled power plant.
Toandmbp	Operating and maintenance cost in M\$/yr of power plant. It includes the cost of coal usage, maintenance, and the operating and maintenance cost of the energy credit.
Tdcbporig	Direct capital cost for uncontrolled power plant in M\$, which is scaled to the net capacity of the uncontrolled power plant, Mwwopce. This does not include the capital cost for the energy credit.
Delttddcbp	Direct capital cost for energy credit in M\$ which equals the average normalized capital cost times the energy credit. The average normalized capital cost is just the direct capital cost determined by Tdcbp, divided by the net uncontrolled power plant size, Mwwopce.
Tdcbp	Total direct capital cost for uncontrolled power plant in M\$, which is the sum of Tdcbporig and Delttddcbp.
Tccbp	Total capital cost for the power plant in M\$, including the indirect capital cost.
Tccbpout	Normalized total capital cost for the power plant in \$/kW.
Trrbpwoutil	Total revenue requirement in M\$/yr for the base plant not including the income from providing electricity to the abatement technologies.

Electricity_cost	Cost of electricity in mills/kWh charged to the abatement technologies. This is how much it costs the uncontrolled power plant to produce electricity. The user has the option of changing this value to an input parameter.
Utilbp	Income in M\$/yr to the uncontrolled power plant from the electricity sold to the abatement technologies. This equals the negative of the sum of all the utility costs for the abatement technologies. Its value is negative since income can be viewed as a negative cost.
Trrbp	Total revenue requirement for the base plant in M\$/yr, including income from the electricity sold to abatement technologies, Utilbp.
Elecbp	Total revenue requirement for the base plant in mills/kWh. This equals the total revenue requirement for the base plant, Trrbp, divided by the net electricity sold to the public, Mwhannual. The difference between this variable and Electricity_cost is that this variable explicitly accounts for the electricity used by the abatement technologies. If Electricity_cost is not changed to an input parameter, then this value should equal Electricity_cost, regardless of the amount of power consumed or any energy produced by the abatement technologies.
Bp_econ_index	Row labels for Planteconchar.
Bpecon_param	Column label for Planteconchar, which must follow Bp_econ_index in order for Planteconchar prints out correctly.
Planteconchar	Base plant summary economic table which displays all the values for the variables in this model.

5. Economics Module

This chapter describes the economics module, which has three main functions. One is to calculate the fixed charge factor and the variable cost levelization factor. The second is to calculate the overall plant costs, Final cost, including the abatement equipment. The third is to perform a year by year analysis. The fixed charge factor and the variable cost levelization factor can either be specified, or they can be calculated using the algorithm given in the EPRI Technical Assessment Guide [4, 5].

5.1 Economic Overview

Before describing the variables in the economics module, an overview of the economic methodology used in the IECM is given. The revenue requirement method is used to determine the electricity cost. This method calculates the annual revenues needed to cover all capital and operating costs for a particular technology. The method is described in detail in Reference [4].

In order to consider the effects of coal cleaning, the overall cost of power generation needs to be estimated as well as environmental control costs. The net cost of electricity provided by the controlled power plant is calculated by dividing the total revenue requirement for the entire plant by the net kilowatt-hours produced in one year. The total revenue requirement is the total capital cost times the fixed charge factor plus the total variable cost times the levelization factor. This electricity cost is actually the marginal cost of electricity for the entire plant.

All abatement technologies have the following variables with the following format. The format is a generic variable name plus the name of the module. For example, TDCwet refers to the direct capital costs for a wet FGD system. Another important point is that total costs for the entire plant equal the sum of all the abatement technology costs plus the base plant cost.

TDC	This is the total direct cost in millions of dollars as a function of performance parameters. The total direct cost includes equipment and other costs associated directly with equipment. It is adjusted to the current year with a cost index. The base plant direct capital cost is adjusted by the Handy-Whitman index. All the other abatement technologies are adjusted with the Chemical Engineering index.
TCC	This is the total capital cost in millions of dollars. This includes the total direct capital costs plus the costs for interest during construction, contingencies, initial catalyst and chemical charge, etc. For most modules the total capital costs calculates the indirect capital costs and adds them to the total direct costs. For these modules, the indirect charge factor is calculated. In the base plant module, the total capital cost is calculated by multiplying the total direct cost by an indirect charge factor, which is an input parameter.
ICF	The indirect charge factor for the total capital cost, expressed as a fraction of the total direct cost. Only the base plant module has the indirect charge factor as a single input parameter. All other modules determine the components of the indirect charges to calculate the total effective indirect charge factor.
TCCout	The total capital cost expressed in dollars per kilowatt. This is obtained by dividing the total capital cost by the net capacity of the power plant.

TOandM	This is the total variable cost (except for utility costs) in millions of dollars per year. It is adjusted to the current year by the Chemical Engineering Index.
TRRwoUtil	This is the total revenue requirement minus the costs for utilities in million of dollars per year. It is the sum of the total variable cost times the levelization factor plus the total capital cost times the fixed charge factor.
Util	This is the total utility cost in millions of dollars per year based on the electricity cost for the base plant. For all abatement technologies it is equal to the energy usage times the electricity cost. For the entire plant, this is zero, while for the base plant, it equals the negative of the sum of all abatement utility costs.
TRR	This is the total revenue cost in million of dollars per year including utility costs. For the entire plant, TRR and TRRwoUtil are equal since Util is zero. For the base plant, TRRbp is less than TRRbpwoUtil since Utilbp is negative.
Elec	This is the total revenue cost in mills per kilowatt hour based on the net kilowatt hours of the power plant.

5.2 Economics.petc File

This file contains all the models and variables in the economics module. It opens Econfunctions to add several functions for performing year-by-year analysis, and two input parameters models, Econmodelinputs and Unc_econ. Finally, it creates Economics, Fixcharge, Finalcostmodel, and Finalryrmodel. The hierarchy created by "Economics.petc," after "Initiecm.mod" and "Initiecm.petc" have been loaded into Demos, is shown in Figure 5-1.

5.2.1 Econfunctions Model

This model, which is located within the Iecmfunctions model, defines several functions used for the year-by-year revenue analysis. These functions can be used for the entire plant or individual abatement technologies. Econfunctions contains Summarycost, Tccyear, Oandmyear, Trryear, and Elecyear.

Summarycost(sumcostvar:numeric)

Creates a table of Baseplantcost, Noxcosts, Tspcosts, Fgdcosts, Swdcosts, Pollutcosts, Totalcosts, and Pollutcostcc when the variable Coals is a vector. It requires one of the variables listed above as an input parameter.

Tccyear(capitalcost:numeric)

Creates a table of the capital charges in M\$/yr over the booklife based on the year-by-year analysis method from the EPRI TAG [4].

Oandmyear(oandm:numeric)

Creates a table of the operating and maintenance cost in M\$/yr over the booklife based on the year-by-year analysis method from the EPRI TAG [4].

Trryear(capitalcost,oandm:numeric)

Creates a table of the total revenue requirement cost in M\$/yr over the booklife based on the year-by-year analysis method from the EPRI TAG [4].

Elecyear(capitalcost,oandm:numeric)

Figure 5-1: Hierarchy Created by Economics.petc after Initiecm.mod and Initiecm.petc have been Loaded into Demos

Creates a table of the total revenue requirement cost in mills/kWh over the booklife based on the year-by-year analysis method from the EPRI TAG [4].

5.2.2 Econmodelinputs Model

Inflat_ctl	Inflation control variable, which controls whether to perform the economic analysis in constant or current dollars. It can be either 'real constant' or 'real current.'
Inflation	Inflation rate over the expected life expressed as a fraction. The default is 6%; however, it equals zero if Inflat_cost equals 'real constant.' The valid range is 0 to 0.20.
Real_fuel_Escal	Real escalation rate for variable costs over the expected life expressed as a fraction. The default is 0.0, with a valid range of 0 to 0.10.
Real_Bnd_Cost	Rate of return for bonds in constant dollars expressed as a fraction. The default is 0.046, with a valid range of 0 to 0.15.
Real_Pref_Stock	Rate of return for preferred stock in constant dollars expressed as a fraction. The default is 0.052, with a valid range of 0 to 0.20.
Real_Com_Stock	Rate of return for common stock in constant dollars expressed as a fraction. The default is 0.071, with a valid range of 0 to 0.25.
Fed_tax	Federal tax rate expressed as a fraction. The default is 0.367, with a valid range of 0.15 to 0.50.
State_tax	State tax rate expressed as a fraction. The default is 0.02, with a valid range of 0 to 0.10.
Advalorem	Property tax rate expressed as a fraction. The default is 0.02, with a valid range of 0 to 0.05
Itc	Investment tax credit expressed as a fraction. The default is 0, with a valid range of 0 to 0.20.
Pref_fraction	Fraction of capital cost financed by preferred stock. The default is 0.15, with a valid range of 0 to 1.
Debt_fraction	Fraction of the capital cost financed by bonds. The default is 0.50, with a valid range of 0 to 1.
Booklife	Expected life of the plant for tax purposes in years. The default is 30, with a valid range of 5 to 60.
Econlabels	Row labels for Econinputs.
Econassu_param	Column labels for Econinputs, which must follow Econlabels in order for Econinputs to print out correctly.
Econinputs	Table of the economic input parameters shown above.
Time	Lifetime vector from one to booklife. The default is sequence(1,booklife).
Depreciation	Depreciation schedule expressed as a vector which has a length of Booklife. The default assumption is a twenty year accelerated depreciation schedule over 15 years based on EPRI TAG [5]. The default is array(time,[.075 .069 .064 .059 .055 .051 .047 .045 .045 .045 .045 .045 .045 .045 .044 .044 .044 0 0 0 0 0 0 0 0 0]). This can be changed to a straight line

depreciation over the booklife by defining it to be $1 / \text{Booklife}$.

5.2.2.1 Unc_econ Model

This model contains the dimensional uncertainty distributions for the Economics model. It contains Ctl_econ, Econunctest, and Ufvclcf.

Ctl_econ	Uncertainty control variable for uncertain parameters in the Economic model. The default is 'off,' but it can also be 'on.'
Econunctest	Uncertainty test variable which determines whether the uncertain variables in this model are probabilistic or deterministic.
Uffcf	Dimensional uncertainty variable for the fixed charged factor. The default is a uniform distribution from 70% to 130% of the deterministic value.
Ufvclcf	Dimensional uncertainty variable for the variable cost levelization factor. The default is a normal distribution with a mean of 1 and a standard deviation of 0.03.

