A Guide to Running Alternative Technology and Pesticide Scenarios with ASM

by

Bruce A. McCarl
Professor
Department of Agricultural Economics
Texas A&M University

(409) 845-7504 (fax)
mccarl@tamu.edu

Uwe Schneider
Research Associate
Department of Agricultural Economics
Texas A&M University

uwe@tamu.edu

Chi-Chung Chen
Research Associate
Department of Agricultural Economics
Texas A&M University

chichung@tamu.edu

This research was supported by USDA - subcontract No. 97008 and by the Texas Agricultural Experiment Station

TABLE OF CONTENTS

LIST OF TABLES AND FIGURES

A GUIDE TO RUNNING ALTERNATIVE TECHNOLOGY AND PESTICIDE SCENARIOS WITH THE AGRICULTURAL SECTOR MODEL (ASM)

Bruce A. McCarl
Uwe Schneider
Chi-Chung Chen

1. INTRODUCTION

This report provides a description of how to do alternative technology and pesticide orientated runs with the **A**gricultural **S**ector **M**odel (**ASM**), a static equilibrium, linear programming model of the agricultural sector in the United States. ASM depicts the allocation of land, labor, and water within the agricultural sector and generates estimates of agricultural prices, quantities produced, consumers' and producers' surplus, exports and imports. It has been developed to investigate the economic impacts of technological change, trade policy, commodity programs, environmental policy, and global warming on the U.S. agricultural sector.

This document is intended to provide users with information on how to use ASM to analyze alternative technology and pesticide related scenarios. Instructions are given about how to define policy scenarios, modify the ASM data to reflect a scenario, run the scenarios with the model, and select/create scenario and cross-scenario output.

This document is not intended to be a stand-alone document. There are four documents in particular to which users should have access, and which will be referenced in the course of this manuscript. These documents are:

A) ASM Model Description — McCarl, B.A., C.C. Chang, J.D. Atwood, and W.I. Nayda. Documentation of ASM: The U.S. Agricultural Sector Model. Resource Policy Analysis, 1994. This document is a technically oriented description of ASM. It discusses the assumptions behind the base data, the assumptions behind the model, and the specific computerized structure of ASM.

B) GAMS Reference Manual — Brooke, A., D. Kendrick, and A. Meeraus. GAMS: A User's Guide. Boyd and Fraser Publishers, Version 2.25, 1993. ASM is implemented in the GAMS programming language. This reference manual provides the most recent (but somewhat behind the times) guide to the GAMS language. Other information regarding GAMS also appears on the GAMS Development Corporation's World Wide Web site (www.gams.com).

C) McCarl, B.A. , A. Meeraus, W. I. Nayda, R. Ramen, and P. Steacy "So Your GAMS Model Didn't Work Right: A Guide to Model Repair." Draft book, Department of Agricultural

Economics, Texas A&M University, 1996. Draft document which covers how to debug a GAMS model. This document shows how to fix a number of things that users might run into, such as compiler error problems, abnormal terminations, memory faults, etc. This is a several hundred page document, covering a lot of cases which might come up in ASM use that are just not practical to cover when discussing the alternative runs set up here.

D) McCarl, B.A., T.H. Spreen. "Applied Mathematical Programming Using Algebraic Systems." Course Notes for Agricultural Economics 641, Texas A&M University, College Station, TX, 1996. This is a basically completed draft textbook which covers GAMS and algebraic mathematical programming modeling. The conceptual and theoretical basis for modeling techniques used in constructing ASM are discussed therein. This document provides an introduction to GAMS applications which may be useful to some users.

1.1. How To Use This Manual

This manual is divided into four Chapters. The first Chapter is introductory and provides the necessary information to install ASM on various computer systems. The second Chapter gives some background information about GAMS - the computer language in which ASM is programmed - and about the model itself. We recommend to “explore” ASM while reading Chapter two. Chapter three describes the general approach for setting up alternative runs using a simple mathematical programming example. The fourth Chapter applies the general approach from Chapter three in an ASM context. Two examples of technical change are integrated along with a complete program description. Understanding Chapter four is essential for users who are interested in setting up their own comparative analyses in ASM. Chapters two and three, however, provide the foundation for the material in Chapter four.

This document is designed to serve users with differing levels of expertise and interest. For those users who have little knowledge and experience about GAMS and comparative analysis but show significant interest in learning, we recommend reading the whole text from the beginning to the end. In addition, these users may turn to the tutorial in the Brooke et al. GAMS manual, or Chapter five of McCarl and Spreen. Users with advanced knowledge may skip over parts of Chapter two and over Chapter three but concentrate on Chapter four.

1.2. Installation Guide

Here we present a step-by-step guide to installing ASM on your computer, followed by notes on particular computer systems. A good part of the material here assumes that users have knowledge of a text editor and DOS, and know enough about GAMS and computer usage to install the program.[1] Deficiencies in general computer use, DOS, text editing, etc. can be made up by consulting computer specialists or reference manuals. Users needing to modify the GAMS batch files should consult Brooke, Kendrick and Meeraus and Appendix A. Users needing to figure out how to alter GAMS instructions and find compiler errors should consult Chapter 3 of McCarl et al.

1.2.1. Step-by-Step Installation

The ASM system can be installed on virtually any computer since there are GAMS versions for most computer types. Here we cover installation starting from GAMS installation.

1.2.1.1. Step 1: Obtain and Install GAMS

Obtain GAMS software from the GAMS Development Corporation, Washington DC (202-342-0180). You will have a choice of solvers; purchase CPLEX if you wish the best linear programming solver and MINOS if you wish to solve nonlinear models. Then install GAMS following the GAMS Corporation's installation instructions. In turn test GAMS as described in the *GAMS Installation and System Notes* that comes with the software.

1.2.1.2. Step 2: Make the ASM File Storage Area

ASM consists of a set of source files and possibly a set of study-specific files. We recommend you set up a set of subdirectories as follows:

Main Subdirectory	ASM	Holds all ASM studies and code
Subservient subdirectories within the ASM directory	SOURCE	Holds all ASM source code
	SMALL	Hold ASM small model
	BWEEVIL	Holds Boll Weevil eradication study example
	POSTHARV	Holds post harvest loss reduction study example
	OTHER	Other study names as needed

[1] A word processor may not be a suitable text editor because, if the margins are not wide enough, it may reformat ASM files so that they are unreadable by GAMS. Also, care must be taken so that only ASCII versions of the files are saved. As an alternative, the DOS text editor works well.

Also make a subdirectory called "t" under each of the above subservient subdirectories.

1.2.1.3. Step 3: Obtain the ASM Software

The most recent version of ASM will need to be obtained on numerous disks or downloaded from the ASM home computer system which is scout.tamu.edu. One should make arrangements through the ASM development team or the authors.

1.2.1.4. Step 4: Copy the ASM Files Onto Your Computer

Once ASM has been obtained, load the ASM files listed in Appendix A onto your computer. Do this by copying all files from any disks obtained or by using file transfer protocol (FTP) if downloading from the ASM home computer system. If needed, use a DOS version of PKUNZIP or an UNIX version of "tar" to uncompress any compressed files.

1.2.1.5. Step 5: Adjust the Solver References

To do this requires several steps:

1. The base version of ASM is set up for use with CPLEX on a UNIX machine. If you do not have CPLEX, use a text editor to alter the files ASMMODEL, ASMSOLVE, and PESTLOOP, removing any references to CPLEX[2]. Either delete these lines (typically OPTION LP=CPLEX;) or change the CPLEX string to the name of the most capable LP solver that you acquired with the GAMS version, whether it be BDMLP, OSL, MINOS5, etc.

2. If the model is to be nonlinear, you need to have a GAMS license to MINOS5. Once you have this, edit the ASMMODEL and PESTLOOP files.

1.2.1.6. Step 6: Adjust the INCLUDE Instructions to Reflect Your Computer Path Names

ASM uses the GAMS $INCLUDE statement to incorporate numerous code segments. These include statements commonly reference the location on the base computer where the ASM files are stored. For example the statement

[2] CPLEX may appear in either upper or lower case.

$include “/bigmac/asm/source/soilfree/demand.96”

incorporates the file demand.96 into the GAMS code to be executed, assuming that code can be found on the associated hard disk in the file storage subdirectory named

/bigmac/asm/source/soilfree/

You need to change this storage reference so it corresponds to the path on your computer where the files are stored. Thus in all include statements alter the path name

/bigmac/asm/source/soilfree/

to a pointer to your ASM source path. If you created the ASM file storage area as suggested in 1.2.2, you would use:

c:\ASM\source\

These changes would need to be made in the following files:

ALLOFIT
ASMMODEL
ASMSOLVE
ASMREPT
ASMCOMPR
PESTLOOP
RUNREPORT

The GAMS compiler will aid you in this endeavor as it will give error messages until the paths are correct.

1.2.1.7. Step 7: Run the Small Model

Go to the subdirectory where the small model is stored. Make sure you have completed step 1.3.2.6, and that there is a subdirectory called t. Execute the job r.bat by typing r.bat at the prompt c:\ASM\SOURCE\; then hit the enter key. ASM then should go through all the way to the end. If you are having trouble, contact Bruce McCarl at (409) 845-1706, to get some help with the initial installation.

1.2.1.8. Step 8: Do Your Studies

ASM is now installed and you may now conduct ASM runs.

1.2.2. Notes on Computer Systems

The above material is general to all computer systems. This section addresses additional considerations regarding particular computer systems.

1.2.2.1. PC's Windows 95/OS2

PC machines running DOS and OS2 require some special treatment. Unfortunately, due to the size of ASM, the GAMS model can malfunction with DOS or Windows 3.11. We have been able to run the full ASM model in a DOS window in Windows 95**.**

An alternative platform that we have found to be stable is the OS/2 operating system. Once the system is installed (follow the instructions on the box), one simply moves to a DOS full screen session and executes ASM from the DOS prompt in the usual fashion described above. Under OS/2 we have been able to load, compile and begin execution of full linked versions of ASM. Using the segmented version and the BDMLP linear solver requires only 3-6 hours.

1.2.2.2. UNIX

The home system for ASM is an HP7000 workstation, running UNIX. The only real difference in using UNIX is the path names which we talked about changing in section 1.2.6 above. The UNIX batch file, R, is also distributed along with the software.

1.2.3. ASM Debugging and Basis Support

GAMSCHK and GAMSBAS are useful utilities for basis support, drawing the map reproduced in Appendix D and debugging the model. If one wishes to install these, one should obtain the GAMSBAS and GAMSCHK utilities installation instructions, and user documents from the World Wide Web page agrinet.tamu.edu/mccarl, and follow the directions. In using either of these utilities, one needs to be careful not to accidentally run ASM with solves in loops because these utilities generate a tremendous amount of output for every element of the loop.

2. BACKGROUND

2.1. GAMS - Why Did We Use It?

ASM is implemented in the GAMS algebraic modeling language. GAMS is an acronym for General Algebraic Modeling System. The preface of Brooke, Kendrick and Meeraus states that GAMS "is designed to make the construction and solution of large and complex mathematical programming models more straight forward for programmers and more comprehensible to users of models from other disciplines... ." Furthermore, they say GAMS allows "concise algebraic statements of models in a language that is easily read by both modelers and computers..." and that GAMS "can substantially improve productivity of modelers and greatly expand the extent and usefulness of mathematical programming applications and policy analysis, and decision making." Let us now review why we choose to use GAMS generally and for ASM specifically.

As a language GAMS possesses two important attributes. First, it requires that the entire problem be cast in an algebraic format. Second, it automatically handles many functions needed when doing computational mathematical programming. Our reasons for using GAMS as the language in which ASM is implemented can be explained along these two lines.[3]

2.1.1. Why Use Algebraic Modeling?

The major advantages of using an algebraic language involve the ability to concisely state problems in an abstract, general fashion, largely independent of the data and the exact application context. One can produce a formulation independent of the specific problem size which initially can be used with smaller test problems, but which will later permit the full problem to be analyzed. The dimensions of the problem can grow as additional cases and data are added without modifying the algebraic specification of the problem.

The above statements are easily illustrated in the case of the resource allocation problem. The algebraic form of the resource allocation problem is

$$\begin{aligned} \text{Max} \quad & \sum_j c_j X_j \\ \text{s.t.} \quad & \sum_j a_{ij} X_j \le b_i \quad \text{for all } i \\ & X_j \ge 0 \quad \text{for all } j \end{aligned}$$

[3] We will briefly review this topic here; readers interested in more on GAMS should refer to McCarl and Spreen, and Brooke, Kendrick and Meeraus.

where j identifies production possibilities, I identifies available resources, c_j is the profit when one unit of production possibility j is manufactured, X_j is the number of units of production possibility j that are made, a_{ij} is the number of units of resource I required per unit of X_j, and b_i is the endowment of resource I. Collectively this algebraic formulation maximizes total profit by determining how much of each production possibility to produce while staying within the bounds of available resources. This algebraic setup is valid for all resource allocation models regardless of the dimensions of I and j of the contents of the data parameters $a_{ij\,,}$ c_j and b_i. The GAMS counterpart of the algebraic model is shown in Table 2.1 (Tables are located at the end of this Chapter).[4] We may explore correspondence between the GAMS and the algebraic formulations by setting up a Table relating the algebraic symbols and their GAMS counterparts with line number references (Table 2.2). The data in Table 2.2 shows a one-to-one correspondence between each and every element of the algebraic model and the elements in the GAMS model. Also note that the GAMS model is more understandable as longer names are given for the symbols.

Table 2.1 can also be used to show the expansion flexibility of GAMS. The Table depicts the resources of interest as the four specified in line 5. If one wished to add capacity, for example, then one would add the name on line 5, an endowment on line 14 or 15, and usage numbers after lines 23 and 28. The model part of this GAMS formulation (lines 30-47 in Table 2.1) would not require modification and is general for all resource allocation problems, regardless of size. Thus, because of the algebraically based language, one can write size-independent GAMS formulations.

The advantages gained by using an algebraically based language are accompanied by several disadvantages. Algebraic modeling and summation notation are difficult for some users. Some people will always desire to deal with the exact problem context, not an abstract general formulation.

2.1.2. Why Use GAMS?

Now, why use GAMS? GAMS requires algebraic modeling and thus, has some of algebraic modeling's advantages and disadvantages. At the same time, GAMS can be tailored to the problem, so its use introduces additional advantages which overcome some of the disadvantages.