5.2.3 Economics Model

This model contains the models to calculate the fixed charged factor, levelization factor, Final_cost table, and the Final_yr table. The Final_cost table breakdown the capital, operating, and revenue requirement for the entire plant by abatement technologies. The Final_yr table displays a year-by-year total revenue analysis for the entire plant. This model contains Fixcharge, Finalcostmodel, and Finalryrmodel.

5.2.4 Fixcharge Model

This section describes the implementation of the algorithms in the EPRI Technical Assessment Guide [4] which calculate the fixed charge factor and variable levelization factor. This model allows the revenue requirement costs to be displayed in either current or constant dollars. Constant dollars are obtained by setting the Inflat_cost to 'real constant.'

Bond_Cost	Rate of return for bonds expressed as a fraction in current dollars.
Pref_stock	Rate of return for preferred stocks expressed as a fraction in current dollars.
Com_stock	Rate of return for common stocks expressed as a fraction in current dollars.
Equity_Cost	Weighted rate of return for the preferred and common stock expressed as a fraction.
Discount	Discount rate, which is the weighted rate of return for debt and equity expressed a fraction
Tax	Composite tax rate of the federal and state taxes expressed as a fraction.
Book_depr	Book depreciation rate net of the investment tax credit expressed as a fraction.
Deferred_tax	Deferred income tax over the booklife, expressed as a fraction over time.
Initial_Balance	This is the remaining capital that has to be depreciated in the first year expressed as a fraction.
Remain_bal	Depreciated book value in each year net the deferred tax. It is expressed as a

	fraction.
Return_Debt	Return on debt over the booklife expressed as a fraction.
Return_Equity	Return on common and preferred stock over the booklife expressed as a fraction.
Tax_paid	Taxes paid in each year expressed as a fraction.
Yr_Yr_Cc	year-by-year carrying charge expressed as a fraction.
PV_factor	Present value factor over the booklife expressed as a fraction.
Annuity	Annuity factor over the booklife expressed as a fraction.
Cum_pres_cc	Cumulative present value of the carrying charges, expressed as a fraction.
Yr_Fcf	Levelized carrying charge in each year.
Fcf	Fixed charge factor for annualizing capital costs. The default value is the levelized carrying charge at the end of the book life, but it can also be changed to a simple input parameter.
Fuel_Escal	Apparent annual variable costs escalation rate.
Escal_Factor	Escalation factor used to calculate the levelization factor.
Vclf	Variable cost levelization factor. This variable can be changed into a simple input parameter.
Econ_index	Row labels for Econchar.
Econ_param	Column labels for Econchar, which must follow Econ_parameter in order for Econchar to print out properly.
Econchar	Summary table for the rate of return for bonds, preferred stock, common stock, equity, total equity and debt, the overall tax rate, book depreciation, fixed charge factor, escalation rate, and levelization factor.
Year_labels1	Column labels for Deprectaxtable.
Deprectaxtable	Summary table for the depreciation schedule, deferred tax, remaining balance, return of debt, return on equity, and taxes paid.
Year_labels2	Column labels for Carryingcharge.
Carryingcharge	Summary table for the annuity factor, yearly carrying charge, cumulative carrying charge, and the levelized carrying charge.

5.2.5 Finalcostmodel Model

The main purpose of this model is to create the Final_cost table, which summarizes the cost for all abatement technologies and the overall plant.

Process	The row index for the final cost summary table, Final_Cost.
Costs	The column index for the final cost summary table, Final_Cost, which must follow Process in order for Final_cost to print out correctly.
Unitlabels	Unit labels for the columns in Final_cost.
Baseplantcost	Summary array of the capital cost, operating cost, and revenue requirement for the uncontrolled power plant.

Noxcost	Summary array of the capitial cost, operating cost, and revenue requirement for the NO _x abatement technology included in the IECM.
Tspcost	Summary array of the capitial cost, operating cost, and revenue requirement for the particulate abatement technology loaded in this configuration.
Fgdcost	Summary array of the capitial cost, operating cost, and revenue requirement for the SO ₂ or combined SO ₂ /NO _x abatement technology loaded in this configuration.
Swdcosts	Summary array of the capitial cost, operating cost, and revenue requirement for the solid waste disposal system for the FGD technologies.
Pollutcosts	Summary array of the capitial cost, operating cost, and revenue requirement for all abatement technologies.
Total_cost	Summary array of the capitial cost, operating cost, and revenue requirement for the entire power plant.
Final_Cost	Main summary table for the IECM, which displays the capital, operating, and revenue requirement for the uncontrolled power plant, abatement technologies, and entire power plant.
Pollutcostcc	Summary array of the capitial cost, operating cost, and revenue requirement for all abatement technologies including coal cleaning. For this variable to work correctly Coals must be an array with 1 as one of the elements. For example, Coals can equal [1 2], or [1 5], but not [4 5].
Elecpc	Total revenue requirement for all abatement technologies not including coal cleaning in mills/kWh.
Elecpccc	Total revenue requirement for all abatement technologies including coal cleaning in mills/kWh. Coals must be an array with 1 as an element in order for it to work correctly. See the description of Pollutcostcc for examples.
Tccoutpc	Total capital cost for all abatement technologies not including coal cleaning in \$/net kW.
Tccoutpccc	Total capital cost for all abatement technologies including coal cleaning in \$/net kW. Coals must be an array with 1 as an element in order for it to work correctly. See the description of Pollutcostcc for examples.

5.2.6 Finalyryrmodel Model

The only purpose of this model is to create Final_yryr, which produces a summary table of the capitial cost, operating cost, and revenue requirement for the entire plant over its book life. This table uses the year-by-year analysis functions defined at the beginning of this file.

Yryrlabel	Column labels for Final_yryr
Yryrtable	Summary table of the total capitial cost, operating cost, and revenue requirement for the entire plant over its book life. The rows are years starting from the completion date, while the columns contain the costs.

6. References

1. Henrion, M. and N. Wishbow. *Demos User's Manual: Version Three*. Carnegie Mellon University, Department of Engineering and Public Policy, Pittsburgh, PA, August 1987.
2. Rubin, E.S., J.S. Salmento, J.G. Barrett, C.N. Bloyd, H.C. Frey. *Modeling and Assessment of Advanced Processes for Integrated Environmental Control of Coal-Fired Power Plants*. Final Report of Contract No. DE-FG22-83PC60271 from Carnegie Mellon University, Pittsburgh, PA to U.S. Department of Energy, Pittsburgh, PA, July 1986, NTIS No. DE86014713/WEP, National Technical Information Service, Springfield, VA.
3. Frey, H.C.. *Performance and Economic Model of the Fluidized Bed Copper Oxide Process*. M.E. Thesis, Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA, May 1987.
4. TAG - *Technical Assessment Guide: Electricity Supply - 1986, Vol. 1*. P-4463-SR, Electric Power Research Institute, Palo Alto, CA, December 1986.
5. TAG - *Technical Assessment Guide: Fundamentals and Methods, Supply-1986, Vol. 3*. P-4463-SR, Electric Power Research Institute, Palo Alto, CA, May 1987.
6. Barin, I. and O. Knacke. "Thermochemical Properties of Inorganic Substances". Springer-Verlag, New York, NY, 1973.
7. Barin, I., O. Knacke and O. Kubaschewski. "Thermochemical Properties of Inorganic Substances: Supplement, Vol. 1". Springer-Verlag, New York, NY, 1977.
8. Babcock, G.H., and S. Wilcox. "Steam/Its Generation and Use". Babcock and Wilcox Company, New York, NY, 1978.
9. Combustion Engineering Inc. "Combustion: Fossil Power Systems". J.G. Singer, Ed., Windsor, CT 06095, 1981.
10. *Economic Evaluation of FGD Systems*. EPRI CS-3342, Vols. 1-3, Prepared by Stearns-Roger Engineering Corporation for Electric Power Research Institute, Palo Alto, CA, December 1983.
11. Burns and Roe Services Corporation. *PETC Flue Gas Cleanup Technologies By-Products*. Prepared by Burns and Roe Services Corporation for the U.S. Department of Energy, Pittsburgh Energy Technology Center, September (1987).
12. Luborsky, F. E. "High Gradient Magnetic Separation for Removal of Sulfur from Coal". Final Report, FE-8969-1 (EPA-600/7-78-208), DoE/EPA, November 1978.

**Appendix B: Evaluation of the Fluidized Bed Copper Oxide Process
Using a Probabilistic Engineering Model**

EVALUATION OF THE FLUIDIZED BED COPPER OXIDE PROCESS USING A PROBABILISTIC ENGINEERING MODEL

H. C. Frey, E. S. Rubin, and J. S. Salmento
Carnegie-Mellon University
Pittsburgh, PA 15213

Presented at:
The Sixth Annual International Pittsburgh Coal Conference
September 25-29, 1989

ABSTRACT

Performance and economic evaluations of innovative clean coal technologies are typically expressed as deterministic ("best guess") estimates that mask the uncertainties associated with processes at an early stage of development. In this paper applications of a new probabilistic engineering model developed for DOE's Pittsburgh Energy Technology Center (PETC) are described. The example presented analyzes the PETC fluidized bed copper oxide process for simultaneous SO_2 and NO_x reduction from coal-fired power plant flue gas. The copper oxide process is in an early stage of development; however, flowsheets for a commercial scale unit have been developed. The probabilistic modeling capability permits evaluation of the effects of simultaneous variation in uncertain performance and cost parameters for a commercial-scale design. The probabilistic engineering model is applied to: (1) identify the uncertain performance and cost parameters that contribute most to the uncertainty in overall cost; (2) estimate process contingency costs; (3) identify potential cost pay-offs of process improvements; and, (4) compare the advanced process with conventional technology to evaluate potential markets for advanced processes. Results of the analysis indicate specific areas for process design improvements; suggest appropriate contingency factors for the copper oxide process based on probabilistic analysis; and indicate the sensitivity of potential cost savings to design conditions and the availability of byproduct markets.

INTRODUCTION

The purpose of research is to provide and improve information regarding the feasibility, promising applications, optimal designs, uncertainties and risks associated with a new process. The information generated from research can be used by process developers to refine a technology and ultimately by potential process adopters to make a decision about whether, and under what circumstances, to use a new technology. Of concern to a process developer is the prioritization of research needs for a new technology. Which technologies are the most promising for further process development? What conditions favor the selection of the new technology? What specific technical areas require more research? What economic and cost uncertainties affect the economic feasibility of the technology?

This paper presents an evaluation method that can be used to help answer these questions for clean coal technology processes. The salient features of the method include:

- Development of engineering performance and cost models;
- A probabilistic modeling capability to incorporate uncertainties;
- Judgments regarding uncertainties; and
- Exercising of the models to answer these questions:
 - What uncertainties most affect the overall costs?
 - What are the key process design trade-offs?
 - What are the potential pay-offs and risks vis-a-vis conventional technology?

ENVIRONMENTAL REGULATIONS

Current federal new source performance standards (NSPS) applicable to coal-fired power plants require up to 90 percent sulfur dioxide (SO₂) removal, over 99 percent particulate matter (PM) removal, and moderate (about 50 percent) reduction of nitrogen oxides (NO_x) emissions. A conventional emission control system for a new coal-fired power plant typically consists of a wet limestone flue gas desulfurization (FGD) system for SO₂ control, an electrostatic precipitator (ESP) for PM removal, and combustion controls for NO_x reduction. The spent limestone reagent used in the FGD system is disposed of with the power plant solid waste. These systems are all commercially available and well-demonstrated. However, recent commercial experience in Japan and West Germany with selective catalytic reduction (SCR) indicates that 80 to 90 percent NO_x removal may be feasible, although SCR has not yet been applied with U.S. coals (Cichanowicz and Offen, 1987; Damon and Giovanni, 1987).

Alternatives to conventional pulverized coal (PC) combustion, such as integrated coal gasification combined cycle (IGCC) systems, are capable of NO_x emissions below those of PC plants equipped with SCR, as well as high (over 95 percent) levels of SO₂ control (Cool Water, 1986). Furthermore, political concern over acid rain (for which SO₂ and NO_x are precursors) may accelerate the time table for more stringent emission regulations of conventional PC plants. Therefore, there is incentive to develop technology options to reliably achieve stringent emission reductions at minimum cost in a timely fashion.

INTEGRATED ENVIRONMENTAL CONTROL

With the prospect of increasingly stringent emission control has evolved the concept of integrated environmental control. The concept has several dimensions. One is to consider interactions among control methods for air, water, and solid waste emissions control. Another is the integrated use of pre-combustion, combustion, and post-combustion control methods (as distinct from one approach alone). A third dimension is the development of new processes for combined pollutant removal in lieu of separate processes for individual pollutants. Thus, integrated environmental control represents good design practice and provides opportunities to minimize costs for a given set of emission reduction requirements (Carr, 1986).

The DOE Pittsburgh Energy Technology Center (PETC) has conducted research on a number of innovative technologies that combine SO₂ and NO_x removal into a single reactor, and that reduce the solid waste produced by air pollution control systems. One of these technologies, which is used as a case study in this research, is the fluidized bed copper oxide process. Testing of the copper oxide process began at PETC in 1975 and has progressed through several stages in three different test units (Demski *et al*, 1982; Yeh *et al*, 1984; Plantz *et al*, 1986; Williamson *et al*, 1987). Key features of the copper oxide process are that, unlike a wet FGD/SCR system, (1) it combines SO₂ and NO_x removal in a single reactor vessel, and (2) it is regenerative (i.e. the reagent is reused rather than disposed of) and produces a saleable sulfur or sulfuric acid byproduct. The solid waste from a copper oxide system consists only of the fly ash collected in a conventional fabric filter. Conceptual designs of commercial scale copper oxide systems have been developed (e.g., SMC, 1983a). A simple schematic of a power plant with the copper oxide process is shown in Figure 1.

In the copper oxide process, a copper impregnated sorbent, consisting of small diameter aluminum spheres, circulates between a fluidized bed reactor, where SO₂ in the flue gas is removed by reaction with copper oxide in the sorbent, and a regenerator, in which SO₂ is evolved in a reaction

of the sulfated sorbent with methane. The SO_2 -rich gas from the regenerator is sent to an elemental sulfur or a sulfuric acid plant for byproduct recovery. The regenerated sorbent is returned to the absorber for another cycle. NO_x is removed from the flue gas by reaction with ammonia, which is injected into the flue gas upstream of the absorber. The absorber reactions are exothermic, increasing the temperature of the flue gas. This energy can be recovered in the power plant furnace through additional preheating of the furnace combustion air by the power plant air preheater.

A PROBABILISTIC ENGINEERING MODEL

The copper oxide process is a technology in an early phase of development, for which limited test data and no commercial design or operating experience are available. Uncertainties in system performance at the commercial scale lead to uncertainties in the required size of process equipment and the consumption of materials (e.g., sorbent) and parasitic power. These uncertainties result in uncertainties in capital and operating costs, which are the ultimate measures of interest for comparative analysis. Furthermore, even if process performance were known with certainty, uncertainties regarding the costs of equipment (particularly equipment not previously used in commercial scale service) and reagents remain. To explicitly characterize these uncertainties, and to evaluate the overall uncertainty in process costs, a probabilistic engineering modeling framework has been developed.

Analytic models for a conventional PC power plant, pre-combustion coal cleaning processes, and the components of conventional and advanced post-combustion pollution control systems are available as part of the Integrated Environmental Control Model (IECM), developed under contract to PETC by Rubin et al. (1986). Details of the copper oxide process model, plus models of the power plant air preheater and a sulfuric acid recovery plant, are described elsewhere (Frey, 1987). A number of more recent refinements have been made to the copper oxide process model. These include replacing a regression equation used to estimate the molar ratio of copper in the sorbent to sulfur in the flue gas with a kinetics-based reaction model developed at PETC (Yeh and Drummond, 1986); development of a sulfur recovery plant model based on an engineering study done for PETC (SMC, 1983a); and refinements to some of the input parameters.

The analytic models are based on mass and energy balances for key process equipment. The cost models are based on equipment cost estimates available in the literature, adjusted for plant size using key process stream flow rates and exponential scaling factors. Indirect costs are calculated based on the direct equipment costs. Variable and fixed operating costs are also calculated. The IECM has a capability to report costs in either constant or current dollars. Constant 1985 dollars, which are exclusive of inflation, are used in this analysis.

A unique feature of the IECM is that it is implemented in Demos, a non-procedural interactive modeling environment developed by Henrion (Henrion, 1982; Henrion and Wishbow, 1987) for performing probabilistic analysis. The key uncertainties in process parameters can therefore be characterized using a variety of distribution functions available in Demos. The resulting uncertainty distributions for model outputs are calculated using median Latin hypercube sampling, a variant of Monte Carlo simulation.

SPECIFICATION OF MODEL PARAMETERS AND UNCERTAINTIES

An integrated emission control system consisting of the copper oxide process with integrated coal cleaning, byproduct recovery, and energy recovery via the power plant air preheater, will be compared with a conventional system consisting of wet FGD and SCR. Table 1 summarizes some of the key parameters, including emission constraints, base plant design, and financial parameters,

assumed for this analysis. Both deterministic values and probability distributions are indicated in the table. Table 2 summarizes the different coals considered, including both unwashed and cleaned (30 percent sulfur reduction on an energy basis) coals. Table 3 summarizes key input values and distributions for the conventional wet FGD/SCR emission control system, which is taken as a technological baseline in this analysis. The main emphasis of this research is on applying the probabilistic evaluation method to the copper oxide process, assuming the FGD/SCR system as a benchmark. The key inputs and distributions assigned to the copper oxide emission control system are summarized in Table 4. The probabilities assigned to key model inputs reflect judgments on the part of model developers as well as judgments solicited from process developers (Rubin et al., 1986; Frey, 1987; Rubin, Salmento, and Frey, 1988).

IDENTIFICATION OF KEY UNCERTAINTIES

The primary advantage of probabilistic simulation over traditional sensitivity analysis is the simultaneous incorporation of uncertainties in multiple model inputs. The resulting interactions among uncertain variables results in uncertainties in total costs, which are the basis for comparative analysis. The uncertainty in total costs provides information about the technical and economic risk of the technology. Research can provide additional information about uncertain input variables. This may result in changes in parameter values or their uncertainty (such as the standard deviation), and thus affect the overall uncertainty of the technology. This can help focus research on key parameters that reduce the uncertainties that contribute most to the risk of technology failure.

The key parameter uncertainties have been identified by estimating correlations between the primary cost results, such as total revenue requirement or capital cost, and the copper oxide process input uncertainties included in Table 4. Correlations provide a measure of the linear dependence of one distribution on another; however, there are some non-linear relationships in the model, such as between sorbent flow rate and regeneration efficiency. Scatter plots can be used to visually identify non-linear dependencies that may not be well-characterized by correlation coefficients.

The uncertainties which were found to contribute most to the levelized process cost uncertainty included sorbent attrition, regeneration efficiency, and the standard error of the copper-to-sulfur molar ratio, with correlations of 0.55, -0.41, and 0.41, respectively. Scatter plots did not reveal any strong non-linear dependencies. Uncertainties regarding sorbent cost and plant capacity factor were also significant. Uncertainty in sorbent attrition is the largest factor affecting uncertainty in total variable costs, while uncertainty in regeneration and the standard error of the copper-to-sulfur mole ratio significantly affect both capital and operating cost uncertainties. Further research on the copper oxide process should explicitly focus on improving understanding of sorbent attrition, regeneration efficiency, and the variability in the copper-to-sulfur ratio required to achieve a given SO₂ removal efficiency.

ESTIMATING CONTINGENCY COSTS

Nearly all capital cost estimates, whether for a new or existing technology or for a preliminary or detailed cost estimate, include a contingency factor. The contingency is often the single largest expense in the cost estimate, and yet it is also the least documented. A contingency is used to represent additional costs that are expected to occur, but that are not included explicitly in the cost estimate (Milanese, 1987). Contingency factors are typically simple multipliers that are applied to installed equipment costs toward the end of an analysis (e.g., after process area costs have been estimated without regard to their uncertainty). A probabilistic modeling approach supplants the traditional contingency factor approach by incorporating expert knowledge about uncertainties explicitly and at a more disaggregated level (e.g., for specific performance and cost parameters). Furthermore, while simple contingency factors provide no explicit insights into the specific

performance or cost parameters that contribute most to the process technical and economic risks, a probabilistic approach permits identification and ranking of the uncertain parameters that contribute most to the overall uncertainty, as discussed above.