[4] True understanding of Table 2.1 requires knowledge of GAMS. An introduction in the context of the resource allocation problem is given in Chapter 5 of McCarl and Spreen.

2.1.2.1. Problem Formulation and Changes in Problem Formulation

GAMS aids both in initially formulating a problem and subsequently revising formulations. GAMS facilitates specification and debugging of an initial formulation by allowing the modeler to begin with a small data set. Then, after verifying correctness, the modeler can expand the model to a much broader context. For example, one could alter the model in Table 2.1 so it covered 10,000 production possibilities and 1,000 resources without changing the code after line 29. Thus, one can test over small data sets, then move on to the full implementation. Both small and large data sets are defined for ASM to permit such testing during implementation of new features.

Second, GAMS makes it easy to alter problem specifications. Historically, large models have been difficult to modify because traditional modeling approaches make it difficult to add new constraints or variables. On the other hand, GAMS allows one to add model features much more simply. Generally, with GAMS, modelers do not try to make a complete formulation the first time around. Instead GAMS modelers usually start with a small formulation and then add structural features as demanded by the analysis. GAMS also allows model code to be transferred between problem contexts (e.g., the code used in McCarl et al.'s ASM model has been slightly rewritten for the Forest and Agricultural Sector Optimization Model).

2.1.2.2. Automated Computational Tasks

GAMS automatically performs calculations; checks the formulation for obvious flaws; chooses the solver; formats the programming problem to meet the exact requirements of the solver; causes the solver to execute the job; saves and submits the "advanced basis" (an intermediate step in calculations) when doing related solutions; and permits usage of the solution for report writing. Note when the model in Table 2.1 is solved, the following functions are carried out automatically without the need for user instructions:

a) the objective function is automatically computed (i.e., as in the case of the (PRICE(PROCESS)-PRODCOST(PROCESS) term in line 39 Table 2.1);

b) the model is set up in a format readable by the solver;

c) the solver is started and told to read the data then the problem is solved and the answer written; and

d) the answer is read by GAMS and displayed.

These are complex computational tasks which would require considerable computer programming effort if another modeling language were used. They are done by GAMS automatically without any commands on the user's behalf other than a GAMS startup instruction. GAMS also verifies the correctness of the algebraic model statements and allows empirical examination of the equations. Furthermore, since GAMS has been implemented on machines ranging from 286 to 486 PCS up to CRAY super computers, it allows portability of a model formulation between computer systems. Switching solvers is also simple, requiring either:

a) alteration of the solver option statement (see Brooke et al. page 105); or

b) changing from using linear to nonlinear programming by altering the SOLVE statement wording from "using LP" to "using NLP" (see Brooke et al. page 98).

The ASM model exploits these features as both linear and nonlinear versions have been developed, and solvers such as CPLEX, OSL and MINOS5 have been used at various times on both HP UNIX-based work stations and PC-based OS2 implementations.

2.1.2.3. Facilitates Documentation and Later Use of Models

One other convenient feature of GAMS is its self-documenting nature. Perhaps the largest benefit GAMS has over traditional modeling techniques is that modelers can use longer names for variables, equations and sets while including within the model comments on model structure, data definitions etc., allowing a more complete and readable problem description within the model itself. Modelers partially document model structure, assumptions, and any calculation procedures used in the report writing as a byproduct of the modeling exercise within the source GAMS model file.[5] Comment statements can be inserted in the model by placing an asterisk in column one, followed by the comment (e.g., text identifying data sources or particular assumptions being used), or by using $ontext/$offtext sequences (see discussion in Brooke et al. page 42). Such documentation makes it easier for either the original author or others to alter the model structure and update data. Note how the documentation and definition content of Table 2.1 is well beyond that inherent in the algebraic form. GAMS models are usually readable to non-technical users.

2.1.2.4. Allows Use by Varied Personnel

Modeling personnel can be rare. Many detailed GAMS applications have been set up by modeling experts and subsequently are used by policy-makers with minimal assistance from modeling experts. Often, given proper documentation and instruction, clerical labor and nontechnical problem analysts can handle an analysis.

[5] Appendix I of McCarl et al. covers practices that can be used to enhance this documentation.

2.2. General Notes on ASM

2.2.1. The Concept of ASM

Conceptually, ASM is a price-endogenous mathematical programming model based on the spatial equilibrium models developed by Samuelson and Takayama and Judge as reviewed by McCarl and Spreen and Norton and Schiefer. ASM was originally designed to simulate competitive equilibrium solutions under a given set of demand and supply conditions. The ASM objective function is the summation of all areas beneath the product demand curves minus the summation of all areas under the import and factor supply curves. Collectively, such an objective represents the area between the aggregate demand and supply curves to the left of their intersection. Economists have frequently referred to that area as producers' and consumers' surplus. This objective function represents a social welfare function which, within the limitations of consumers' and producers' surplus (Just, Hueth and Schmitz), measures the benefits of producers' and consumers' from producing and consuming agricultural commodities. The production and consumption sectors are assumed to be made up of a large number of individuals operating under competitive market conditions. When producers' plus consumers' surplus is maximized, the model solution represents an intersection of the supply and demand curves and, thus, simulates a perfectly competitive market equilibrium. Prices and quantities for all factors of production and outputs are endogenously determined by the supply and demand relationships for all the commodities in the model.

Various versions of the Agricultural Sector Model(ASM) have been used to investigate the economic impacts of technological change, trade policy, commodity programs, the introduction of new products, environmental policy, global warming on the U.S. agricultural sector.

The basic structure of ASM is summarized in the following underlying mathematical equations. Consider a sector containing agricultural commodity markets with the quantity traded being (Q_i) produced by production possibilities X_{nk} using land (L_k), labor (R_k) and water (W_k) as well as other variable inputs (including seed, fertilizer, etc.). Let integrable inverse demand functions for the commodities exist and be given by:

(1) $P_i^Q = R_i(Q_i)$, for all commodities i

Furthermore, assume inverse supply relations exist for land, labor and water by region which are given by

$$P_k^L = \text{"}_k(L_k) \text{ , for all regions } k$$
(2) $$P_k^R = \$_k(R_k) \text{ , for all regions } k$$
$$P_k^W = T_k(W_k) \text{ for all regions } K$$

where P^Q, P^L, P^R, and P^W are the corresponding commodity and regional factor prices.

The ASM formulation is essentially given by the following optimization problem:

$$
\begin{array}{llll}
(3) & \text{Max} & \sum_i \int R_i(Q_i)\,dQi - \sum_n \sum_k c_{nk} X_{nK} - \sum_k \int \mu_k(L_k)\,dL_k - \sum_k \int \#_k(R_k)\,dR_k \quad \sum_k \int T_k(W_k)\,dW_k & \\
(4) & \text{subject to} & Q_1 - \sum_n \sum_k y_{ink} X_{nk} & \leq 0, \text{for all } i, \\
(5) & & \sum_n X_{nk} - L_k & \leq 0, \text{for all } k, \\
(6) & & \sum_n f_{nk} X_{nk} \quad R_k & \leq 0, \text{for all } k \\
(7) & & \sum_n g_{nk} X_{nk} - W_k & \leq 0, \text{for all } k,
\end{array}
$$

where the parameters other than those discussed above are:

c_{nk}. the per unit variable input cost of producing the nth production possibility in the kth region;
$y_{ink.}$ the per acre yield of the ith commodity using the nth production possibility in the kth region;
f_{nk}. the per acre labor use of the nth production possibility in the kth region; and
g_{nk}. the per acre water use of the nth production possibility in the kth region.

The problem contains non-negative decision variables Q_i which give the amount of commodity i consumed; X_{nk} the amount of the nth production possibility employed in the kth region; L_k which is the amount of land supplied in region k; R_k the amount of labor supplied in region k; and W_k the amount of water supplied in region k.

Equation (3) is the objective function where producers' plus consumers' surplus is optimized. Equation (4) depicts the supply/demand balance equations which are defined for all commodities. Equations (5)-(7) balances usages of land, labor and water with their respective supplies. On solution, this model yields market clearing levels for Q, X, L, R and W which simulate a competitive equilibrium. The shadow prices on equation (4) give market clearing prices for the commodities while the shadow prices on (5) - (7) give regional specific land, labor and water prices.

2.2.2. Basic Structure of ASM Computer Files

It is important for analysts to know the structure of the ASM computer files before attempting to run the model in applied policy studies. Because of the large number of data, model components and report writing features which compose ASM, we find it convenient to decompose it into different components with a set of files. The current version of the ASM is implemented using a software package called GAMS (General Algebraic Modeling System). The GAMS language provides a concise way to describe the model structure and, permits modifications for application-specific needs.

The main computer files in ASM include:

(I) ALLOFIT

(II) ASMMODEL

(III) ASMSOLVE

(IV) ASMREPT

(V) ASMCOMPR

The ALLOFIT file contains all set definitions, and the basic data required to run the ASM. The ASM model structure is contained in ASMMODEL file and it includes the model's objective function and constraints. The ASMSOLVE file on the other hand, contains the solution procedure. ASMREPT and ASMCOMPR contain report writing instructions. Full details on the ASM files are provided in the documentation by McCarl et. al. It should be noted that all these files are saved and run in a batch mode.

Table 2.1. GAMS Formulation of Resource Allocation Example

```
 SET    PROCESS       TYPES OF PRODUCTION PROCESSES
                          /FUNCTNORM , FUNCTMXSML , FUNCTMXLRG
                          ,FANCYNORM , FANCYMXSML , FANCYMXLRG/
        RESOURCE      TYPES OF RESOURCES
                          /SMLLATHE,LRGLATHE,CARVER,LABOR/ ;

 PARAMETER PRICE(PROCESS)     PRODUCT PRICES BY PROCESS
                   /FUNCTNORM  82, FUNCTMXSML  82, FUNCTMXLRG  82
                   ,FANCYNORM 105, FANCYMXSML 105, FANCYMXLRG 105/
           PRODCOST(PROCESS)     COST BY PROCESS
                   /FUNCTNORM  15, FUNCTMXSML  16  , FUNCTMXLRG  15.7
                   ,FANCYNORM  25, FANCYMXSML  26.5, FANCYMXLRG  26.6/
           RESORAVAIL(RESOURCE) RESOURCE AVAILABLITY
                   /SMLLATHE 140, LRGLATHE  90,
                    CARVER   120, LABOR    125/;

 TABLE RESOURCES(RESOURCE,PROCESS) RESOURCE USAGE

               FUNCTNORM   FUNCTMXSML  FUNCTMXLRG
SMLLATHE          0.80       1.30        0.20
LRGLATHE          0.50       0.20        1.30
CARVER            0.40       0.40        0.40
LABOR             1.00       1.05        1.10
 +             FANCYNORM   FANCYMXSML  FANCYMXLRG
SMLLATHE          1.20       1.70        0.50
LRGLATHE          0.70       0.30        1.50
CARVER            1.00       1.00        1.00
LABOR             0.80       0.82        0.84;

 POSITIVE VARIABLES
          PRODUCTION(PROCESS) ITEMS PRODUCED BY PROCESS;
 VARIABLES
          PROFIT              TOTALPROFIT;
 EQUATIONS
          OBJT                  OBJECTIVE FUNCTION ( PROFIT )
          AVAILABLE(RESOURCE)   RESOURCES AVAILABLE ;

  OBJT..  PROFIT =E=
             SUM(PROCESS,(PRICE(PROCESS)-PRODCOST(PROCESS))
                           * PRODUCTION(PROCESS)) ;

  AVAILABLE(RESOURCE)..
      SUM(PROCESS,RESOURCES(RESOURCE,PROCESS)*PRODUCTION(PROCESS))
                       =L= RESORAVAIL(RESOURCE);

 MODEL RESALLOC /ALL/;
 SOLVE RESALLOC USING LP MAXIMIZING PROFIT;
```

Table 2.2 Correspondence Between Algebraic Model And GAMS Code

Algebraic Model Symbol	Gams Model Item	Table 2.1 Line Number
j	PROCESS set	1
I	RESOURCE set	3
c_j	PRICE(PROCESS)-PRODCOST(PROCESS) data and computation	7-12,39
X_j	PRODUCTION(PROCESS) variable	31
a_{ij}	RESOURCES(RESOURCE,PROCESS) data table	17-28
b_i	RESORAVAIL(RESOURCE) parameter	13-15
Objective Function	OBJT expression	33,38-40
Resource Constraints	AVAILABLE(RESOURCE)	36,42-44

3. GENERAL NOTES ON RUNNING ALTERNATIVE RUNS[6]

Most models are built for use in a comparative analysis. Multiple model solutions are generated where in each solution (hereafter called a scenario) alterations are made to the data, constraints, or variables. Comparative statics studies compare reactions by the modeled entity, by examining differences between scenario-based solutions and a "base case" solution. GAMS has facilities for doing repeated comparative analyses and ASM exploits them. This Chapter covers those facilities in a simpler case than ASM for expository purposes. Chapters 4 covers the same steps in the ASM context.

3.1. Structure of a Comparative Analysis

The basic structure of a comparative analysis is outlined in Figure 3.1 (figure is located at the end of this Chapter). The first three boxes reflect a conventional GAMS program where the initial data and model are set up, and then a solution is executed. The comparative analysis begins with box four. The analysis is set up by identifying the scenarios, defining data for the scenarios, and saving the data that is to be changed between scenarios. In turn, for each scenario the data are restored to original levels, then the data and model differences for that scenario are imposed, the model is solved, the solution is reported, and data are saved for a comparative report. Finally, the comparative report is displayed.

Below we present an example of such a comparative analysis, in the context of the basic resource allocation problem used in Table 2.1 above (see McCarl and Spreen, Chapter 5, section 5.2 for a discussion of the problem). Here we solve a base case scenario, then solve scenarios with the labor and large lathe constraints in the AVAILABLE equation suppressed, and finally solve a scenario with 25% higher prices for fancy chairs.

Table 3.1 (Tables are located at the end of this Chapter) presents the GAMS specification for a comparative analysis. The specification consists of the following steps:

1) Set up the model with any modifications needed to accommodate the alternative runs to be done. In this example the model is set up in lines 2-52 of Table 3.1 (line 1 is empty and is not shown). These lines are a repeat of the model in Table 2.1 but with one major modification. Namely in line 43 the AVAILABLE(RESOURCE) constraint has been shown with a condition ($

[6] This section assumes the reader can read GAMS code.

sign) attached to permit us to suppress constraints in those equations[7]. This condition causes the constraints to be active only when there is a non-zero value entered in the resource availability parameter — RESORAVAIL(RESOURCE). Thus we will be able to suppress the constraints by setting the resource availability to zero. Beyond this we have added report writing.[8]

2) Define the parameters of any reports to be done on the scenarios or across the scenarios — lines 55-66.