The Electric Power Research Institute (EPRI) uses two types of contingency factors: project and process contingency (EPRI, 1986). The "project" contingency is intended to cover the costs of additional equipment or other costs that would result from a more detailed design of a definitive project at a specific site. This implies that as costing proceeds from a preliminary to a detailed final estimate, the project contingency factor should be reduced. In the present analysis, a project contingency of 25 percent is assumed for the copper oxide process, based on an estimate by Science Management Corporation (1983c). The "process" contingency is intended to quantify the additional costs expected due to uncertainty in the technical performance and commercial scale cost of a new technology. This contingency factor is reduced as a technology proceeds from bench scale to full commercial use. Both of the project and process contingency factors are deterministic estimates of additional costs that are expected to occur. However, even in the EPRI Technical Assessment Guide (1986), there is little substantive discussion of how these factors should be derived.

The Rand Corporation conducted a survey of 18 companies in the chemical and petroleum industries to determine the actual methods used to develop contingency factors (Milanese, 1987). The study indicates that contingency factors are often badly under-estimated, which may be leading to bad decisions about certain projects. Rand recommends the greater and more formalized use of experience, the use of a "delphi" technique to get multiple expert inputs, and the inclusion of costs associated with risks and innovation. A probabilistic approach to cost estimating provides a systematic, quantitative method to explicitly incorporate detailed expert judgments about uncertainties. Because the uncertainties contributing to "contingencies" are considered at a disaggregated level, more realistic estimates of performance and cost will generally result.

A deterministic capital cost estimate can include information developed in a probabilistic estimate through appropriate selection of the contingency factor. The contingency factor can be defined as the value that adjusts the deterministic estimate (without contingency) to some specified fractile of the probabilistic estimate. Typically, some "best estimate" value from the probabilistic analysis, such as the mean or the median, would be used. However, if there is significant risk aversion on the part of an investor, who may want to minimize the chance of a cost over-run, then an upper fractile from the probability distribution (e.g., 90th percentile) may be used.

An example of this type of analysis is shown graphically in Figure 2, which shows the results of a probabilistic analysis of the capital cost of the copper oxide process. The mean value of the distribution is found to be \$111 million, which is slightly above the median value (50 percent probability) due to the skewness assumed for some of the input parameter probability distributions. For a completely deterministic case using nominal values with no contingencies the cost is found to be \$74 million. Thus, the mean value of the probabilistic analysis corresponds to a deterministic overall contingency factor of 80 percent or a 55 percent process contingency, assuming a 25 percent project contingency, as illustrated in Figure 3. The probability of an overrun at this contingency factor is seen to be 45 percent.

The contingency factor estimated in this fashion is significantly higher than the 55 percent value (30 percent process contingency, 25 percent project contingency) assumed in previous analyses (SMC, 1983c). Contributing factors to the difference are the uncertainties assigned to the regeneration efficiency and the capital costs for each major process section, which are skewed. The difference is not surprising, since the previous contingency was based on a rule-of-thumb, rather than a detailed probabilistic risk assessment. The fact that the original estimate seems to be low is also supported by the results of the Rand study, which indicates that contingency factors are generally grossly under-estimated, especially early in process development.

DESIGN TRADE-OFFS AND MARKET NICHES

Analyses regarding trade-offs and potential emission control technology market niches must consider performance and cost interactions between the control technology and the balance of the power plant system, in addition to interactions within the technology itself. Key process trade-offs are identified using probabilistic estimates of the mean costs associated with various design decisions. Comparative analysis of the copper oxide process and an FGD/SCR system are used to identify potential market niches for the new technology.

Design Trade-Offs

A number of controllable design parameters are evaluated to determine their effect on process performance and cost. These design parameters include the fluidized bed height, air preheater size, weight percent of copper in the sorbent, sulfur recovery option, and coal characteristics (including coal cleaning).

The value of the copper oxide process model in identifying process trade-offs is best illustrated by comparing a common assumption with modeling results for different fluidized bed heights. The common assumption is that increasing the fluidized bed height, which reduces the required sorbent flow rate and increases the pressure drop in the bed (increasing the flue gas fan operating costs), results in a net process cost increase. However, the model results indicate otherwise. Because much of the equipment in the copper oxide process, including the sorbent transport system, the solids heater, and the regenerator, are sized based on the sorbent flow rate, a reduction in sorbent flow rate can yield significant capital cost savings. Furthermore, operating costs are reduced, due primarily to lower sorbent replacement costs and lower methane requirements to heat and regenerate the sorbent. Model results indicate a net levelized cost savings as the fluidized bed depth is increased.¹ For example, the mean levelized savings for an increase in bed height from 36 to 48 inches is 1.1 mills/kWh with a washed Illinois No. 6 coal. Therefore, an integrated process model of a technology in the research phase can yield important insights into performance and cost trade-offs.

Optimization of the sorbent copper content is an important area for further research. The primary trade-off is between reducing the sorbent mass flow rate and sorbent attrition. PETC tests have indicated that at a very high sorbent copper loading of 18 percent, the sorbent essentially disintegrates (Yeh and Drummond, 1986). An engineering study for PETC reports that copper loadings from 5 to 11 percent may have comparable attrition characteristics (SMC, 1983b). However, most testing has been with sorbents containing between 5 and 7 percent copper (e.g., Yeh *et al.*, 1984; Plantz *et al.*, 1986; Williamson *et al.*, 1987). Figure 4 shows the effect of increasing the sorbent copper loading from 5 to 10 percent for three coals.² The results indicate a significant advantage for the 7 percent copper sorbent compared to the 5 percent copper sorbent. The 7 percent copper sorbent is used as the basis for later case studies. Additional savings may be realized by a further increase to 10 percent weight copper. These results indicate that additional research on sorbent attrition at the higher copper loadings is merited.

¹ Although the model does not account for additional structural support costs required for the absorber to handle the increased sorbent weight and pressure drop, it is expected that an incremental increase in bed height from the commonly assumed 36 inch bed depth to a 48 inch bed depth can still yield a significant cost savings.

² The cost of sorbent is assumed not to vary for moderate increases in sorbent loading. Impregnation of the alumina pellets with copper is a relatively small cost of sorbent production compared to manufacturing the alumina spheres. Sorbent attrition is also assumed not to vary significantly for sorbent copper loadings in the 5 to 10 percent range.

One process integration issue is the recovery of energy added to the flue gas by the exothermic reactions in the fluidized bed absorber. The sizing of the power plant air preheater, which heats the inlet air to the plant furnace, affects the magnitude of the additional energy that can be recovered in the furnace, thereby displacing some coal consumption and resulting in an energy credit. A deterministic "best guess" analysis indicates that there is an overall cost penalty to enlarging the air preheater. However, a probabilistic analysis indicates that an enlarged air preheater does provide an overall cost savings. The difference in results is due to the skewness of many of the distributions assigned to key parameters in the probabilistic model, indicating that there is a value of including the additional information about uncertainty in the analysis. Furthermore, the model accounts for downstream effects, such as the size of the fabric filter particulate collector, which are often neglected by process developers. The cost of the fabric filter is reduced by the larger air preheater, because the fabric filter inlet flue gas temperature, and the corresponding volumetric flue gas flow rate, are reduced. The mean value of the total levelized pollution control system cost saving is 0.32 mills/kWh with a cleaned Illinois No. 6 coal. These potential cost savings, which are revealed by probabilistic modeling and an integrated consideration of downstream effects, are measures of both the value of including uncertainty in the model and the value of an integrated model that accounts for downstream effects.

Another process integration issue is the sulfur byproduct recovery option, which has a significant effect on process costs. Elemental sulfur recovery is more expensive than sulfuric acid recovery, adding to the levelized pollution control costs a mean value of 2 mills/kWh for the cleaned Illinois #6 coal and 0.44 mills/kWh for the unwashed Pittsburgh coal. The sulfur plant consumes a significant amount of methane as a reducing reagent, which is the primary contributor to increasing the system operating costs. Partly because the removal efficiency of the copper oxide process must be increased to compensate for the lower recovery efficiency of the sulfur plant,³ and partly because the capital cost of an integrated sulfur plant is somewhat higher than a sulfuric acid plant for the cases considered, the total capital cost of a copper oxide system with sulfur recovery is higher than with sulfuric acid recovery. However, market conditions are generally expected to favor the sale of sulfur over sulfuric acid as a byproduct (Burns and Roe Services Corp., 1987). Therefore, even though the system costs are higher, the potentially larger market penetration of elemental sulfur may lead to a larger overall pay-off from R&D.

Identification of Potential Market Niches

The ultimate pay-off from R&D on the copper oxide process is significantly influenced by expected cost savings compared to conventional systems. Key factors which influence the cost savings include the sulfur recovery option and byproduct market, plant size, capacity factor, and coal characteristics. To determine potential market niches, comparisons between copper oxide and conventional FGD/SCR systems are made on the basis of total pollution control system costs, which are exclusive of the base plant and include SO₂, NO_x, and PM removal, solid waste handling, and coal cleaning. Any emission control system-related changes to the base plant are charged to the pollution control system. As a result, interactions among components of the pollution control system and between the pollution control system and the base plant are integrated into the analysis.

³ Two other approaches to compensate for the comparatively low sulfur recovery efficiency of the elemental sulfur plant (only 95 percent compared to 99.5 percent for a sulfuric acid plant) are treatment of the tail gas leaving the sulfur plant, which is a common feature of integrated gasification combined cycle (IGCC) designs (e.g., Fluor, 1984) or recycle of the sulfur plant tail gas to the power plant flue gas (SMC, 1983a). These options may yield significant cost savings.

Although the elemental sulfur recovery option is shown to be more expensive than the sulfuric acid option, both systems enjoy levelized cost savings over conventional FGD/SCR systems for a wide range of capacity factors (0.50 to 0.80) and plant sizes (300 MW to 700 MW). For the base case 500 MW plant with a 65 percent capacity factor, the cost savings are 8 percent and 20 percent for the elemental sulfur and sulfuric acid recovery options, respectively, compared to the FGD/SCR system.

The effect of coal cleaning on process costs is an important consideration in comparative analysis. Because many of the costs of the copper oxide process are sensitive to sorbent flow rate, which in turn is proportional to the coal sulfur content, a reduction in the coal sulfur content through coal cleaning can reduce the overall cost of the process in some cases. In contrast, the FGD/SCR system is comprised of separate reactor vessels for SO₂ and NO_x control, both of which are proportional in cost primarily to the flue gas flow rate and not significantly influenced by coal cleaning. Although coal cleaning results in higher coal costs, for the copper oxide process with high sulfur coal this is more than offset by decreased process costs. For medium and low sulfur coals, however, the increased costs of coal cleaning are larger than the reduction in process costs.

As an example, results for the Illinois No. 6 coal are shown in Figure 5 for the copper oxide process with sulfuric acid recovery. The contribution of each emission control process to the total costs is shown in the figure. Integration of 30 percent coal sulfur reduction through coal cleaning with the copper oxide process is shown to reduce levelized costs by approximately one mill/kWh; the cost saving is similar for the sulfur recovery option. For the lower sulfur Pittsburgh coal, coal cleaning increases costs for all the pollution control systems. However, the cost advantage of the copper oxide process is increased for the lower sulfur coal, primarily due to a reduction in sorbent handling requirements. In the remaining comparisons of FGD/SCR and copper oxide systems, optimal levels of coal cleaning are used; i.e., no coal cleaning for the FGD/SCR systems or for the copper oxide systems with the Pittsburgh coal, and 30 percent sulfur reduction through washing for copper oxide systems with Illinois No. 6 coal.

PROBABILISTIC COMPARATIVE ANALYSIS: FOUR CASE STUDIES

Based on the identification of copper oxide process design issues, four case studies involving probabilistic comparisons of copper oxide and conventional environmental control systems were made. These comparisons were made for optimal levels of coal cleaning and for both sulfuric acid and sulfur recovery options. All of these comparisons are based on a 48 inch bed height and 7 percent copper sorbent. Probabilistic comparison of the pollution control systems provides information about the likely cost savings that can be achieved by the new process, as well as the risks that the new technology may be more expensive than existing technology.

Because many of the input parameter distributions are common to both systems (e.g., financial parameters, base plant characteristics, solid waste disposal, and ammonia cost), there is, in general, a positive correlation between the cost distributions for the two systems.⁴ Therefore, the distributions have been determined for the cost *differences* between the copper oxide and FGD/SCR systems in which samples for the distributions of costs of each system were paired.

Figure 6 shows differences in levelized pollution control costs between FGD/SCR and copper oxide systems for two coals and the two sulfur recovery options. In all cases, the copper oxide process is most likely to be less expensive than the FGD/SCR system; however, for the higher

⁴ In this example, the correlation between the uncertainty distributions of levelized pollution control costs for conventional FGD and advanced copper oxide/sulfuric acid systems is estimated to be 11 percent for optimal levels of coal cleaning with the Illinois No. 6 coal.

sulfur coal there is a substantial risk that the copper oxide process will be more expensive. Taking the case with sulfur recovery and the Illinois No. 6 washed coal as an example, there is nearly a 30 percent probability that the new process will be more expensive than conventional technology, based on the levelized costs. For the medium sulfur Pittsburgh coal, the probability of the new technology being more expensive than the conventional system is negligible. Furthermore, the magnitude of cost savings is likely to be larger for the Pittsburgh coal than for the higher sulfur Illinois No. 6 coal. In all cases, there is considerable uncertainty in the amount of the cost savings. The 90 percent probability range for the Illinois No. 6 coal with sulfur recovery is -5 mills/kWh to 8 mills/kWh. There is a small probability that the cost savings could be significantly higher.

CONCLUSIONS

The foregoing analyses have demonstrated several points about research-phase modeling of a new technology. A probabilistic capability has permitted an evaluation of the effect of simultaneous uncertainties in multiple performance and cost parameters and the identification of key uncertainties contributing most to the uncertainty in process cost. An integrated model of the copper oxide process has permitted the evaluation of interactions involving components of the copper oxide process, the pollution control system, and the power plant. These interactions, which can be overlooked if not included in a systematic modeling framework, significantly influence process costs. Identification of important interactions provides the basis for determining research priorities, such as evaluating the effects of increased bed height on sorbent circulation rate. The explicit characterization of uncertainty in the model provides additional insights that may be overlooked in deterministic analysis, as demonstrated with the air preheater sizing analysis. Integration of pre- and post-combustion pollution control measures can lead to significant cost savings with high sulfur coals, although results indicate that the copper oxide process has an increased comparative advantage over FGD/SCR systems on medium sulfur coals.

While the magnitude of cost savings may be greatest on medium sulfur coals, the copper oxide system appears to dominate FGD/SCR systems for all cases considered. Cost savings appear to be larger for sulfuric acid byproduct recovery systems, although the available markets may be more limited than for elemental sulfur. The availability of byproduct markets will be significant in determining the extent of process application and the pay-off from R&D. Probabilistic comparisons of innovative and conventional technologies provides quantitative information about the risk that a new technology may be more expensive, and the potential pay-off of process research. By explicitly considering uncertainties and key process interactions, the probabilistic engineering models can be used to improve research planning and ultimately to assist potential process adopters in decision making.

REFERENCES

- Burns and Roe Services Corp. (1987). PETC Flue Gas Cleanup Technologies By-Products. Prepared by Burns and Roe Services Corporation for the U.S. Department of Energy, Pittsburgh Energy Technology Center. September 1987.
- Carr, R.C. (1986). Integrated Environmental Control in the Electric Utility Industry. Presented at the APCA/ASME/EPRI Third Symposium on Integrated Environmental Controls, Pittsburgh, PA. February 1986.
- Cichanowicz, J.E., and G.P. Offen (1987). Applicability of European SCR Experience to U.S. Utility Operation. Presented at the EPA/EPRI 1987 Joint Symposium on Stationary Combustion NOx Control, New Orleans, LA. March 23-26, 1987.
- Cool Water (1986). Cool Water Coal Gasification Program: Fourth Annual Progress Report. Prepared by Cool Water Coal Gasification Program, Bechtel Power Corporation, and Radian Corporation for the Electric Power Research Institute, Palo Alto, CA. EPRI AP-4832. October 1986.
- Damon, J.F., and D.V. Giovanni (1987). Updated Technical and Economic Review of Selective Catalytic Reduction NOx Systems. Presented at the EPA/EPRI 1987 Joint Symposium on Stationary Combustion on NOx Control, New Orleans, LA. March 23-26, 1987.
- Demski, R.J., S.J. Gasior, E.R. Bauer, Jr., J.T. Yeh, J.P. Strakey, and J.I. Joubert (1982). Simultaneous Removal of SO₂ and NO_x From Flue Gas Employing a Fluidized Bed Copper Oxide Process. Presented at 1982 Summer Session of AIChE Meeting. August 29 - September 1, 1982.
- EPRI (1986). TAG(tm) - Technical Assessment Guide, Volume 1: Electricity Supply - 1986. EPRI P-4463-SR. Electric Power Research Institute, Inc. December 1986.
- Fluor (1984). Cost and Performance for Commercial Applications of Texaco-Based Gasification-Combined-Cycle Plants, Volume 1: Summary and Discussion of Results. Prepared by Fluor Engineers, Inc. for the Electric Power Research Institute, Palo Alto, CA. AP-3486. April 1984.
- Frey, H.C. (1987). Performance and Economic Model of the Fluidized Bed Copper Oxide Process. Master's Thesis. Department of Mechanical Engineering, Carnegie-Mellon University. Pittsburgh, PA. May 1987.
- Henrion, M. (1982). The Value of Knowing How Little You Know: The Advantages of a Probabilistic Treatment of Uncertainty in Policy Analysis. Ph.D. Thesis, School of Urban and Public Affairs, Carnegie-Mellon University, Pittsburgh, PA. March 1982.
- Henrion, M., and N. Wishbow (1987). Demos User's Manual: Version 3. Department of Engineering and Public Policy, Carnegie-Mellon University, Pittsburgh, Pa.
- Milanese, J.J. (1987). Process Industry Contingency Estimation: A Study of the Ability to Account for Unforeseen Costs. N-2386-PSSP. Rand Corporation. June 1987.
- Plantz, A.R., C.J. Drummond, S.W. Hedges, and F.N. Gromicko (1986). Performance of the Fluidized Bed Copper Oxide Process in an Integrated Test Facility. Presented at the 79th

Annual Meeting of the Air Pollution Control Association, Minneapolis, Minnesota. June 22-27, 1986.

Rubin, E.S., J.S. Salmento, J.G. Barrett, C.N. Bloyd, and H.C. Frey. (1986). Modeling and Assessment of Advanced Processes for Integrated Environmental Control of Coal-Fired Power Plants. Prepared by the Center for Energy and Environmental Studies, Carnegie-Mellon University, for the U.S. Department of Energy, Pittsburgh Energy Technology Center. July 1986.

Rubin, E.S., J.S. Salmento, and H.C. Frey (1988). Evaluating Combined SO₂/NO_x Processes. Presented at the Fourth Integrated Environmental Control Symposium, Sponsored by the Electric Power Research Institute, Washington, DC, March 2-4, 1988.

Shah, P.P, and D.G. Leshock (1988). Sorbent Life Cycle Testing: Fluidized Bed Copper Oxide Process. Presented at the Fourth Integrated Environmental Control Symposium, Sponsored by the Electric Power Research Institute, Washington, DC, March 2-4, 1988.

SMC (1983a). Technical and Economic Evaluation of the Fluidized Bed Copper Oxide Flue Gas Treatment Process and Integrated Sulfur-Producing Plants. Prepared by Science Management Corporation for the U.S. Department of Energy, Pittsburgh Energy Technology Center, Pittsburgh, PA. June 1983.

SMC (1983b). Technical and Economic Evaluation of Design Optimization Issues Associated With the Fluidized Bed Copper Oxide Flue Gas Treatment Process. Prepared by Science Management Corporation for the U.S. Department of Energy, Pittsburgh Energy Technology Center, Pittsburgh, PA. November 1983.

SMC (1983c). Economic Evaluation of the Fluidized Bed Copper Oxide Flue Gas Treatment Process and Integrated Sulfuric Acid Plant. Prepared by Science Management Corporation for the U.S. Department of Energy, Pittsburgh Energy Technology Center, Pittsburgh, PA. April 1983.