3) Save any data which will be changed in the analysis scenarios in lines 72-77. Here we define arrays in which we will save the resource availability and prices, then put data into those arrays.

4) Define the scenarios (line 70) and any associated data (lines 79-85). Note: The scenarios are (a) a base case which leaves all the data alone (base); (b) a no labor constraint case which eliminates consideration of labor as a constraint (nolabor) ; (c) a no large lathe constraint case which eliminates consideration of the large lathe as a constraint (nolrglathe) ; and (d) a high price fancy chair scenario (hifancy).

5) Proceed one at a time to solve the scenarios — lines 87-105. This portion of the code consists of several substeps:

   (a) Reestablish data to base levels — Lines 88-89

   (b) Change the data so it reflects the particular scenario — Lines 90-92

   (c) Solve the scenario — Line 94

   (d) Summarize the results in a report — lines 96-98

   (e) Enter results for this scenario into a report comparing results across scenarios — lines 101-104

6) Output the comparative report — line 108.

[7] See Brooke et al. pages 92-95 for an explanation of the $ operator.

[8] We also suppress excessive printouts with lines 48-50.

The resultant output is given in Table 3.2. Note the results show varying profits, production patterns, resource usages, and resource values across the scenarios.

3.2. Components of a Comparative Analysis

The above example shows the components of a comparative analysis involving alternative runs. Such a structure generally involves steps beyond the establishment and setup of the base model which include: (a) data alterations; (b) model structural component activation and deactivation; (c) comparative report writing ; and (d) repeated model solutions. Each merits separate discussion.

3.2.1. Data Alterations

One important process when running multiple scenarios involves revising data. Modelers must be aware that when revising data, GAMS changes all data items permanently regardless of their initial values. If one goes through a loop and changes a data item, that value is permanently changed. Thus the scenarios would become cumulative. To avoid this, one needs to reset the data to original values before beginning the execution of a scenario. This occurs in the above example through the commands involving saving data on lines 72-73 and 74-77, and the commands for restoring the saved values in lines 88-89. If this were not done the data changes would accumulate during the scenarios. For example the GAMS commands

```
SCALAR LAND /100/;
PARAMETER   SAVELAND;
            SAVELAND = LAND;
SET LANDCHANGE    SCENARIOS FOR CHANGES IN LAND/R1,R2,R3/
PARAMETER VALUE(LANDCHANGE) PERCENT CHANGE IN LAND
                  /R1 +10 , R2 + 20 , R3 +30/;
LOOP ( LANDCHANGE,
            LAND = LAND * (1 + VALUE ( LANDCHANGE ) / 100. ) );
```

results in land equaling 110, 132 and 171.6 during the loop, with the original value of 617 being lost. However, alteration of the calculation statement so it operated from a saved parameter value

```
      LAND = SAVELAND * (1 + VALUE ( LANDCHANGE ) / 100. )
```

results in values of 110 , 120, and 130.

One other important item involves computations. Whenever a SOLVE command is entered, GAMS automatically recomputes all terms specified in the optimization model equations (in all the .. expressions such as in lines 39-45 of Table 3.1). However, no other computations are repeated when setting up the model. This leads to two concerns. First, all calculations that involve data changed within the scenarios must either be included in the model equations (the .. terms) or the calculations must be repeated. This can be illustrated as follows.

Suppose a model was set up as follows:

```
 1               Price(Crop)   = 2.00;
 2               Yield(Crop)   = 100;
 3               Cost(Crop)    = 50;
 4               PROFIT (Crop) = Price(Crop)*Yield(Crop)-Cost(Crop);
 5               Equations
 6                                       Obj         objective function
 7                                       Land        Land available;
 8               Positive Variables      Acres(Crop) Cropped Acres
 9               Variables               Objf        Objective function;
10                     Obj.. Objf=E=Sum(Crop,PROFIT(Crop)*Acres(Crop));
11               Land  Sum(Crop, Acres(Crop))=L=100;
12               Model FARM /ALL/
13         *solve number 1
14               SOLVE FARM USING LP Maximizing OBJF;
15               Price (CROP)=2.50;
16         *solve number 2
17               SOLVE FARM USING LP Maximizing OBJF;
18               PROFIT (Crop)=Price (CROP)*Yield(Crop)-Cost(Crop);
19         *solve number 3
20               SOLVE FARM USING LP Maximizing OBJF;
```

In this case the second solve in line 16 would yield identical results to those obtained in the first solve in line 13, since even though PRICE is changed, the PROFIT term is not recomputed. The third solution resulting from line 19 would differ since PROFIT is recomputed in line 17. One could also fix this by changing the objective function specification to:

```
Obj..Objf=E=SUM(Crop,(Price(CROP)*Yield(CROP)-Cost(CROP))Acres(CROP);
```

In that case the full objective function terms are automatically recomputed in every solve. To the extent possible ASM is designed to avoid such problems. Users may avoid such problems by recalculating any computed items.[9]

Similarly, variable bounds and scaling factors are not automatically recomputed unless one reissues the statement defining .LO , .UP and .SCALE cases. Thus, when any of these items are computed and the data entering those computations are revised, their calculation needs to be repeated. ASM does this for some of the bounds and scaling factors in FASCALE and ASMCALRN.

[9] Procedures to identify items which need to be recomputed are discussed in section 8.5 of this document.

3.2.2. Changing Model Structure — Activation and Deactivation of Structure

Many comparative studies involve model structure modifications. One of the big advantages of using a modeling system is the ability to add/delete constraints, variables, or equation terms and reanalyze the problem. GAMS permits such modifications to be done using $ controls as in lines 43-45 of Table 3.1. Suppose we consider an alternative example. Suppose the following lines are used in a GAMS problem:

```
SCALAR                  ISITACTIVE  TELLS WHETHER ITEMS ARE ACTIVE
                        /0/;
CONDEQ$ISITACTIVE..     SUM(STUFF,X(STUFF)) =L= 1;
EQNOTH(INDEX)..         SUM(STUFF,R(INDEX,STUFF)*X(STUFF)) +
                             4*SUM(STUFF,Y(STUFF))$ISITACTIVE =L= 50;
```

This addition would cause the CONDEQ equation and the Y term in the EQNOTH equations to only appear in the empirical model when the ISITACTIVE parameter was nonzero. Thus, the sequence

```
ISITACTIVE = 0;
SOLVE MODELNAME USING LP MAXIMIZING OBJ;
ISITACTIVE =1;
SOLVE MODELNAME USING LP MAXIMIZING OBJ;
```

would cause the model to be solved with and without the CONDEQ constraint and the Y term in the EQNOTH equations.[10]

3.2.3. Solving Repeatedly

More than one model can be solved in a run. Thus, one can stack solve statements as in the example immediately above or loop over solves as in Table 3.1 in line 87.

3.2.4. Comparative Report Writing

The development of a comparative report writer is usually attractive when doing multiple runs. Report writing commands always use values from the most recent solution, so one must save the data if comparative reports are desired. The code in Table 3.1 contains such a report writer. In that case a place to store the report data is defined (parameter COMPAR) indexed over the loop set (RUNS)-see line 74. In turn, during loop execution the COMPAR array is saved with scenario-dependent values of variables and shadow prices in lines 101-104. Finally, when the output is displayed a comparison across scenarios appears (Table 3.2).

[10] See Brooke et al. pages 92-95 for more explanation of the $ operator.

3.3. Comparative Runs in Complex Models

The above implementation of a comparative set of runs in a single file works for small simple models, but is not totally satisfactory for a complex model such as ASM. ASM contains thousands of lines of data, and the model solution from a "cold start" without a good initial basis can take a number of hours, particularly when farm program target price convergence is required. As a consequence, the alternative run structure of ASM, while conceptually the same, is operationally different.

ASM is divided into 7 modules as discussed in section 2.2 . The first module, ALLOFIT, integrates more than 40,000 lines of base data. The second module, ASMMODEL, defines the programming model structure and gets the initial farm program convergence solution. The third module, ASMSOLVE contains the iterative solution procedure. The fourth modul, ASMREPT, defines report writer items. The fifth module, PESTLOOP, conducts the alternative runs analysis. The sixth modul, ASMCOMPR, saves the ASM results for each scenario run. The seventh modul, RUNREPORT, writes comparative reports between scenarios. Thus, the ASM system goes through the same steps as in Figure 3.1, but uses a more complex computerized structure.

Table 3.1. Example of Comparative Run

```
 SET    PROCESS       TYPES OF PRODUCTION PROCESSES
                          /FUNCTNORM , FUNCTMXSML , FUNCTMXLRG
                          ,FANCYNORM , FANCYMXSML , FANCYMXLRG/
        RESOURCE      TYPES OF RESOURCES
                          /SMLLATHE,LRGLATHE,CARVER,LABOR/ ;

 PARAMETER PRICE(PROCESS)     PRODUCT PRICES BY PROCESS
                   /FUNCTNORM  82, FUNCTMXSML  82, FUNCTMXLRG  82
                   ,FANCYNORM 105, FANCYMXSML 105, FANCYMXLRG 105/
           PRODCOST(PROCESS)     COST BY PROCESS
                   /FUNCTNORM  15, FUNCTMXSML  16  , FUNCTMXLRG  15.7
                   ,FANCYNORM  25, FANCYMXSML  26.5, FANCYMXLRG  26.6/
           RESORAVAIL(RESOURCE) RESOURCE AVAILABLITY
                   /SMLLATHE 140, LRGLATHE  90,
                    CARVER   120, LABOR    125/

 TABLE RESOURUSE(RESOURCE,PROCESS) RESOURCE USAGE

               FUNCTNORM   FUNCTMXSML  FUNCTMXLRG
SMLLATHE          0.80       1.30        0.20
LRGLATHE          0.50       0.20        1.30
CARVER            0.40       0.40        0.40
LABOR             1.00       1.05        1.10
 +             FANCYNORM   FANCYMXSML  FANCYMXLRG
SMLLATHE          1.20       1.70        0.50
LRGLATHE          0.70       0.30        1.50
CARVER            1.00       1.00        1.00
LABOR             0.80       0.82        0.84;

 POSITIVE VARIABLES
          PRODUCTION(PROCESS) ITEMS PRODUCED BY PROCESS;
 VARIABLES
          PROFIT              TOTALPROFIT;
 EQUATIONS
          OBJT                  OBJECTIVE FUNCTION ( PROFIT )
          AVAILABLE(RESOURCE)   RESOURCES AVAILABLE ;

  OBJT..  PROFIT =E=
             SUM(PROCESS,(PRICE(PROCESS)-PRODCOST(PROCESS))
                           * PRODUCTION(PROCESS)) ;

  AVAILABLE(RESOURCE)$RESORAVAIL(RESOURCE)..
      SUM(PROCESS,RESOURUSE(RESOURCE,PROCESS)*PRODUCTION(PROCESS))
                       =L= RESORAVAIL(RESOURCE);

 MODEL RESALLOC /ALL/;
 option solprint=off;
 option limrow=0;
 option limcol=0;
 SOLVE RESALLOC USING LP MAXIMIZING PROFIT;
 set type   types of chairs               /functional,fancy/
     item   items for reports             /level,production,usage,value/
     map(type,process)    map of chair types to processes
                  /functional.(FUNCTNORM , FUNCTMXSML , FUNCTMXLRG)
                   fancy     .(FANCYNORM , FANCYMXSML , FANCYMXLRG)/;

 parameter resourstat(resource,item) resource status
           chairs(type)              production of chairs;

           resourstat(resource,"usage")=available.l(resource);
           resourstat(resource,"value")=available.m(resource);
           chairs(type)=sum(map(type,process),production.l(process));

 display chairs,resourstat;

 set runs  /base,nolabor, noLRGLATHE ,hifancy/

 parameter savRESORAv(RESOURCE) saved resource availability
           savprice(process)    saved prices
           compar(item,*,runs)       comparative report;

           savRESORAv(RESOURCE)=RESORAVAIL(RESOURCE);
           savprice(process)=price(process);

 table adjust(*,runs) alternative run configuration
            base nolabor noLRGLATHE  hifancy
smllathe     0
lrglathe     0                1
labor        0     1
fancy        0                          0.25
functional   0                              ;

 loop(runs,
           RESORAVAIL(RESOURCE)=savRESORAv(RESOURCE);
           price(process)=savprice(process);
           RESORAVAIL(RESOURCE)$adjust(resource,runs)=0;
           price(process)$sum(map(type,process),adjust(type,runs))=
           price(process)*(1+sum(map(type,process),adjust(type,runs)));

           SOLVE RESALLOC USING LP MAXIMIZING PROFIT;

           resourstat(resource,"usage")=available.l(resource);
           resourstat(resource,"value")=available.m(resource);
           chairs(type)=sum(map(type,process),production.l(process));
           display chairs,resourstat;
           compar("level","profit",runs)=profit.l;
           compar("usage",resource,runs)=resourstat(resource,"usage");
           compar("value",resource,runs)=resourstat(resource,"value");
           compar("production",type,runs)=chairs(type);
      );
           option decimals=2;
           display compar;
```

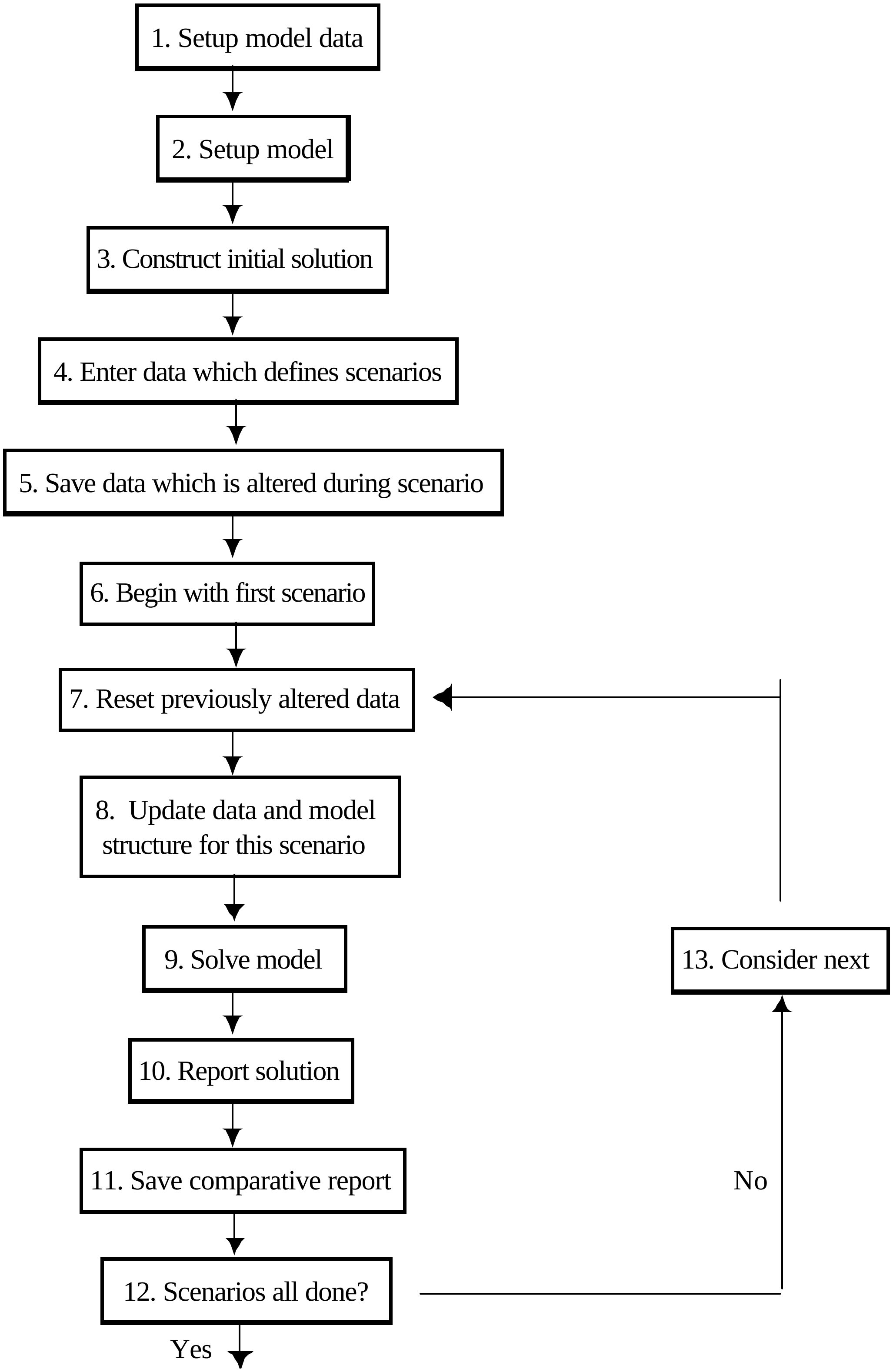
Table 3.2. Comparative Report Writing Output

```
----    108 PARAMETER COMPAR        comparative report

                                 BASE     NOLABOR  NOLRGLATHE     HIFANCY

LEVEL     .PROFIT        10417.29    11830.43    11002.82    12798.83
PRODUCTION.FUNCTIONAL       62.23      176.60       41.20        2.44
PRODUCTION.FANCY            78.20                  103.52      119.02
USAGE     .SMLLATHE        140.00      140.00      140.00      140.00
USAGE     .LRGLATHE         90.00       90.00       90.00       90.00
USAGE     .CARVER          103.09       70.64      120.00      120.00
USAGE     .LABOR           125.00      125.00      125.00       97.93
VALUE     .SMLLATHE         33.33       57.39        5.09       48.66
VALUE     .LRGLATHE         25.79       42.17       42.17       40.58
VALUE     .CARVER                                   34.63       19.45
VALUE     .LABOR            27.44       27.44       49.08
```

Figure 3.1 Outline of a Comparative Model Analysis


1. Setup model data
2. Setup model
3. Construct initial solution
4. Enter data which defines scenarios
5. Save data which is altered during scenario
6. Begin with first scenario
7. Reset previously altered data
8. Update data and model structure for this scenario
9. Solve model
10. Report solution
11. Save comparative report
12. Scenarios all done?
Yes
No
13. Consider next

14. Display comparative report

4. SETTING UP AN ASM ALTERNATIVE RUN

4.1. General Notes On Computerized File Structure For Alternative Runs

To implement a technological change study, we need to create two additional files and modify the existing batch file. First, we generate a PESTLOOP file that will contain the gams code for all comparative runs for our study. Second, we will specify report writing options in a file called RUNREPORT. Finally, we have to modify the existing batch file RUN.BAT.

4.1.1. PESTLOOP-file

PESTLOOP contains the specification of the different scenarios created to examine the economic impacts of technological change on the variables of interest. The structure of code is similar to that listed in boxes 4 - 12 of Figure 3.1. It involves the following procedure:

Step 1: The names of all possible scenarios or runs are defined.
Step 2: The particular subset of runs to implement for the current study is defined.
Step 3: The data are entered which will define the scenarios.
Step 4: Parameters are defined into which we will save the original data that will be changed during the scenario runs.
Step 5: Parameters and sets are defined for use in comparative report writing.
Step 6: The solve loop is begun and all potentially altered data are restored to their original values.
Step 7: Data alteration are performed in accordance with scenarios.
Step 8: The model is solved.
Step 9: Comparative reports are solved.

4.1.2. RUNREPORT-File

RUNREPORT relates to box 14 in Figure 3.3 and accomplishes several tasks. Foremost, it displays comparative results of the analysis. For each scenario, these results include absolute solution values of relevant variables as well as absolute and percentage deviations of these values from the base scenario values[11]. In addition, RUNREPORT downsizes the considerable ASM output. A complete listing of all ASM report items is neither necessary nor desirable to analyze the economic implications of

[11] This is a convention but not a restriction. One could also compute changes with respect to any of the alternative scenario solution values.

any technology or policy change. Which of the many report items ought to be displayed depends on the nature and objective(s) of the particular study.

RUNREPORT does not have to be created from new. We use a pre-setup version and activate or deactivate the display of various report items. Furthermore, one could define new report items and add them to the existing ones. New items can be created by imposing conditions on existing report items. For example, one might be interested in reports on the changes of the crop mix between different scenarios. RUNREPORT already contains a parameter CROPMIXPCN (SUBREG, CROP, RUN) that computes the percentage change in crop mix for each subregion, crop, and scenario. Suppose, the analysis consists of 10 different scenarios. The above parameter, then, could display up to 12,800 numerical values (64 subregions, 20 crops, 10 scenarios). One useful addition would be to create a new parameter that lists only values that exceed a certain threshold, i.e. 3%. The way this can be implemented in GAMS is as following:

```
PARAMETER CROPMIXPC3(SUBREG,CROP,RUN)  cases where change is greater than 3% ;
CROPMIXPC3(SUBREG,CROP,RUN)  $(ABS(CROPMIXPCN (SUBREG,CROP,RUN)) gt 3)
 = CROPMICPCN (SUBREG, CROP, RUN);
*DISPLAY    CROPMIXPC;
 DISPLAY    CROPMIXPC3:2:2:1;
```

where CROPMIXPC3, then lists only values greater than 3% for each scenario, crop, and subregion.

4.1.3. RUN.BAT-File

The batch file that runs ASM is listed in the first part of Table 4.1. This batch file makes use of the GAMS save and restart capabilities and therefore can be started and stopped at any point in the process. The usual way that alternative runs are done in ASM is that the first three boxes of Figure 3.1 are executed by including the GAMS commands in the batch file up through the ASMREPT.[12] This means the model is setup and solved, then the solution is saved. In turn comparative statics runs are done using alternative versions of the PESTLOOP and executing the remainder of the batch file as illustrated by the second batch file in Table 4.1. The system is designed that way to avoid the need for repeating the time consuming execution of the data setup and initial solution construction steps. The PESTLOOP module can be restarted utilizing the stored results from the preceding modules.

4.2. Use of Names And Labels When Setting Up Alternative Runs

Technological change occurs when the inputs used decrease without changing output, or when output increases with decrease in inputs used or when the input mix changes. Thus, for modeling alternative technologies, one needs to alter data on input and/or output parameters. To make those

[12] In the ASM context this is done using previously created batch files.

alterations effective in ASM, parameter, set, or variable names used in PESTLOOP must be consistent with those used in ASM. Names for sets and set items in ASM or file references for them can be found at the beginning of ALLOFIT. Variable names are declared in ASMMODEL.

An example may be in order. Suppose, one wishes to examine alternative pesticide usage scenarios. Between scenarios one would probably have different levels in crop yields and chemical costs according to the assumptions. Therefore, one must change yields and chemical costs at the beginning of each scenario run. In ASM, data on crop inputs and crop yields are stored in a parameter called CBUDDATA. This parameter has six dimensions: ALLI, SUBREG, CROP, WTECH, CTECH, TECH. The first dimension refers to "all items" including output and input items, thus, including yields and chemical costs. SUBREG defines all subregions of the US (usually consistent with states) and CROP arrays the 20 different crops used in ASM. WTECH, CTECH, and TECH specify irrigation, crop budget and technology alternatives, respectively.

ASM uses CHEMICALCO as name for the input item chemical cost. To update the level of chemical costs in ASM one has to use this particular name. Also, if the change in pesticide use only affects a certain location, or/and a certain crop one has to use the relevant ASM names for this location and crop. By replacing SUBREG with the ASM name for the relevant location, one could limit the data change to this location only. To model a 10% increase in chemical costs for wheat growers in Kansas, the following GAMS code would appropriate:

```
CBUDDATA (“CHEMICALCO”,”KANSAS”,”WHEAT”,WTECH, CTECH, TECH)
 = CBUDDATA (“CHEMICALCO” ,”KANSAS” ,”WHEAT” ,WTECH, CTECH, TECH)*1.1;
```

This calculation would take the existing data on chemical costs for Kansas wheat growers and multiply it by 1.1 which is equivalent to a 10% increase. The change applies to both irrigated and dryland acres and to all CTECH and TECH specifications.

Similarly, one could alter crop yields. CBUDDATA ("WHEAT", "KANSAS", "WHEAT", "DRYLAND", CTECH, TECH) contains the data on yield of wheat on dryland in Kansas. A 5% yield increase would be modeled as following:

```
CBUDDATA (“WHEAT”, ”KANSAS”, ”WHEAT”, “DRYLAND”, CTECH, TECH)
  = CBUDDATA (“WHEAT”, ”KANSAS”, ”WHEAT”, “DRYLAND”, CTECH, TECH) * 1.05;
```

Note that wheat yields from irrigated land are unaffected in the above example. However, the adjustment takes place for all CTECH and TECH options. If wheat yields were to increase not only in Kansas but also in Missouri and Mississippi, one could make the parameter adjustment as shown below.

```
SET FEWREGIONS (SUBREG)     / KANSAS, MISSISSIPP, MISSOURI /;
CBUDDATA (“WHEAT”, FEWREGIONS, ”WHEAT”, “DRYLAND”, CTECH, TECH)
  = CBUDDATA (“WHEAT”, FEWREGIONS, ”WHEAT”, “DRYLAND”, CTECH, TECH) * 1.05;
```

Again, names for individual subregions have to be consistent with those defined in ASM. The set name FEWREGIONS , however, was chosen arbitrarily. To indicate that FEWREGIONS is a subset of SUBREG, the latter set is added in parentheses to the set declaration statement.

4.3. An Example: Post-Harvest Loss Reduction And Boll Weevil Eradication

Here, we will present the modification of the program to run analyses for two alternative technologies: boll weevil eradication (BWE) and post-harvest loss reduction (PHL) technology. In setting up this example, we will make the assumption, that both cases can be represented by regionally specific alterations in the crop yields and costs. To save space and to illustrate how one could combine analyses of technology or policy change we will incorporate the two alternative technologies in the same PESTLOOP file. Alternatively, one could set up two different files, one for post-harvest loss reduction and one for boll weevil eradication. Even though we use two very specific technologies, one can easily apply the same technique to model many other crop production technologies.

We assume loss reduction is equivalent to an increase in yields. The scenario assumptions, therefore, will involve different levels of yield gain for the crops under consideration. Here, we analyze post-harvest loss reduction strategies for tomatoes, potatoes, and citrus, where citrus represents both oranges and grapefruit. No changes will be made on the cost (input) side. The magnitude of reduced losses (increased yields) is assumed to be the same across all regions. Boll weevil eradication involves both changes in yields and chemical costs. It applies only to cotton production. The magnitude of yield and cost changes is different across different regions. For regions where no significant boll weevil eradication is present, yields and input costs stay the same.

4.3.1. GAMS Implementation of PESTLOOP

4.3.1.1 Step 1: Define Scenario Superset

The first action in setting up PESTLOOP is to create a set of all possible scenario runs for the analysis. In our example this set is called ALLRUNS and contains 15 items (lines 11-26). Base represents the reference scenario where no data are changed. The next 12 items refer to the scenarios dealing with post-harvest loss reduction. In order to easily recognize the associated assumption, names of the post-harvest loss reduction technology were chosen to reflect the affected crops and the magnitude of the loss reduction. In particular, TOM stands for tomatoes, CIT for citrus, POT for potatoes and JOINT for all of them. Thus, TOM1 labels the scenario for which tomato post-harvest losses are decreased by 1%.

The boll weevil eradication study involves two alternative scenarios: national eradication of the boll weevil and eradication in the Texas High plains only. We named these scenarios BELTWIDE and HIGHPLAINS, respectively.

Note that the names for different scenarios could be chosen arbitrarily as long as they conform to GAMS standards, i.e. do not exceed 10 characters and do not use illegal characters.

4.3.1.2. Step 2: Define Scenario Subset

The scenario subset identifies the exact runs to be done. Scenarios not carried over from the full ALLRUNS set to the subset are not executed during this particular run. One can retain the capability to execute old runs simply by expanding the PESTLOOP program but only need run the ones of current interest. The specific scenarios to execute are named in the RUN subset. In Table 4.2, ten scenarios have been activated in RUN (line 39).

4.3.1.3. Step 3: Enter Data

First, we create a parameter ADJUST that will contain the adjustments for each scenario (line 46). The dimensions of ADJUST allow us to change any of the items in the budgets (ALLI), in any region (SUBREG), for any crop (CROP) and let these changes differ across scenarios. Then, we initialize ADJUST by setting it to zero (line 47). In turn we will fill ADJUST with data for post-harvest loss reduction and boll weevil eradication.