Williamson, R.R., J.A. Morici, and T.L. LaCosse (1987). Sorbent Life Cycle Testing, Fluidized Bed Copper Oxide Process. Prepared by UOP, Inc., for the U.S. Department of Energy, Pittsburgh Energy Technology Center.

Yeh, J.T., R.J. Demski, J.P. Strakey, and J.I. Joubert (1984). PETC Fluidized Bed Copper Oxide Process for Combined SO₂/NO_x Removal From Flue Gas. Presented at the AIChE Winter National Meeting. Atlanta, Georgia. March 11-14, 1984.

Yeh, J.T., and C.J. Drummond (1986). Process Simulation of the Fluidized Bed Copper Oxide Process Sulfation Reaction. Presented at the AIChE 1986 Spring National Meeting, New Orleans, LA, April 6-10, 1986.

Table 1: Selected Input Parameter Assumptions for Case Studies

Model Parameter	Nominal Value	Probability Distribution	Values (or σ as % of mean)
Emission Constraints			
Nitrogen Oxides	90% Reduction		
Sulfur Oxides	90% Reduction		
Particulates	0.03 lb/MBtu		
Power Plant Parameters			
Gross Capacity	522 MW		
Gross Heat Rate	9500 Btu/kWh	-1/2 Normal	(1.8 %)
Capacity Factor	65 %	Normal	(7 %)
Excess Air (boiler/total)	20 %/39 %	Normal	(2.5 %)
Ash to Flue Gas	80 %		
Sulfur to Flue Gas	97.5 %		
Economizer Outlet Temp	700 °F		
Preheater Outlet Temp	300 °F		
Financial Parameters			
Inflation Rate	0 %		
Debt Fraction	50 %		
Common Stock Fraction	35 %		
Preferred Stock Fraction	15 %		
Real Return on Debt	4.6 %	Normal	(10 %)
Real Return on Com. Stock	8.7 %	Normal	(10 %)
Real Return on Pref. Stock	5.2 %	Normal	(10 %)
Federal Tax Rate	36.7 %		
State Tax Rate	2.0 %		
Ad Valorem Rate	2.0 %		
Investment Tax Credit	0 %		
Book Life	30 years		
Real Fuel Escalation	0 %	1/2 Normal	$\sigma = 0.06$ %

Table 2. Selected Properties of Coals Used for Case Studies (As-Fired Basis)

Coal Property	Illinois No. 6 Coal		Pittsburgh Coal	
	Run-of-Mine	Washed ^a	Run-of-Mine	Washed ^a
Heating Value, Btu/lb	10,190	10,330	13,400	12,900
Sulfur, wt %	4.36	3.09	2.15	1.66
Carbon, wt %	57.0	57.7	74.8	72.1
Hydrogen, wt %	3.7	4.0	4.6	4.5
Oxygen, wt %	7.2	8.4	5.3	5.4
Nitrogen, wt %	1.1	1.1	1.4	1.3
Moisture, wt %	12.3	17.5	2.7	7.9
\$/ton (at mine)	26.10	30.68	33.40	34.99
\$/ton (transport)	7.90	7.90	7.90	7.90

^a Model results for a 30 % sulfur reduction on a lb/MBtu basis using conventional coal cleaning (Level 3 plant design)

Table 3. Nominal Parameter Values and Uncertainties for the Conventional Environmental Control System

Model Parameter	Nominal Value	Probability Distribution	Values (or σ as % of mean) ^a
<u>Wet FGD System</u>			
Molar Stoichiometry	(calc)	Normal	(5 %)
No. Operating Trains	4	Chance	10 % @ 1; 20 % @ 2; 40 % @ 3; 30 % @ 4
No. Spare Trains	1	Chance	75 % @ 0; 25 % @ 1
Reheat Energy	(calc)	Chance	75 % @ 0; 25 % @ x
Total Energy Use	(calc)	Normal	(10 %)
Limestone Cost	\$15/ton	Uniform	\$10-15/ton
Direct Capital Costs	(calc)	Normal	(10 %)
Operating Costs	(calc)	Normal	(10 %)
<u>Selective Catalytic Reduction</u>			
Space Velocity	2,850/hr	Normal	(10 %)
NH ₃ Stoichiometry	(calc)	Normal	(10 %)
Catalyst Life	15,000 hrs	Chance	5 % @ 1,275 hrs 30 % @ 5,700 hrs 50 % @ 11,400 hrs 14 % @ 17,100 hrs 1 % @ 28,500 hrs
Energy Requirement	(calc)	Normal	(10 %)
Ammonia Cost	\$150/ton	Uniform	\$150-225/ton
Catalyst Cost	\$460/ft ³	Normal	(7.5 %)
Direct Capital Cost	(calc)	Triangular	0.8x, x, 2x
Operating Cost (excl. Cat.)	(calc)	Normal	(10 %)
<u>Cold-Side Electrostatic Precipitator</u>			
Specific Collection Area	(calc)	Normal	(5 %)
Energy Requirement	(calc)	Normal	(10 %)
Total Capital Cost	(calc)	Normal	(10 %)
Operating Cost	(calc)	Normal	(10 %)
<u>Solid Waste Disposal</u>			
Land Cost	\$6,500/acre	Normal	(10 %)
Direct Cost	(calc)	Normal	(10 %)
Operating Cost	(calc)	Normal	(10 %)

^a For uniform distributions actual values are shown. For triangular distributions, endpoints and median are shown. For chance distributions, the probabilities of obtaining specific values are shown.

Table 4. Nominal Parameter Values and Uncertainties for the Advanced Environmental Control System

Model Parameter	Nominal Value	Probability Distribution	Values (or σ as % of mean) ^a
Copper Oxide Process^b			
Fluidized Bed Height	48 inches		
Sorbent Copper Loading	7 wt-%		
Regeneration Efficiency	99.2 %	-1/2 Normal	(20 %)
Fluidized Sorbent Density	400 kg/m ³	Normal	(10 %)
Standard Error, Cu/S Ratio	0	Normal	$\sigma = 0.39$
Sorbent Attrition	0.06 %	Normal	(41 %)
Ammonia Stoichiometry	(calc)	Normal	(6.25 %)
Regeneration Temp	900 °F	Normal	(2 %)
No. Operating Trains	4	Chance	10 % @ 1; 20 % @ 2; 40 % @ 3; 30 % @ 4
No. Spare Trains	1	Chance	50 % @ 0; 50 % @ 1
Sorbent Cost	\$5.00/lb	-1/2 Normal	(25 %)
Methane Cost	\$4.50/mscf	1/2 Normal	(25 %)
Ammonia Cost	\$150/ton	Uniform	\$150-225/ton
Sulfuric Acid Cost	\$40/ton	-1/2 Normal	(30 %)
Sulfur Cost	\$125/ton	-1/2 Normal	(30 %)
Absorber Direct Cap. Cost	(calc)	Uniform	1.0x - 1.5x
Solids Heater DCC	(calc)	Uniform	1.0x - 1.5x
Regenerator DCC	(calc)	Uniform	1.0x - 1.5x
Solids Transport DCC	(calc)	Uniform	1.0x - 2.0x
Sulfur Recovery DCC	(calc)	Uniform	1.0x - 1.2x
Total Capital Cost	(calc)	1/2 Normal	(10 %)
Fabric Filter			
Air-to-Cloth Ratio	2.0 acfm/ft ²	-1/2 Normal	(10 %)
Bag Life	(calc)	Normal	(25 %)
Energy Requirement	(calc)	Normal	(10 %)
Bag Cost	\$0.80/ft ²	Normal	(5 %)
Operating Cost	(calc)	Normal	(15 %)
Total Capital Cost	(calc)	Normal	(15 %)
Solid Waste Disposal			
Land Cost	\$6,500/acre	Normal	(10 %)
Direct Cost	(calc)	Normal	(10 %)
Operating Cost	(calc)	Normal	(10 %)

^a For uniform distributions actual values are shown. For triangular distributions, endpoints and median are shown. For chance distributions, the probabilities of obtaining specific values are shown.

^b As part of integration of the copper oxide process with the base power plant, the plant air preheater is resized to maintain an exit flue gas temperature of 300 °F.

Figure 1. Power Plant Design with a Copper Oxide Emission Control System

Figure 3. Overall Contingency Factor and Total Capital Cost: Deterministic Analysis of the Copper Oxide Process

Figure 4. Mean Levelized Pollution Control Costs for Copper Oxide System with Acid Recovery: Effect of Sorbent and Coal Characteristics

Figure 5. Mean Levelized Pollution Control Cost versus Sulfur Reduction from Coal Cleaning: Copper Oxide/Sulfuric Acid Plant with Illinois #6 Coal

Figure 6. Comparison of Levelized Pollution Control Cost Savings for Copper Oxide vs. FGD/SCR Systems: Effect of Coal Cleaning and Byproduct Recovery Option

Appendix C: Prototype Screens for A User-Friendly Interface

Select one of the following options:

Tutorial

Documentation

Run the Model

Integrated Environmental Control Model

I. Configure Plant

II. Set Parameters

III. Run Model

1. Coal Cleaning ☐

2. Combustion ☐

3. Flue Gas Cleanup ☐

4. Waste Disposal & By-Product Recovery ☐

5. Plant Diagram ☐

Exit

Default

Click 1 - 4 to configure plant

Integrated Environmental Control Model

I. Configure Plant

II. Set Parameters

III. Run Model

1. ☐ Coal Properties	11. ☐ Overall Economics
2. ☐ Coal Cleaning	12. ☐ Uncertainty Control
3. ☐ Plant Performance	
4. ☐ Ambient Conditions & Emissions	
5. ☐ Plant Economics	
6. ☐ Reagent Properties	
7. ☐ NOx Cleanup : Selective Catalytic Reduction	
8. ☐ Flue Gas Cleanup : Copper Oxide	
9. ☐ Particulate Removal : Fabric Filter	
10. ☐ Solid Waste Disposal	

Exit

Default

Click 1 - 12 to set input parameters

I. Configure Plant

II. Set Parameters

III. Run Model

Summary

Exit

Click on Diagram

Integrated Environmental Control Model

I. Configure Plant

II. Set Parameters

III. Run Model

Exit

Click on diagram or select from menu

Firing Method :