4.3.1.3.1 Data on post-harvest loss reduction

We assume post-harvest loss reduction scenarios may be simulated by increasing yields for the crops in question across the nation. Our assumed changes in yields vary by scenario and crop. Thus, when entering the data we define a Table PHLADJ (lines 54-68) where each row represents a scenario and each column a crop. The values in the table represent the percentage yield change. For example, the Table entrees in line 66 list all assumptions associated with scenario JOINT3. In particular, we have a 3% yield increase for fresh tomatoes, potatoes, fresh oranges, and fresh citrus.

Once the Table is completed we copy these data into the ADJUST parameter as shown in lines 72-74. In putting in the data we divide the Table entries by 100 to adjust from percentage change to proportional differences (line 74).

4.3.1.3.2 Data on boll weevil eradication

Next, we enter the data that describe the changes due to boll weevil eradication technology (lines 83-101). These data involve changes in yields and chemical costs for cotton production and differ among regions. Hence, we have three dimensions for each data point, one to reflect the kind of parameter that is changed (yield, chemical costs), one to reflect the subregion in which these particular changes occur, and one to reflect the particular scenario (BELTWIDE, HIGHPLAINS). There are several ways to enter more than two dimensional data. In lines 85-100 we use dots to incorporate two dimensions into each row of Table BWEADJ. The table entrees, then, consist of one column dimension (ALLI) and two row dimensions (SUBREG, ALLRUNS). Lines 85-99 display the adjustment data for the beltwide eradication scenario while line 100 displays the adjustment data for the Texas High Plains eradication only scenario. As demonstrated for the post-harvest loss reduction technology, we need to call up these data into the ADJUST parameter (lines 105-107). Note that boll weevil eradication applies only to cotton production, thus, we substitute “COTTON” for the CROP dimension of CBUDDATA.

4.3.1.4. Step 4: Sets And Parameters Are Defined For Use In Comparative Report Writing

Lines 115-180 declare sets and parameters for comparative reports between scenarios. Unless there is need for specific output, this GAMS code produces a comprehensive output that can be used for many alternative studies. Descriptions of the various sets and parameters are given within the program code. These are generally not changed between runs and should be left alone.

4.3.1.5. Step 5: Save Original (Base) Data

In each scenario, data are changed. To avoid improper carryover of changed data between loops, we restore all modified data to their original (base) values at the beginning of each loop (step 7). In this demonstrative study we modify yields for tomatoes, potatoes, and citrus as well as yields and chemical costs for cotton. All modified data are contained in CBUDDATA. Therefore, we create a new parameter SCBUDDATA (line 188) and save the original values of CBUDDATA to this parameter (lines 197-198).

ASM contains several farm program options such as target prices, deficiency payments, and loan rates. Here, we did not want to include farm program features in our analysis. Consequently, all farm program parameters were set to zero (lines 204-209).

4.3.1.6. Step 6: Solve Loop

The actual loop procedure begins in line 216. Data are restored in lines 221-222. Subsequently, data modifications are made according to each scenario assumption (lines 224-228). Technical changes due to boll weevil eradication or post-harvest loss reduction are proportional changes. Therefore, we multiply the original data by (1 + ADJUST) to get the absolute value of the changed parameter. If the changes associated with the technological or policy change were given in absolute values one could modify the existing parameters by simply adding the ADJUST and the CBUDDATA parameter values. Finally, lines 230-240 contain the code to run the ASM model (Step 8) and to write reports for each scenario (Step 9).

4.3.2. Modification Of RUNREPORT

The RUNREPORT module for our example is listed in Table 4.3. The module consists of five major parts. The first part (lines 6-61) declares parameters for comparative reports. The second part (lines 67-266) contains the equations to compute each parameter. Note that some equations include $-control options. These options are used to avoid dividing by zero and to impose certain thresholds for some parameters as described in section 4.1.2. The third part (lines 272-336) rounds report items to an appropriate number of decimal places. The fourth part (lines 342-379) defines display options where the first value specifies the number of decimals to be displayed, the second value the number of row dimensions, and the third value the number of column dimensions to be used. The sum of the second and third value must not exceed the number of dimensions of the parameter. The fifth part has the display statements (lines 386-427). Asterisks in front of a display statement deactivate it.

The example file in Table 4.3 contains a fairly comprehensive list of output parameters that can be used for many comparative studies. Table 4.4 shows the output for the boll weevil eradication study example that is produced if one uses the RUNREPORT version from Table 4.3. Note that some display items are deactivated by placing an asterisk in front of the display statement. If one wants to re-activate currently suppressed items he simply removes the asterisk from the first column. In case one wants to set up additional parameter that are not contained in the original version of RUNREPORT, more effort is required. Section 4.1.2 showed how one can reduce the output of an existing parameter by imposing a minimum value condition. To incorporate this parameter into RUNREPORT one first declares the new parameter in part one of RUNREPORT, second puts the calculation statements in part two, third defines display options in part three and finally enters a display statement in part four of RUNREPORT.

4.3.3. Verification

Verification is of fundamental importance for all models and ASM makes no exception. An error free run does not imply that the model is setup correctly. The basic model structure of ASM has been verified by its authors. Users of ASM who run alternative scenarios with different technology or policy parameters need to verify that the alternative scenarios were specified in a proper way. Using the post-harvest loss reduction and the boll weevil eradication example, we will demonstrate how one could verify the alternative scenario setup.

First, we create parameters that will display data changes on a scenario basis. Here, we use the names PHLDATA and BWEDATA to capture the two respective technologies. The parameter statements are put in PESTLOOP in any place above the loop statement. The dimensions of the parameters depend on how specific the technology in question is. Post-harvest loss reduction is assumed to occur on a national scale. Hence, we don’t need to have a dimension for subregions. Instead we could pick one subregion and check the changes of relevant technical parameters across all scenarios.

```
PARAMETERS
PHLDATA (RUN,ALLI,CROP)            PHL DATA ASSUMPTIONS BY SCENARIO
BWEDATA (RUN,SUBREG,ALLI,WTECH)    BWE DATA BY SCENARIO ;
```

The next step is to define the new parameters. We want to capture the technical changes for each scenario. Technical parameters that are altered by post-harvest loss reduction technology or by boll weevil eradication are contained in CBUDDATA. The base scenario settings are contained in SCBUDDATA. Thus, we can use these two parameters to calculate the percentage change of CBUDDATA across scenarios. The statements have to be placed within the loop after the statement that adjusts CBUDDATA for each scenario (after line 228 in PESTLOOP).

```
PHLDATA (RUN,ALLI,CROP)
  $((SCBUDDATA(ALLI,"CALIFORNIS",CROP,"IRRIG","BASE","0")) and
     (ADJUST("PHL",ALLI,"CALIFORNIS",CROP,RUN)))
 = (  CBUDDATA(ALLI,"CALIFORNIS",CROP,"IRRIG","BASE","0")
    -SCBUDDATA(ALLI,"CALIFORNIS",CROP,"IRRIG","BASE","0"))* 100 /
        (SCBUDDATA(ALLI,"CALIFORNIS",CROP,"IRRIG","BASE","0")) ;


BWEDATA (RUN,SUBREG,ALLI,WTECH)
  $((SCBUDDATA(ALLI,SUBREG,"COTTON",WTECH,"BASE","0")) and
     (ADJUST("BWE",ALLI,SUBREG,"COTTON",RUN)))
 = (  CBUDDATA(ALLI,SUBREG,"COTTON",WTECH,"BASE","0")
    -SCBUDDATA(ALLI,SUBREG,"COTTON",WTECH,"BASE","0"))* 100 /
           (SCBUDDATA(ALLI,SUBREG,"COTTON",WTECH,"BASE","0")) ;
```

Note that the dimensions of PHLDATA and BWEDATA have to be compatible with the dimensions of CBUDDATA. PHLDATA has no dimension for SUBREG, WTECH, CTECH, and TECH. We need to choose an appropriate member of these sets and substitute it for the set name as done above. Appropriate set members are those that do not imply zero parameter values for CBUDDATA. For example, a subregion in which no oranges are grown is not appropriate to display changes in orange yields after post-harvest loss reduction for oranges is in effect. Here, we choose South California (CALIFORNIS) for it produces potatoes, tomatoes, and citrus. In addition, we choose the irrigation technology ("IRRIG") for irrigation alternatives, the "base" option for crop budget alternatives, and "0" for crop technology. Similarly, we set up the computation for BWEDATA. The $-conditions are included in order to avoid dividing by zero and to reduce computational time.

As a final step, we need to display PHLDATA and BWEDATA. The display commands have to be placed below the scenario loop, i.e. below line 241 in the current PESTLOOP. In addition, one could deactivate ASMCALRN, ASMSOLVE, ASMREPT, and ASMCOMPR (lines 230-240) to avoid the model being executed during the validation process.

```
DISPLAY     PHLDATA;
DISPLAY     BWEDATA;
```

Tables 4.5 and 4.6 show the output for this particular validation procedure. Since the values conform to the assumptions made for each scenario of the particular technology we have verified that the technical parameters were altered correctly. Subsequently, we can reactivate lines 230-240 in PESTLOOP and proceed in our analysis.

4.3.4. Combining Post-Harvest Loss Reduction And Boll Weevil Eradication Technology

This section shows how one can conduct a combined analysis of post-harvest loss reduction and boll weevil eradication . Let us consider the case, where one wants to jointly examine the effects of post-harvest loss reduction and beltwide boll weevil eradication. We want the adjustment for beltwide boll weevil eradication to be active across all scenarios (including the base scenario) and in addition assume various degrees of post-harvest loss reduction for the alternative scenarios.

First, we setup additional scenarios, expanding the set ALLRUNS in PESTLOOP. This is shown below where the first line corresponds to line 26 in Table 4.2. The subsequent lines contain the new scenarios and a short description of the scenario assumption. Note that the "/;" has to be moved 13 lines down to the new end of ALLRUNS.

```
BELTWIDE      BELTWIDE (NATIONAL) BOLL WEEVIL ERADICATION,
NBWEBASE      NATIONAL BOLL WEEVIL ERADICATION (NBWE) ,
NBWETOM1      NBWE + TOMATO YIELDS INCREASED BY 1% NATIONALLY,
NBWETOM3      NBWE + TOMATO YIELDS INCREASED BY 3% NATIONALLY,
NBWETOM5      NBWE + TOMATO YIELDS INCREASED BY 5% NATIONALLY,
NBWEPOT1      NBWE + POTATO YIELDS INCREASED BY 1% NATIONALLY,
NBWEPOT3      NBWE + POTATO YIELDS INCREASED BY 3% NATIONALLY,
NBWEPOT5      NBWE + POTATO YIELDS INCREASED BY 5% NATIONALLY,
NBWECIT1      NBWE + CITRUS YIELDS INCREASED BY 1% NATIONALLY,
NBWECIT3      NBWE + CITRUS YIELDS INCREASED BY 3% NATIONALLY,
NBWECIT5      NBWE + CITRUS YIELDS INCREASED BY 5% NATIONALLY,
NBWEJOINT1    NBWE + TOMATO POTATO CITRUS YIELDS INCREASED BY 1%,
NBWEJOINT3    NBWE + TOMATO POTATO CITRUS YIELDS INCREASED BY 3%,
NBWEJOINT5    NBWE + TOMATO POTATO CITRUS YIELDS INCREASED BY 5% /;
```

In the following section we will refer to the new scenarios also as the combined scenarios for they combine technological adjustment due to boll weevil eradication and post-harvest loss reduction.

Next, we carry over all scenarios of current interest into RUN. Here, we choose to carry over the combined scenarios NBWEBASE, NBWETOM1, and NBWEPOT1. In addition, we remove CIT1 and JOINT1 from the set of active scenarios. There is no need to delete the old set of active scenarios. It is sufficient to place an asterisk in column 1 of line 39 in Table 4.2.

```
SET RUN(ALLRUNS)     NAMES SET OF ALL RUNS THAT WILL BE DONE
*/BASE, TOM1, POT1, CIT1, JOINT1 /;
*/BASE, TOM1, TOM3, TOM5 /;
*/BASE, POT1, POT3, POT5 /;
*/BASE, CIT1, CIT3, CIT5 /;
*/BASE, JOINT1, JOINT3, JOINT5 /;
*/BASE, TOM1, TOM3, TOM5 , POT1, POT3, POT5, JOINT1, JOINT3, JOINT5 /;
 /BASE, TOM1, POT1, NBWEBASE, NBWETOM1, NBWEPOT1/;
```

Finally, we have to specify the technological adjustment associated with all newly integrated scenarios in ALLRUNS. Even though there are several alternative ways to complete this task, one would prefer a simple procedure. We already have formulated the adjustments for the old boll weevil eradication and post-harvest loss reduction scenarios. For all combined scenarios, we need to make an adjustment for both national boll weevil eradication and the particular post-harvest loss reduction assumption. First, we include all new scenarios that involve post-harvest loss reduction assumptions in Table PHLADJ. One only needs to duplicate lines 56-67 in Table 4.2 and change the scenario names. The semicolon has to be moved to the new table end. The new table is shown below.

```
 TABLE PHLADJ(ALLRUNS,CROP)   POST-HARVEST LOSS REDUCTION ASSUMPTIONS
              TOMATOFRSH  POTATOES  ORANGEFRSH  GRPFRTFRSH
 TOM1               1
 TOM3               3
 TOM5               5
 POT1                          1
 POT3                          3
 POT5                          5
 CIT1                                      1        1
 CIT3                                      3        3
 CIT5                                      5        5
 JOINT1             1          1           1        1
 JOINT3             3          3           3        3
 JOINT5             5          5           5        5
 NBWETOM1           1
 NBWETOM3           3
 NBWETOM5           5
 NBWEPOT1                      1
 NBWEPOT3                      3
 NBWEPOT5                      5
 NBWECIT1                                  1        1
 NBWECIT3                                  3        3
 NBWECIT5                                  5        5
 NBWEJOINT1         1          1           1        1
 NBWEJOINT3         3          3           3        3
 NBWEJOINT5         5          5           5        5
 ;
```