- ☒ Tangential
- ☐ Wall
- ☐ Cyclone

Combustion Controls :

- ☒ Low NO_x Burners
- ☐ Sorbent Injection
- ☐ Reburning
- ☐ No Controls

Previous

CheckList

Next

Configure Flue Gas Cleanup System

NO_x Removal :

- ☐ Hot-Side SCR
- ☒ None

SO₂ Removal :

- ☐ Wet Limestone FGD
- ☐ Lime Spray Dryer
- ☐ Duct Injection
- ☒ None

Particulate Removal :

- ☐ Cold-Side ESP
- ☒ Fabric Filter

Combined SO₂/NO_x :

- ☒ Copper Oxide
- ☐ NOHSO Process
- ☐ Electron Beam
- ☐ None

Previous

CheckList

Next

Ash Disposal :

- ☒ Landfill
- ☐ Ponding
- ☐ Co-Disposal with FGD Waste

FGD Waste Disposal

- ☐ Landfill
- ☐ Ponding

By-Product :

- ☐ Gypsum
- ☐ Sulfur
- ☒ Sulfuric Acid
- ☐ Fertilizer
- ☐ None

Previous

CheckList

Next

Input Parameters for Plant Performance

Parameter Description:		Value	Calc	Min	Max	Default
Gross Electrical Output	(MWg)	540	☐	250	1000	540
Steam Cycle Heat Rate	(Btu/kWh)	7880	☐	6000	10000	7880
Capacity Factor	(%)	75	☐	25	100	75
Excess Air for Furnace	(%)	20	☐	10	30	20
Leakage Air at Preheater	(%)	19	☐	0	40	19
Boiler Efficiency	(%)	X	☒	85	95	Calc
Unaccounted Losses in Boiler	(%)	0.005	☐	0	1	0.005
Flue Gas Temperature Exiting:						
	Economizer (°R)	1160	☐	1100	1250	1160
	Air Preheater (°R)	760	☐	660	860	760
Energy	Pulverizers (%MWg)	0.6	☐	0.1	1.0	0.6
Penalties:	Steam Pumps (%MWg)	0.65	☐	0.5	0.7	0.65
(Excluding	FD Fans (%MWg)	1.5	☐	0.7	4.0	1.5
Gas Cleanup	Cooling Sys. (%MWg)	1.8	☐	0.4	2.0	1.8
Systems)	Misc. (%MWg)	1.3	☐	0.0	2.0	1.3

Previous

Show ☒ Value ☐ Uncertainty

CheckList

Next

[Previous](#)

[CheckList](#)

[Next](#)

Coal Name **Illinois #6** Run-of-Mine Coal Cost \$ 22.48 per ton
 Rank **Bituminous** Transportation Cost \$ 3.00 per ton
 Black Lung Benefit \$ 2.00 per ton

	HHV	Carbon	Hydrogen	Oxygen	Sulfur	Nitrogen	Ash	Moisture
ROM Coal	10194	56.95	3.7	7.19	4.36	1.12	14.38	12.30

Previous

Read Coal File

CheckList

Next

Conventional Coal Cleaning (Page 1)

Parameter Description: Value Calc Min Max Default

Overhead:

Equipment Maintenance	(% TCC)	<u>7.0</u>	☐		<u>7.0</u>
Payroll	(% Payroll)	<u>30.0</u>	☐		<u>30.0</u>
Plant	(% Payroll, Main't & Chem.)	<u>26.0</u>	☐		<u>26.0</u>
Taxes and Insurance	(% TCC)	<u>4.0</u>	☐		<u>4.0</u>
Working Capital	(% TUC)	<u>25.0</u>	☐		<u>25.0</u>

Clean Coal Output	(dry tons/hr)	<u>500</u>	☐		<u>500</u>
Annual Operating Hours	(hours)	<u>3325</u>	☐		<u>3325</u>
Economic Life of Plant	(years)	<u>20</u>	☐		<u>20</u>
Interest Rate	(%)	<u>12.0</u>	☐		<u>12.0</u>
Refuse Disposal Cost	(88\$/wet ton)	<u>2.0</u>	☐		<u>2.0</u>
Indirect Capital Cost Factor	(% TDC)	<u>62.5</u>	☐		<u>62.5</u>

Previous

Show ☒ Value ☐ Uncertainty

CheckList

Next

Froth Flotation Circuit:

Flotation Cells Cost	(88\$/ft ³)	10630	☐			10630
Solids Concentration	(%)	5.0	☐			5.0
Effective Volume	(%)	85.0	☐			85.0
MIBC Usage	(lb/ton)	0.0	☐			0.0
MIBC Cost	(88\$/lb)	0.94	☐			0.94
#2 Fuel Oil Usage	(lb/ton)	0.0	☐			0.0
#2 Fuel Oil Cost	(88\$/lb)	0.10	☐			0.10

Previous

Show ☒ Value ☐ Uncertainty

CheckList

Next

Conventional Coal Cleaning (Page 2)

Cleaned Coal : ☒ Sulfur ☐ Ash Content : ☐ weight% ☒ lbs/MBh

Inherent Moisture Content of Raw Coal : %

Thermal Dryer Efficiency : %

Surface Moisture Exiting Thermal Dryer : %

Washability Data :

☒ Cumulative ☒ Include HHV Data
☐ Adjusted for Efficiency

Is undersize distribution data available for streams?

☐ Yes ☒ No

Stream	Exiting Surface Moisture (%)	Topsize of Stream (Inches)	Undersize Distribution Data (%)	Washability Error (%)
Coarse	4	1.5		0.3
Middling	8	0.375		3.5
Fine	12	0.0464		9.0

Previous

CheckList

Next

Washability Data

Specific Gravity	1.3	1.4	1.6	1.9	2.2			
TopSize (Inches) = 1.5								
Mass Yield (%)	28.1	74.5	85.2	90.0	100			
Ash Content (%)	4.7	7.2	8.9	10.4	16.3			
Sulfur Content (%)	3.01	3.33	3.60	3.82	4.92			
HHV (Btu/lb)	13252	12905	12668	12460	11639			
TopSize (Inches) = 0.375								
Mass Yield (%)	16.5	71.4	84.1	88.4	100			
Ash Content (%)	3.4	6.0	8.1	9.5	16.5			
Sulfur Content (%)	2.87	3.25	3.48	3.61	5.05			
HHV (Btu/lb)	13433	13072	12780	12585	11611			
TopSize (Inches) = 0.0464								
Mass Yield (%)	18.2	60.1	85.1	89.2	100			
Ash Content (%)	2.6	5.2	9.2	10.4	16.9			
Sulfur Content (%)	2.73	3.02	3.33	3.44	4.89			
HHV (Btu/lb)	13544	13183	12627	12460	11556			

[Previous](#)
[Read File](#)
[CheckList](#)
[Next](#)

Conventional Coal Cleaning (Page 4)

Coal Name : Illinois #6

Rank : Bituminous

Run-of-Mine Coal Cost \$ 22.48 per ton

Transportation Cost \$ 3.00 per ton

Black Lung Benefit \$ 2.00 per ton

Cleaning Cost \$ 6.05 per ton

	HHV	Carbon	Hydrogen	Oxygen	Sulfur	Nitrogen	Ash	Moisture
Cleaned Coal	10565	59.02	4.06	8.57	3.16	1.16	8.39	15.64

[Previous](#)
[CheckList](#)
[Next](#)

Parameter Description:	Value	Calc	Min	Max	Default
Ambient Air Temperature (°F)	80	☐	50	110	80
Pressure (PSIA)	14.696	☐	12	15	14.696
Specific Humidity (lb H ₂ O/lb dry air)	0.018	☐	0	0.03	0.018
Sulfur Retained in Bottom Ash (%)	X	☒	0	50	Calc
Conversion of S to SO ₂ (%)	97	☐	93	100	97
Ash Entering Flue Gas Stream (%)	60	☐	0	100	60
Fraction of C as CO (%)	0	☐	0	100	0
Fraction of C in Bottom Ash (%)	0	☐	0	100	0
Fraction of C Unburned (%)	X	☒	0	100	Calc
Nitrogen Oxide Emission Factor (lb equiv.NO ₂ /ton coal)	X	☒	5	40	Calc
Fraction of NO _x as NO (%)	90	☐	85	100	90
Emission Limit of SO _x (lb/MBtu)	X	☒	0	15	Calc
of NO _x (lb/MBtu)	0.08	☐	0	1	0.08
of TSP (lb/MBtu)	0.03	☐	0	1	0.03

Previous

Show ☒ Value ☐ Uncertainty

CheckList

Next

Basic Plant Economics Parameters

Parameter Description:	Value	Calc	Min	Max	Default
Year to Report Costs	1988	☐	1977	1989	1988
Direct Capital Cost (M 88\$)	X	☒	0	1000	Calc
Indirect Charge Factor (% TDC)	50	☐	0	100	50
Land Cost (88\$/Acre)	6800	☐	0	10000	6800
Delivered Cost of Coal (88\$/ton)	X	☒	0	65	Calc
Operating Labor (88\$/Man-Hour)	19.70	☐	0	30	19.70
O & M Cost (% TDC)	4.35	☐	0	10	4.35
Electricity Cost (Mills/kWh)	X	☒	0	100	Calc

Previous

Show ☒ Value ☐ Uncertainty

CheckList

Next

Parameter Description:	Value	Calc	Min	Max	Default
Maximum SO ₂ Removal Efficiency (%)	95	☐	90	100	95
Particulate Removal Efficiency (%)	50	☐	0	100	50
FGD Train Size (Kacfm)	425	☐	0	1000	425
Minimum Bypass (%)	10	☐	0	100	10
Number of Operating Absorbers	XX	☒			Calc
Number of Spare Absorbers	XX	☒			Calc
Gas Temp. Exiting Scrubber (°F)	XX	☒	0	1000	Calc
Gas Temp. After Reheater (°F)	178	☐	0	1000	178
Entrained Water Past Demister (% Water Evaporated)	0.79	☐	0	100	0.79
Gas Pressure Exiting FGD ("Water)	4	☐	0	10	4
Reagent Purity (% Weight)	92.8	☐	80	100	92.8
<div>Reagent Type</div> <div> ☐ CaO ☒ CaCO₃ </div>		<div>Oxidation Process</div> <div> ☒ Forced ☐ Natural </div>		<div>Bypass Control</div> <div> ☒ No Bypass ☐ Bypass </div>	
Previous		Show ☒ Value ☐ Uncertainty		CheckList Next	

Copper Oxide Parameters (Page 1)

Parameter Description:	Value	Calc	Min	Max	Default
Molar Stoichiometry of Cu/S (Ratio)	XX	☒	0.5	3	Calc
" " of NH ₃ /NO _x (Ratio)	XX	☒	0.5	2	Calc
Weight Fraction of Cu in Sorbent (%)	7	☐	1	10	7
Sorbent Attrition Rate (% Sorbent)	0.6	☐	0.1	2.0	0.6
Absorber Size (MW)	125	☐	100	250	125
Bed Height (inches)	36	☐	18	42	36
Superficial Velocity (ft/sec)	4.2	☐	3	6	4.2
Absorber Pressure Drop (acfm)	XX	☒			Calc
Recovery Plant Conversion Eff. (%)	99.5	☐	90	99.99	99.5

Previous

Show ☒ Value ☐ Uncertainty

CheckList

Next

Parameter Description:	Value	Calc	Min	Max	Default
------------------------	-------	------	-----	-----	---------

Regenerator Efficiency	(%)	99.2	☐	30	99.99	99.2
Gas Temp. into Regenerator	(°F)	XXXX	☒			Calc
Sorbent Temp. into Regenerator	(°F)	XXXX	☒			Calc
H ₂ Content of Syngas	(%)	XXXX	☒			Calc
Syngas Conversion Efficiency	(%)	99.5	☐	30	99.99	99.5

Regeneration Gas Used : ☒ Methane ☐ Syngas 1 ☐ Syngas 2

Previous

Show ☒ Value ☐ Uncertainty

CheckList

Next

Copper Oxide Parameters (Page 3)

Parameter Description:	Value	Calc	Min	Max	Default
------------------------	-------	------	-----	-----	---------

Sorbent Cost	(88\$/lb)	5	☐	2	7	5
Total Contingency Factor	(% TDC)	55.0	☐	0	100	55.0
Engineering Fees	(% TDC)	7	☐	0	20	7
Architectural Costs	(% TDC)	25	☐	0	50	25
Construction Expenses	(% TDC)	25	☐	0	50	25
Contractor Expenses	(% TDC)	9.6	☐	0	20	9.6
Royalties	(% TDC)	0.5	☐	0	2	0.5

Previous

Show ☒ Value ☐ Uncertainty

CheckList

Next

Parameter Description:	Value	Calc	Min	Max	Default
------------------------	-------	------	-----	-----	---------

TSP Removal Efficiency	(%)	☒			Calc
Energy Penalty	(% MWg)	☒			Calc
Engineering & Arch. Fees	(% TDC)	☒			Calc
Contractor Expenses	(% TDC)	☒			Calc
Total Contingency Factor	(% TDC)	☒			Calc

Previous

Show ☒ Value ☐ Uncertainty

CheckList

Next

Overall Economics (Page 1)

Parameter Description:	Value	Calc	Min	Max	Default
Inflation Rate	(%)	☐	0	25	0
Real Escalation Rate	(%)	☐	0	20	0
Real Bond Rate	(%)	☐	4.6	15	4.6
Real Preferred Stock Rate	(%)	☐	5.2	15	5.2
Real Common Stock Rate	(%)	☐	8.7	25	8.7
Preferred Stock Fraction	(%)	☐	15.0	100	15.0
Debt Fraction	(%)	☐	50.0	100	50.0
Federal Tax Rate	(%)	☐	36.7	50	36.7
State Tax Rate	(%)	☐	2.0	10	2.0
Ad Valorem Tax Rate	(%)	☐	2.0	10	2.0
Investment Tax Credit	(%)	☐	0.0	20	0.0
Booklife	(Years)	☐	15.0	60	15.0

Previous

Show ☒ Value ☐ Uncertainty

CheckList

Next

☐ Straight Line

Depreciation Schedule :

Year	Value	Year	Value	Year	Value
1988	0.075	1998	0.045	2008	0
1989	0.069	1999	0.045	2009	0
1990	0.064	2000	0.045	2010	0
1991	0.059	2001	0.045	2011	0
1992	0.055	2002	0.045	2012	0
1993	0.051	2003	0.044	2013	0
1994	0.047	2004	0.044	2014	0
1995	0.045	2005	0.044	2015	0
1996	0.045	2006	0.044	2016	0
1997	0.045	2007	0.044	2017	0

Previous

CheckList

Next

Uncertainty Control

Sampling Method : ☒ Median Latin Hypercube

☐ Random Latin Hypercube

☐ Monte Carlo

Sample Size :

20

☒ **Parametric Analysis :** Cases ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☒ 5

Uncertainty Distributions : ☒ Use All ☐ Use Only the Following :

☒ Coal Properties

☐ Coal Cleaning

☐ Plant Performance

☐ Ambient Conditions & Emissions

☒ Plant Economics

☐ Flue Gas Cleanup : Copper Oxide

☐ Particulate Removal : Fabric Filter

☐ Reagent

☐ NOx Cleanup

☐ Overall Economics

☐ Solid Waste Disposal

Previous

CheckList

Next

Parameter Description:		Value	Calc	Min	Max	Default
Engineering Fees	(% TDC)	XXXX	☒			Calc
Architectural Fees	(% TDC)	XXXX	☒			Calc
Construction Expenses	(% TDC)	XXXX	☒			Calc
Contractor Expenses	(% TDC)	5.0	☐			5.0
Total Contingency Factor	(% TDC)	XXXX	☒			Calc
Startup Cost	(M \$)	XXXX	☒			Calc
Royalties	(% TDC)	0.0	☐			0.0
Working Capital	(% TDC)	5.0	☐			5.0
% Solids from TSP Device		100	☐	90	100	100

Previous

Show ☒ Value ☐ Uncertainty

CheckList

Next

Copper Oxide Performance

Exit

Probability ☒ Excluded ☐ Included

Copper Oxide Process Performance Mean Values

Parameter	Probability	
	Excluded	Included
Ammonia Usage, mole/hr	58.3	68.3
Methane Usage, mole/hr	670.651	670.651
CO2 Usage, mole/hr	0	0
Acid Production, tons/hr	19.6	19.6
SO2 Removal Efficiency, %	90.2	90.2
NOx Removal Efficiency, %	88.0	88.0
Inlet Flue Gas Flow Rate, acfm	2.31M	2.31M
Outlet Flue Gas Flow Rate, acfm	2.73M	2.73M
Cu / S Ratio	2.40	2.40
Sorbent Flow Rate, lbs/hr	971.5K	971.5K
Makeup Flow Rate, lbs/hr	582.9	582.9
Coal Credit, tons/hr	8.59	8.59
Energy Penalty, % MWg	2.14	2.14
Operating Absorbers	4	2.9
Spare Absorbers	1	0.5

Exit

Copper Oxide Operating and Maintenance Cost

M 88\$	Probability	
	Excluded	Included
Maintenance Labor	0.34	0.33
Maintenance Material	0.51	0.49
Operating Labor	0.45	0.45
Admin. & Support Labor	0.24	0.24
Fixed Operating Costs	1.54	1.51
Sorbent Cost	3.18	3.18
Ammonia Cost	0.42	0.52
Methane Cost	1.63	1.63
Miscellaneous Costs	0.42	0.42
Variable Operating Cost	5.65	5.76
Sulfuric Acid Credit	2.09	2.09
Coal Credit	0.30	0.30

Exit

Copper Oxide Capital Cost

M 88\$	Probability	
	Excluded	Included
Ammonia Handling Equip.	0.63	0.63
Absorber Cost	10.01	9.39
Flue Gas Handling	5.65	5.65
Solids Heater	1.13	1.13
Regenerator	0.46	0.46
Transport Lines	0.80	0.80
Incremental Air Preheat	1.43	1.43
Sulfuric Acid Plant	6.76	6.76
General Facilities	1.12	1.08
Total Process Capital	27.99	27.33
Engineering Fees	1.48	1.40
Architect Fees	0.37	0.36
Construction Expense	3.17	3.08
Acid Plant Indirect Costs	2.96	2.96

Exit

Copper Oxide Performance : Flue Gas Temperature Exiting Absorber

Exit

☒ CDF
 ☐ PDF
 ☐ Statistics

Copper Oxide Performance : Flue Gas Flow Entering Absorbers

Exit

☒ CDF ☐ PDF ☐ Statistics

Copper Oxide Performance : Flue Gas Flow Entering Absorbers

Exit

☐ CDF ☒ PDF ☐ Statistics

Copper Oxide Performance : Total Levelized Revenue Requirement

Cum. Prob.	Probability	
	Excluded	Included
0 %	14.34	6.776
5 %	14.34	7.160
10 %	14.34	10.18
20 %	14.34	11.58
30 %	14.34	12.14
40 %	14.34	12.67
50 %	14.34	13.35
60 %	14.34	14.02
70 %	14.34	14.60
80 %	14.34	14.88
90 %	14.34	15.41
95 %	14.34	15.55
100 %	14.34	17.53
Mean	14.34	12.95
Variance	2.317	7.059
St. Dev.	0	2.657

Exit

☐ CDF ☐ PDF ☒ Statistics

Copper Oxide Performance : Total Levelized Revenue Requirement

Exit

☒ CDF ☐ PDF ☐ Statistics

Economics Summary

Item	Total Capital Cost 1988 \$M	Total Non-Utility O & M 1988 \$M/Yr	Electric Utility Cost 1988 \$M/Yr	Total Level-ized Revenue Requirement 1988 \$M/Yr	Total Level-ized Revenue Requirement Mills/kWh	Total Capital Cost \$/kW
Base Plant	526.96	47.05	-2.55	98.97	33.70	1022
Fabric Filter	33.65	0.49	0.33	4.31	1.47	65
Copper Oxide	107.66	21.69	2.22	35.04	11.93	209
Solid Waste	9.60	1.78	0.00	2.78	0.94	19
Total	677.86	71.01	0.00	141.10	48.04	1315
Pollution Control Cost	150.90	23.96	2.55	42.13	14.34	292

Exit

Performance Summary

Exit

Units ● lbs/MBtu ○ tons/hr

Species ● Ash ○ SO₂ ○ NO_x

END

**DATE
FILMED**

8 / 10 / 92