Second we need to incorporate the technological adjustments for national boll weevil eradication in all combined scenarios. Note that the BWE adjustment is the same across all combined scenarios. We create a new set NEWRUNS which includes all new scenarios that combine BWE and PHL.

```
SET NEWRUNS(ALLRUNS)           RUNS THAT COMBINE NATIONAL BWE AND PHL
/ NBWEBASE, BWETOM1, NBWETOM3, NBWETOM5, NBWEPOT1, NBWEPOT3, NBWEPOT5,
  NBWECIT1, NBWECIT3, NBWECIT5, NBWEJOINT1, NBWEJOINT3, NBWEJOINT5    /;
```

Next, we create a parameter COMBINE(ALLRUNS) that has one dimension for scenarios. Our objective is to use this parameter as an indicator of combined scenarios, i.e. we want this parameter to take on a value of one for all combined scenarios and a value of zero otherwise. Below, the appropriate GAMS code is shown.

```
PARAMETER COMBINE(ALLRUNS)     INDICATOR OF COMBINED SCENARIOS;
COMBINE(ALLRUNS)    = 0;
COMBINE(NEWRUNS)    = 1;
```

Once we have setup our switch we can finish the BWE adjustment for all combined scenarios. The dollar condition limits the adjustment to combined scenarios only.

```
 ADJUST(ALLI,SUBREG,”COTTON”,RUN)
 $(COMBINE(RUN) AND BWEADJ(SUBREG,”BELTWIDE”,ALLI))
  = 0.01 * BWEADJ(SUBREG,”BELTWIDE”,ALLI);
```

4.3.5. Use of SKIPINPUT

The execution of PESTLOOP can be time consuming, especially when there is a large number of alternative scenarios to be run. One of the time demanding procedures in ASM is the recalculation of all input costs for each scenario (input prices times input quantities plus fixed costs for each subregion, crop or animal type, and technology). In some studies as in the case of the post-harvest loss reduction study, we do not consider a change in inputs. Hence, there is no need to carry out the input calculation again and again for each alternative run. A simple way to turn off the input recalculation is to change the value of SKIPINPUT from 0 to 1. SKIPINPUT is a scalar that is defined and initialized to zero in ASMCALSU (as shown below).

```
  SCALAR SKIPINPUT  SKIP INPUT RECALCULATION /0/;
```

SKIPINPUT then is used in ASMCALRN to suppress the input recalculation whenever its value is different from zero. The GAMS statement in ASMCALRN is copied below.

```
  IF(SKIPINPUT EQ 0,
     COSTC(SUBREG,CROP,WTECH,CTECH,TECH)
      $ CBUDDATA("CROPLAND",SUBREG,CROP,WTECH,CTECH,TECH)=
           SUM(INPUT,INPUTPRICE(INPUT)*
                    CBUDDATA(INPUT,SUBREG,CROP,WTECH,CTECH,TECH))
         + SUM(COST,CBUDDATA(COST,SUBREG,CROP,WTECH,CTECH,TECH));
     COSTL(SUBREG,ANIMAL,LIVETECH)=
          SUM(INPUT,INPUTPRICE(INPUT)*
                    LBUDDATA(INPUT,SUBREG,ANIMAL,LIVETECH))
         + SUM(COST,LBUDDATA(COST,SUBREG,ANIMAL,LIVETECH));
     );
```

To suppress the input recalculation for alternative runs, one simply places a statement at the beginning of PESTLOOP which sets SKIPINPUT equal to 1 (Table 4.2, Line 4). Note that this statement must be deactivated for runs that involve boll weevil eradication assumptions since boll weevil eradication alters the chemical inputs.

Table 4.1 Batch Files

Panel A Two cases of the DOS batch file

Case I: A complete ASM run

```
command /c GAMS   ALLOFIT                                   s=.\t\f1
command /c GAMS   ASMMODEL                      r=.\t\f1    s=.\t\f2
command /c GAMS   ASMSOLVE                      r=.\t\f2    s=.\t\f3
```

Case II: An ASM Alternative runs batch file starting from a stored solution

```
rem   command /c GAMS   ALLOFIT                             s=.\t\f1
rem   command /c GAMS   ASMMODEL                r=.\t\f1    s=.\t\f2
rem   command /c GAMS   ASMSOLVE                r=.\t\f2    s=.\t\f3
command /c GAMS         PESTLOOP                r=.\t\f3    s=.\t\f4
command /c GAMS         RUNREPORT               r=.\t\f4
```

Panel B Two cases of the UNIX batch file

Case III A complete ASM run

```
gams        ALLOFIT                                         s=./t/f1
gams        ASMMODEL                            r=./t/f2    s=./t/f2
gams        ASMSOLVE                            r=./t/f3    s=./t/f3
gams        PESTLOOP                            r=./t/f4    s=./t/f4
gams        RUNREPORT                           r=./t/f5    s=./t/f5
```

Case IV An ASM Alternative runs batch file starting from a stored solution

```
#gams       ALLOFIT                                         s=./t/f1
#gams       ASMMODEL                            r=./t/f2    s=./t/f2
#gams       ASMSOLVE                            r=./t/f3    s=./t/f3
gams        PESTLOOP                            r=./t/f4    s=./t/f4
gams        RUNREPORT                           r=./t/f5    s=./t/f5
```

Table 4.2 Listing of Post-Harvest Loss Reduction and Boll Weevil Eradication Version of PESTLOOP

```
OPTION LP=CPLEX;
OPTION PROFILETOL=5;
OPTION SOLVEOPT=REPLACE;
SKIPINPUT =1;

*
* Step 1
* Define scenario sets for all technological changes
*

SET ALLRUNS  NAMES SET OF ALL SCENARIO RUNS THAT COULD BE DONE
/BASE        NO CHANGE,
 TOM1         TOMATO YIELDS INCREASED BY 1% NATIONALLY,
 TOM3         TOMATO YIELDS INCREASED BY 3% NATIONALLY,
 TOM5         TOMATO YIELDS INCREASED BY 5% NATIONALLY,
 POT1         POTATO YIELDS INCREASED BY 1% NATIONALLY,
 POT3         POTATO YIELDS INCREASED BY 3% NATIONALLY,
 POT5         POTATO YIELDS INCREASED BY 5% NATIONALLY,
 CIT1         CITRUS YIELDS INCREASED BY 1% NATIONALLY,
 CIT3         CITRUS YIELDS INCREASED BY 3% NATIONALLY,
 CIT5         CITRUS YIELDS INCREASED BY 5% NATIONALLY,
 JOINT1 TOMATO POTATO CITRUS YIELDS INCREASED BY 1% NATIONALLY,
 JOINT3 TOMATO POTATO CITRUS YIELDS INCREASED BY 3% NATIONALLY,
 JOINT5 TOMATO POTATO CITRUS YIELDS INCREASED BY 5% NATIONALLY,
 HIGHPLAINS  BOLL WEEVIL ERADICATION IN THE TEXAS HIGH PLAINS,
 BELTWIDE    BELTWIDE BOLL WEEVIL ERADICATION  /;

*
* Step 2
* Define scenario subset
*

SET RUN(ALLRUNS)     NAMES SET OF ALL RUNS THAT WILL BE DONE
*/BASE, TOM1, POT1, CIT1, JOINT1 /;
*/BASE, TOM1, TOM3, TOM5 /;
*/BASE, POT1, POT3, POT5 /;
*/BASE, CIT1, CIT3, CIT5 /;
*/BASE, JOINT1, JOINT3, JOINT5 /;
/BASE, TOM1, TOM3, TOM5 , POT1, POT3, POT5, JOINT1, JOINT3, JOINT5 /;

*
* Step 3
* Enter data
*

```

```
46 PARAMETER ADJUST(ALLI,SUBREG,CROP,RUN)   ADJUSTMENT FACTORS FOR EACH RUN;
47 ADJUST(ALLI,SUBREG,CROP,RUN)=0;
48
49 *The following table contains the assumed percentage increase in
50 *tomato, potato, orange, and grapefruit yields for each scenario run
51 *on post-harvest loss reduction (PHL)
52 *Remove asterisk in lines 71 and 75 to disregard PHL technology
53
54 TABLE PHLADJ(ALLRUNS,CROP)   POST-HARVEST LOSS REDUCTION ASSUMPTIONS
55            TOMATOFRSH  POTATOES  ORANGEFRSH  GRPFRTFRSH
56 TOM1           1
57 TOM3           3
58 TOM5           5
59 POT1                      1
60 POT3                      3
61 POT5                      5
62 CIT1                                  1        1
63 CIT3                                  3        3
64 CIT5                                  5        5
65 JOINT1         1          1           1        1
66 JOINT3         3          3           3        3
67 JOINT5         5          5           5        5
68 ;
69
70 *Update ADJUST parameter with data on post-harvest loss reduction
71 *$ONTEXT
72 ADJUST(CROP,SUBREG,CROP,RUN)
73    $PHLADJ(RUN,CROP)
74    =0.01 * PHLADJ(RUN,CROP)  ;
75 *$OFFTEXT
76
77
78 *The following table contains the percentage change in cotton yields and
79 *chemical costs for each subregion and boll weevil eradication (BWE)
80 *scenario.
81 *Remove asterisk in lines 104 and 108 to disregard BWE technology
82
83 TABLE  BWEADJ(SUBREG,ALLRUNS,ALLI)   BOLL WEEVIL ERADICATION ASSUMPTIONS
84                        COTTON     CHEMICALCO
85 ALABAMA   .BELTWIDE     1.95        - 19.1
86 ARKANSAS  .BELTWIDE     1.23        - 17.4
87 LOUISIANA .BELTWIDE     3.51        - 14.7
88 MISSOURI  .BELTWIDE     2.09        - 23.4
89 MISSISSIPP.BELTWIDE     2.28        -  8.7
90 OKLAHOMA  .BELTWIDE     2.51        - 27.4
91 TENNESSEE .BELTWIDE     6.12        - 48.9
92 TXHIPLAINS.BELTWIDE     0.25        -  1.1
93 TXROLINGPL.BELTWIDE     3.47        - 23.4
94 TXCNTBLACK.BELTWIDE     4.24        - 15.0
95 TXCOASTBE .BELTWIDE     4.29        - 29.7
```

```
TXTRANSPEC.BELTWIDE     1.56        - 30.4
TXSOUTH   .BELTWIDE     3.29        - 38.4
TXEAST    .BELTWIDE     4.35        - 35.5
TXEDPLAT  .BELTWIDE     5.89        - 44.2
TXHIPLAINS.HIGHPLAINS   0.25        -  1.1
;

*Update ADJUST parameter with data on boll weevil eradication
*$ONTEXT
ADJUST(ALLI,SUBREG,"COTTON",RUN)
  $BWEADJ(SUBREG,RUN,ALLI)
  =0.01 * BWEADJ(SUBREG,RUN,ALLI);
*$OFFTEXT

*
* Step 4
* Define sets and parameters for comparative report writing
*

SETS

SURITEM          ITEMS FOR THE OVERALL WELFARE COMPARISON TABLE
/CONSURPLUS, PROSURPLUS, FORSURPLUS, GOVCOST, TOTDOMSURP, TOTSURPLUS,
 NETSURPLUS/

REPITEM          REPORT ITEMS
/CS+PS, CS, PS, FOREIGN, GOV-COST, NET-DOM, TOT-SO, TOT-NETSO/

INPITEM          REGIONAL INPUT USE COMPARISON ITEMS
/W3-8LAND,SVEILAND,MDEILAND,LOEILAND,CROPLAND, SET-ASIDE,CONVERTED,
 TOTCROPLAN,PASTURE,FALLOW,AUMS,WATER,FIXED,PUMPED,LABOR,DIVERTLAND,
 HARVESTIRR, HARVESTDRY, HARVESTTOT/

REGITEM    REGIONAL INPUT USE COMPARISON ITEMS
/W3-8LAND,SVEILAND,MDEILAND,LOEILAND,CROPLAND,SET-ASIDE, CONVERTED,
 TOTCROPLAN, PASTURE, WATER, FIXED, PUMPED, LABOR, AUMS, PRODSUR, CONSUR,
 TOTSURP, DIVERTLAND, HARVESTIRR, HARVESTDRY, HARVESTTOT/

TEXAS(SUBREG)   SUBREGIONS WITHIN TEXAS
/TXHIPLAINS, TXROLINGPL, TXCNTBLACK, TXEAST,
 TXEDPLAT, TXCOASTBE, TXSOUTH, TXTRANSPEC/ ;



PARAMETERS
SURCOMP    (SURITEM,RUN)             WELFARE COMPARISON IN BILLION $
EXPORTCOMP (ALLI,RUN)                EXPORT COMPARISON
INPCOMP    (INPITEM,RUN)             RESOURCE USE RESULTS COMPARISON
NATINPCOMP (INPUT,RUN)               NATIONAL INPUT USE COMPARISON
PRICECOMP  (ALLI,RUN)                COMMODITY PRICE COMPARISON
```

```
PRODNCOMP  (ALLI,RUN)                COMMODITY PRODUCTION COMPARISON
REVCOMP    (ALLI,RUN)                GROSS REVENUE COMPARISON
REGCOMP    (REGITEM,ALLSUBREG,RUN)   REGIONL INPUT AND WELFARE COMPARISON
ACRECOMP   (CROP,ALLSUBREG,RUN)      REGIONAL HARVESTED ACRE COMPARISON
CROPPRCOMP (CROP,ALLSUBREG,RUN)      REGIONAL CROP PRODUCTION COMPARISON
LIVECOMP   (LIVESTOCK,ALLSUBREG,RUN) REGIONAL LIVESTOCK PRODUCTION
CBALANCEP  (PRIMARY,BALITEM,RUN)     PRODUCTION BALANCE SHEET
CBALANCES  (SECONDARY,BALITEM,RUN)   SECONDARY PRODUCTS BALANCE SHEET
CPCONSUR   (ALLI,ITEMCS,RUN)         DETAIL ON DOMESTIC CONSUMER SURPLUS
CFWELFARE  (ALLI,ITEMFOR,RUN)        DETAIL ON FOREIGN SURPLUS
IMPORTCOMP (ALLI,RUN)                QUANTITY IMPORTED
INPVALCOMP (ALLI,ALLSUBREG,RUN)      AVERAGE VARIABLE FACTOR PRICES
DRYCOMP    (*,*,RUN)                 DRYLAND CROP ACREAGE
IRRCOMP    (*,*,RUN)                 IRRIGATED CROP ACREAGE
DRYREGCOMP (ALLSUBREG,*,RUN)         DRYLAND REGIONAL ACREAGE
IRRREGCOMP (ALLSUBREG,*,RUN)         IRRIGATED REGIONAL ACREAGE
GOVCOMP    (*,ITEMDEF,RUN)           GOVERNMENT PAYMENT COMPARISON
PCONSCOMP  (*,ITEMCS,RUN)            CONSUMER SURPLUS
FCONSCOMP  (*,RUN)                   FOREIGN SURPLUS
EROSIONCP  (*,RUN)                   NATIONAL EROSION
OTAEXPORT  (RUN)                     NATIONAL EXPORT VALUE IN MIL $
REROSION   (RUN,*,ALLSUBREG)         REGIONAL EROSION
RNATINPUT  (RUN,INPUT, ALLSUBREG)    REGIONAL INPUT USE RESULT COMPARISON
OTAEXPORT  (RUN)                     NATIONAL EXPORT VALUE IN MIL $
TXREPORT   (*)                       TEXAS INPUT USE AND PRODUCER SURPLUS
IACR  (*,CROP,WTECH,CTECH,TECH,RUN)  SUBREGIONAL CROP ACREAGE
IACRSUM    (*,WTECH,CTECH,TECH,RUN)  CROP ACREAGE SUMMED OVER CROPS
IDRYACRSUM (*,CTECH,RUN)             DRYLAND ACREAGE
ITOTALDRY  (*,RUN)                   TOTAL SUBREGIONAL DRYLAND ACREAGE
IIRRIGSUM  (*,CTECH,TECH,RUN)        SUBREGIONAL IRRIGATED ACREAGE
ITOTALIRRI (*,RUN)                   TOTAL SUBREGIONAL IRRIGATED ACREAGE
IWATERUSE  (*,RUN)                   WATER USE BY SUBREGION
ITXRPT     (REPITEM,RUN)             TEXAS WELFARE REPORT
IWELREPORT (REPITEM,RUN)             SUBREGIONAL WELFARE IRRIGATED LAND
IREGWEL    (ALLSUBREG,REPITEM,RUN)   NATIONAL WELFARE IRRIGATED LAND;

*
* Step 5
* Save original parameters
*

PARAMETERS
SCBUDDATA(ALLI,SUBREG,CROP,WTECH,CTECH,TECH)    SAVED CROP BUDGET DATA,
SFARMPROD(*,CROP)                               SAVED FARM PROGRAM DATA ;

*Save original farm program and crop budget data
SFARMPROD("TARGET",CROP)      = FARMPROD("TARGET",CROP);
SFARMPROD("DEFIC",CROP)       = FARMPROD("DEFIC",CROP);
SFARMPROD("LOANRATE",CROP)    = FARMPROD("LOANRATE",CROP);
SFARMPROD("MKTLOANY-N",CROP)  = FARMPROD("MKTLOANY-N",CROP);
```

```

SCBUDDATA(ALLI,SUBREG,CROP,WTECH,CTECH,TECH)
                  =CBUDDATA(ALLI,SUBREG,CROP,WTECH,CTECH,TECH);

*
* Set farm program to zero for all runs
*

FARMPROD("TARGET",CROP)
  $(FARMPROD("TARGET",CROP)) = FARMPROD("TARGET",CROP)  * 0.0000001 ;
FARMPROD("DEFIC",CROP)
  $(FARMPROD("DEFIC",CROP))  = FARMPROD("DEFIC",CROP)   * 0.0000001 ;
FARMPROD("LOANRATE",CROP)    = FARMPROD("LOANRATE",CROP)* 0.0000001 ;
FARMPROD("MKTLOANY-N",CROP)  = 0                            ;

*
* Step 6
* Loop over all activated scenarios
*

LOOP(RUN,
*
* Step 7
* Adjustments according to scenario assumption
*
CBUDDATA(ALLI,SUBREG,CROP,WTECH,CTECH,TECH)
      = SCBUDDATA(ALLI,SUBREG,CROP,WTECH,CTECH,TECH) ;

CBUDDATA(ALLI,SUBREG,CROP,WTECH,CTECH,TECH)
  $((CBUDDATA(ALLI,SUBREG,CROP,WTECH,CTECH,TECH)) and
    (ADJUST(ALLI,SUBREG,CROP,RUN)))
     = (SCBUDDATA(ALLI,SUBREG,CROP,WTECH,CTECH,TECH)*
          (1+ ADJUST(ALLI,SUBREG,CROP,RUN)));
*
* Steps 8
* Solve model, solve comparative reports
*
$INCLUDE asmcalrn
$INCLUDE asmsolve
*
* Steps 9
* Solve comparative reports
*
$INCLUDE asmrept
$INCLUDE asmcompr
);
```

Table 4.3 Listing of RUNREPORT

```
*
*
* Declare parameters
*
*
PARAMETERS
SURPLUS  (*,RUN)                NATIONAL TOTAL SURPLUS   IN MILLION
SURABS   (*,RUN)                ABSOLUTE CHANGE WRT BASE IN MILLION
SURPERCHN(*,RUN)                PERCENTAGE CHANGE WRT BASE
PRIMARYCS(PRIMARY,RUN)          PRIMARY C.S.
PRICSDIF (PRIMARY,RUN)          PRIMARY C.S. DIFFERENCE WITH BASE

REGSURPLUS(RUN,REGIONS,SURITEM) REGIONAL SURPLUS IN MILLION
REGSURABS (RUN,REGIONS,SURITEM) REGIONAL SURPLUS ABSOLUTE CHANGE
REGSURPER (RUN,REGIONS,SURITEM) REGIONAL SURPLUS PERCENT CHANGE
REGSURPER1(RUN,REGIONS,SURITEM) PERCENTAGE CHANGE IS HIGHER THAN 0.1

SRGSURPLUS(SUBREG,RUN,SURITEM)  SUB-REGIONAL SURPLUS IN MILLION
SRGSURABS (SUBREG,RUN,SURITEM)  SUB-REGIONAL SURPLUS ABSOLUTE CHANGE
SRGSURPER (SUBREG,RUN,SURITEM)  SUB-REGIONAL SURPLUS PERCENT CHANGE
SRGSURPER1(SUBREG,RUN,SURITEM)  PERCENTAGE CHANGE IS HIGHER THAN 0.1

FPPARTICIP(PRIMARY,RUN)         FARM PROGRAM PARTICIPATION
DEFICPRODN(CROP,RUN)            PRODUCTION WITH DEFICIENCY PAYMENT

CROPMIXR  (SUBREG,CROP,RUN)     SUBREGIONAL TOTAL HARVEST ACRE IN 1000
CROPMIXABS(SUBREG,CROP,RUN)     ABSOLUTE   CHANGE WRT BASE
CROPMIXPCN(SUBREG,CROP,RUN)     PERCENTAGE CHANGE WRT BASE
CROPMIXPC2(SUBREG,CROP,RUN)     PERCENTAGE CHANGE IS HIGHER THAN 2

NATIONLAND(CROP,RUN)            NATIONAL TOTAL HARVEST ACRE IN 1000
NATLANDABS(CROP,RUN)            NATIONAL LAND ABSOLUTE   CHANGE WRT BASE
NATLANDPER(CROP,RUN)            NATIONAL LAND PERCENTAGE CHANGE WRT BASE
NATLANDP2 (CROP,RUN)            PERCENTAGE CHANGE IS HIGHER THAN 2

IRRMIXR  (SUBREG,CROP,RUN)      SUBREGIONAL IRRIGATION ACRE IN 1000
IRRMIXABS(SUBREG,CROP,RUN)      ABSOLUTE   CHANGE WRT BASE
IRRMIXPCN(SUBREG,CROP,RUN)      PERCENTAGE CHANGE WRT BASE
IRRMIXPC2(SUBREG,CROP,RUN)      PERCENTAGE CHANGE IS HIGHER THAN 2

DRYMIXR  (SUBREG,CROP,RUN)      SUBREGIONAL DRYLAND ACRE IN 1000 ACRES
DRYMIXABS(SUBREG,CROP,RUN)      ABSOLUTE   CHANGE WRT BASE
DRYMIXPCN(SUBREG,CROP,RUN)      PERCENTAGE CHANGE WRT BASE
DRYMIXPC2(SUBREG,CROP,RUN)      PERCENTAGE CHANGE IS HIGHER THAN 2

BALANCE(RUN,PRIMARY,BALITEM)    PRIMARY PRODUCT DEMAND AND SUPPLY BALANCE

PRICE    (PRIMARY,RUN)          MAJOR CROP PRICE IN DOLLARS
PRICEDIF (PRIMARY,RUN)          PRICE CHANGE FOR MAJOR CROPS IN DOLLARS
PRICEPER (PRIMARY,RUN)          PERCENTAGE PRICE CHANGE FOR MAJOR CROPS
```

```
PRICEPC1 (PRIMARY,RUN)          PERCENTAGE PRICE CHANGE HIGHER THAN 1

PRODUCTION(CROP,RUN)            NATIONAL PRODUCTION IN 1000 UNITS
PRODUCTDIF(CROP,RUN)            ABSOLUTE   CHANGE WRT BASE
PRODUCTPER(CROP,RUN)            PERCENTAGE CHANGE WRT BASE
PRODUCTPC1(CROP,RUN)            PERCENTAGE CHANGE IS HIGHER THAN 1

LIVEPROD(SUBREG,LIVESTOCK,RUN)  LIVESTOCK PRODUCTION FOR EACH SUBREGION
LIVEABS (SUBREG,LIVESTOCK,RUN)  ABSOLUTE   CHANGE WRT BASE
LIVEPCNT(SUBREG,LIVESTOCK,RUN)  PERCENTAGE CHANGE WRT BASE
LIVEPCN2(SUBREG,LIVESTOCK,RUN)  PERCENTAGE CHANGE IS HIGHER THAN 2
;
*
*
* Calculations
*
*
* Get surplus measures from comparative report
SURPLUS("TOTSURPLUS",RUN) = SURCOMP("TOTSURPLUS",RUN)*1000;
SURPLUS("CONSURPLUS",RUN) = SURCOMP("CONSURPLUS",RUN)*1000;
SURPLUS("PROSURPLUS",RUN) = SURCOMP("PROSURPLUS",RUN)*1000;
SURPLUS("FORSURPLUS",RUN) = SURCOMP("FORSURPLUS",RUN)*1000;

* Compute absolute changes in surplus measures
SURABS("TOTSURPLUS",RUN)$SURPLUS("TOTSURPLUS",RUN)
= SURPLUS("TOTSURPLUS",RUN) - SURPLUS("TOTSURPLUS","BASE");
SURABS("CONSURPLUS",RUN)$SURPLUS("CONSURPLUS",RUN)
= SURPLUS("CONSURPLUS",RUN) - SURPLUS("CONSURPLUS","BASE");
SURABS("PROSURPLUS",RUN)$SURPLUS("PROSURPLUS",RUN)
= SURPLUS("PROSURPLUS",RUN) - SURPLUS("PROSURPLUS","BASE");
SURABS("FORSURPLUS",RUN)$SURPLUS("FORSURPLUS",RUN)
= SURPLUS("FORSURPLUS",RUN) - SURPLUS("FORSURPLUS","BASE");

* Compute percentage changes in surplus measures
SURPERCHN("TOTSURPLUS",RUN)$SURPLUS("TOTSURPLUS",RUN)
= SURABS("TOTSURPLUS",RUN) / SURPLUS("TOTSURPLUS","BASE")*100;
SURPERCHN("CONSURPLUS",RUN)$SURPLUS("CONSURPLUS",RUN)
= SURABS("CONSURPLUS",RUN) / SURPLUS("CONSURPLUS","BASE")*100;
SURPERCHN("PROSURPLUS",RUN)$SURPLUS("PROSURPLUS",RUN)
= SURABS("PROSURPLUS",RUN) / SURPLUS("PROSURPLUS","BASE")*100;
SURPERCHN("FORSURPLUS",RUN)$SURPLUS("FORSURPLUS",RUN)
= SURABS("FORSURPLUS",RUN) / SURPLUS("FORSURPLUS","BASE")*100;

```

```

* Get sub-regional surplus measures from comparative report
SRGSURPLUS(SUBREG,RUN,"PROSURPLUS")
=REGCOMP("PRODSUR",SUBREG,RUN)/1000;
SRGSURPLUS(SUBREG,RUN,"CONSURPLUS")
=REGCOMP("CONSUR",SUBREG,RUN)/1000;
SRGSURPLUS(SUBREG,RUN,"TOTSURPLUS")
=REGCOMP("TOTSURP",SUBREG,RUN)/1000;

* Compute absolute changes in sub-regional surplus measures
SRGSURABS(SUBREG,RUN,"PROSURPLUS")
  $SRGSURPLUS(SUBREG,RUN,"PROSURPLUS")
     = SRGSURPLUS(SUBREG,RUN,"PROSURPLUS")
      -SRGSURPLUS(SUBREG,"BASE","PROSURPLUS");
SRGSURABS(SUBREG,RUN,"CONSURPLUS")
  $SRGSURPLUS(SUBREG,RUN,"CONSURPLUS")
     = SRGSURPLUS(SUBREG,RUN,"CONSURPLUS")
      -SRGSURPLUS(SUBREG,"BASE","CONSURPLUS");
SRGSURABS(SUBREG,RUN,"TOTSURPLUS")
  $SRGSURPLUS(SUBREG,RUN,"TOTSURPLUS")
     = SRGSURPLUS(SUBREG,RUN,"TOTSURPLUS")
      -SRGSURPLUS(SUBREG,"BASE","TOTSURPLUS");

* Compute percentage changes in sub-regional surplus measures
SRGSURPER(SUBREG,RUN,"PROSURPLUS")
  $SRGSURPLUS(SUBREG,RUN,"PROSURPLUS")
    = 100 * SRGSURABS(SUBREG,RUN,"PROSURPLUS")/
        SRGSURPLUS(SUBREG,"BASE","PROSURPLUS") ;
SRGSURPER(SUBREG,RUN,"CONSURPLUS")
  $SRGSURPLUS(SUBREG,RUN,"CONSURPLUS")
    = 100 * SRGSURABS(SUBREG,RUN,"CONSURPLUS")/
        SRGSURPLUS(SUBREG,"BASE","CONSURPLUS") ;
SRGSURPER(SUBREG,RUN,"TOTSURPLUS")
  $SRGSURPLUS(SUBREG,RUN,"TOTSURPLUS")
    = 100 * SRGSURABS(SUBREG,RUN,"TOTSURPLUS")/
        SRGSURPLUS(SUBREG,"BASE","TOTSURPLUS") ;

* Compute regional surplus measures
REGSURPLUS(RUN,REGIONS,"PROSURPLUS")
 =SUM(MAPPING(REGIONS,SUBREG),SRGSURPLUS(SUBREG,RUN,"PROSURPLUS"));
REGSURPLUS(RUN,REGIONS,"CONSURPLUS")
 =SUM(MAPPING(REGIONS,SUBREG),SRGSURPLUS(SUBREG,RUN,"CONSURPLUS"));
REGSURPLUS(RUN,REGIONS,"TOTSURPLUS")
 =SUM(MAPPING(REGIONS,SUBREG),SRGSURPLUS(SUBREG,RUN,"TOTSURPLUS"));


```

```
* Compute absolute changes in regional surplus measures
REGSURABS(RUN,REGIONS,"PROSURPLUS")
  $REGSURPLUS(RUN,REGIONS,"PROSURPLUS")
     = REGSURPLUS(RUN,REGIONS,"PROSURPLUS")
      -REGSURPLUS("BASE",REGIONS,"PROSURPLUS");
REGSURABS(RUN,REGIONS,"CONSURPLUS")
  $REGSURPLUS(RUN,REGIONS,"CONSURPLUS")
     = REGSURPLUS(RUN,REGIONS,"CONSURPLUS")
      -REGSURPLUS("BASE",REGIONS,"CONSURPLUS");
REGSURABS(RUN,REGIONS,"TOTSURPLUS")
  $REGSURPLUS(RUN,REGIONS,"TOTSURPLUS")
     = REGSURPLUS(RUN,REGIONS,"TOTSURPLUS")
      -REGSURPLUS("BASE",REGIONS,"TOTSURPLUS");

* Compute percentage changes in regional surplus measures
REGSURPER(RUN,REGIONS,"PROSURPLUS")
  $REGSURPLUS(RUN,REGIONS,"PROSURPLUS")
    = 100 * REGSURABS(RUN,REGIONS,"PROSURPLUS")/
        REGSURPLUS("BASE",REGIONS,"PROSURPLUS") ;
REGSURPER(RUN,REGIONS,"CONSURPLUS")
  $REGSURPLUS(RUN,REGIONS,"CONSURPLUS")
    = 100 * REGSURABS(RUN,REGIONS,"CONSURPLUS")/
        REGSURPLUS("BASE",REGIONS,"CONSURPLUS") ;
REGSURPER(RUN,REGIONS,"TOTSURPLUS")
  $REGSURPLUS(RUN,REGIONS,"TOTSURPLUS")
    = 100 * REGSURABS(RUN,REGIONS,"TOTSURPLUS")/
        REGSURPLUS("BASE",REGIONS,"TOTSURPLUS") ;

* Truncate percentage change to only report values greater than 0.1%
SRGSURPER1(SUBREG,RUN,SURITEM)
   $(ABS(SRGSURPER(SUBREG,RUN,SURITEM)) GT 0.1)
       = SRGSURPER(SUBREG,RUN,SURITEM);
REGSURPER1(RUN,REGIONS,SURITEM)
   $(ABS(REGSURPER(RUN,REGIONS,SURITEM)) GT 0.1)
       = REGSURPER(RUN,REGIONS,SURITEM);

* Copy in more items from comparative report
PRIMARYCS(PRIMARY,RUN)
         =PCONSCOMP(PRIMARY,"DOMEST-CS",RUN);
PRICSDIF(PRIMARY,RUN)
         =PRIMARYCS(PRIMARY,RUN)-PRIMARYCS(PRIMARY,"BASE");
BALANCE(RUN,PRIMARY,BALITEM) = CBALANCEP(PRIMARY,BALITEM,RUN);

* Revert prices to dollars
PRICE   (CROP,RUN)   = BALANCE(RUN,CROP,"PRICEX100")/100;
* Compute absolute price difference to base scenario
PRICEDIF(CROP,RUN)   = PRICE(CROP,RUN) - PRICE(CROP,"BASE");

```

```
* Compute percentage price change to base scenario
PRICEPER(CROP,RUN)   = (PRICEDIF(CROP,RUN)/PRICE(CROP,"BASE"))*100
                       $ PRICE(CROP,"BASE");
* Truncate percentage change to only report values greater than 1%
PRICEPC1(CROP,RUN)   = PRICEPER(CROP,RUN)
                       $ (ABS(PRICEPER(CROP,RUN)) GT 1);

* Compute change, percentage change in production, truncate percentage
* to only report values greater than 1%
PRODUCTION(CROP,RUN) = BALANCE(RUN,CROP,"PRODUCTION");
PRODUCTDIF(CROP,RUN) = PRODUCTION(CROP,RUN)
                               - PRODUCTION(CROP,"BASE");
PRODUCTPER(CROP,RUN) =(PRODUCTDIF(CROP,RUN) /
                                 PRODUCTION(CROP,"BASE"))*100
                       $ PRODUCTION(CROP,"BASE");
PRODUCTPC1(CROP,RUN) = PRODUCTPER(CROP,RUN)
                       $ (ABS(PRODUCTPER(CROP,RUN)) GT 1);

* Compute crop mix, absolute change, percentage change with respect
* to the base scenario
CROPMIXR(SUBREG,CROP,RUN)
= SUM((WTECH,CTECH,TECH),IACR(SUBREG,CROP,WTECH,CTECH,TECH,RUN));
CROPMIXABS(SUBREG,CROP,RUN)$CROPMIXR(SUBREG,CROP,"BASE")
= CROPMIXR(SUBREG,CROP,RUN) - CROPMIXR(SUBREG,CROP,"BASE");
CROPMIXPCN(SUBREG,CROP,RUN)$CROPMIXR(SUBREG,CROP,"BASE")
= CROPMIXABS(SUBREG,CROP,RUN)/CROPMIXR(SUBREG,CROP,"BASE")*100;
* If crop acreage is zero for base secnario, percentage change is N.A.
CROPMIXABS(SUBREG,CROP,RUN) $(NOT CROPMIXR(SUBREG,CROP,"BASE")
   AND CROPMIXR(SUBREG,CROP,RUN) NE 0) = NA;
CROPMIXPCN(SUBREG,CROP,RUN) $(NOT CROPMIXR(SUBREG,CROP,"BASE")
   AND CROPMIXR(SUBREG,CROP,RUN) NE 0) = NA;
* truncate percentage change to only report values greater than 2%
CROPMIXPC2(SUBREG,CROP,RUN) $(ABS(CROPMIXPCN(SUBREG,CROP,RUN)) GT 2)
= CROPMIXPCN(SUBREG,CROP,RUN);

*Compute national harvested acreage, absolute change, percentage change
*with respect to the base scenario
*truncate percentage change to only report values greater than 2%
NATIONLAND(CROP,RUN)
= SUM(SUBREG,CROPMIXR(SUBREG,CROP,RUN));
NATLANDABS(CROP,RUN) $NATIONLAND(CROP,"BASE")
= NATIONLAND(CROP,RUN)- NATIONLAND(CROP,"BASE");
NATLANDPER(CROP,RUN) $NATIONLAND(CROP,"BASE")
= NATLANDABS(CROP,RUN)/NATIONLAND(CROP,"BASE")*100;
NATLANDP2(CROP,RUN)  $(ABS(NATLANDPER(CROP,RUN)) GT 2)
= NATLANDPER(CROP,RUN);

* Compute irrigated acreage, absolute change and percentage
* change between scenarios, set non-defined items to N.A.
* truncate percentage change to only report values greater than 2%
```

```
IRRMIXR(SUBREG,CROP,RUN)
= SUM((CTECH,TECH),IACR(SUBREG,CROP,"IRRIG",CTECH,TECH,RUN));
IRRMIXABS(SUBREG,CROP,RUN)$IRRMIXR(SUBREG,CROP,"BASE")
= IRRMIXR(SUBREG,CROP,RUN)-IRRMIXR(SUBREG,CROP,"BASE");
IRRMIXABS(SUBREG,CROP,RUN) = NA $( NOT IRRMIXR(SUBREG,CROP,"BASE")
   AND IRRMIXR(SUBREG,CROP,RUN) NE 0);
IRRMIXPCN(SUBREG,CROP,RUN)$IRRMIXR(SUBREG,CROP,"BASE")
=((IRRMIXR(SUBREG,CROP,RUN)- IRRMIXR(SUBREG,CROP,"BASE"))/
   IRRMIXR(SUBREG,CROP,"BASE")*100);
IRRMIXPCN(SUBREG,CROP,RUN) = NA $( NOT IRRMIXR(SUBREG,CROP,"BASE")
   AND IRRMIXR(SUBREG,CROP,RUN) NE 0);
IRRMIXPC2(SUBREG,CROP,RUN) $(ABS(IRRMIXPCN(SUBREG,CROP,RUN)) GT 2)
= IRRMIXPCN(SUBREG,CROP,RUN);

* Compute dryland acreage, absolute change and percentage
* change between scenarios, set non-defined items to N.A.
* truncate percentage change to only report values greater than 2%
DRYMIXR(SUBREG,CROP,RUN)
= SUM((CTECH,TECH), IACR(SUBREG,CROP,"DRYLAND",CTECH,TECH,RUN));
DRYMIXABS(SUBREG,CROP,RUN) $DRYMIXR(SUBREG,CROP,"BASE")
= DRYMIXR(SUBREG,CROP,RUN)-DRYMIXR(SUBREG,CROP,"BASE");
DRYMIXABS(SUBREG,CROP,RUN) = NA $( NOT DRYMIXR(SUBREG,CROP,"BASE")
   AND DRYMIXR(SUBREG,CROP,RUN) NE 0);
DRYMIXPCN(SUBREG,CROP,RUN)$DRYMIXR(SUBREG,CROP,"BASE")
=((DRYMIXR(SUBREG,CROP,RUN)- DRYMIXR(SUBREG,CROP,"BASE"))/
   DRYMIXR(SUBREG,CROP,"BASE")*100);
DRYMIXPCN(SUBREG,CROP,RUN) = NA $ ( NOT DRYMIXR(SUBREG,CROP,"BASE")
   AND DRYMIXR(SUBREG,CROP,RUN) NE 0);
DRYMIXPC2(SUBREG,CROP,RUN) $(ABS(DRYMIXPCN(SUBREG,CROP,RUN)) GT 2)
= DRYMIXPCN(SUBREG,CROP,RUN);
*
*
* Round report items to limit number of decimal places to be displayed
*
*
SURPLUS("CONSURPLUS",RUN) = ROUND(SURPLUS("CONSURPLUS",RUN),0);
SURPLUS("PROSURPLUS",RUN) = ROUND(SURPLUS("PROSURPLUS",RUN),0);
SURPLUS("FORSURPLUS",RUN) = ROUND(SURPLUS("FORSURPLUS",RUN),0);
SURPLUS("TOTSURPLUS",RUN) = ROUND(SURPLUS("TOTSURPLUS",RUN),0);

SURABS("CONSURPLUS",RUN)  = ROUND(SURABS("CONSURPLUS",RUN),1);
SURABS("PROSURPLUS",RUN)  = ROUND(SURABS("PROSURPLUS",RUN),1);
SURABS("FORSURPLUS",RUN)  = ROUND(SURABS("FORSURPLUS",RUN),1);
SURABS("TOTSURPLUS",RUN)  = ROUND(SURABS("TOTSURPLUS",RUN),1);
```

```
SURPERCHN("CONSURPLUS",RUN)  = ROUND(SURPERCHN("CONSURPLUS",RUN),2);
SURPERCHN("PROSURPLUS",RUN)  = ROUND(SURPERCHN("PROSURPLUS",RUN),2);
SURPERCHN("FORSURPLUS",RUN)  = ROUND(SURPERCHN("FORSURPLUS",RUN),2);
SURPERCHN("TOTSURPLUS",RUN)  = ROUND(SURPERCHN("TOTSURPLUS",RUN),2);

SRGSURABS(SUBREG,RUN,"PROSURPLUS")
  = ROUND(SRGSURABS(SUBREG,RUN,"PROSURPLUS"),1);
SRGSURABS(SUBREG,RUN,"CONSURPLUS")
  = ROUND(SRGSURABS(SUBREG,RUN,"CONSURPLUS"),1);
SRGSURABS(SUBREG,RUN,"TOTSURPLUS")
  = ROUND(SRGSURABS(SUBREG,RUN,"TOTSURPLUS"),1);

REGSURABS(RUN,REGIONS,"PROSURPLUS")
  = ROUND(REGSURABS(RUN,REGIONS,"PROSURPLUS"),1);
REGSURABS(RUN,REGIONS,"CONSURPLUS")
  = ROUND(REGSURABS(RUN,REGIONS,"CONSURPLUS"),1);
REGSURABS(RUN,REGIONS,"TOTSURPLUS")
  = ROUND(REGSURABS(RUN,REGIONS,"TOTSURPLUS"),1);

SRGSURPER(SUBREG,RUN,"PROSURPLUS")
  = ROUND(SRGSURPER(SUBREG,RUN,"PROSURPLUS"),2);
SRGSURPER(SUBREG,RUN,"CONSURPLUS")
  = ROUND(SRGSURPER(SUBREG,RUN,"CONSURPLUS"),2);
SRGSURPER(SUBREG,RUN,"TOTSURPLUS")
  = ROUND(SRGSURPER(SUBREG,RUN,"TOTSURPLUS"),2);

REGSURPER(RUN,REGIONS,"PROSURPLUS")
  = ROUND(REGSURPER(RUN,REGIONS,"PROSURPLUS"),2);
REGSURPER(RUN,REGIONS,"CONSURPLUS")
  = ROUND(REGSURPER(RUN,REGIONS,"CONSURPLUS"),2);
REGSURPER(RUN,REGIONS,"TOTSURPLUS")
  = ROUND(REGSURPER(RUN,REGIONS,"TOTSURPLUS"),2);

PRICE(CROP,RUN)             = ROUND(PRICE(CROP,RUN),2);
PRICEDIF(CROP,RUN)          = ROUND(PRICEDIF(CROP,RUN),2);
PRICEPER(CROP,RUN)          = ROUND(PRICEPER(CROP,RUN),2);
PRICEPC1(CROP,RUN)          = ROUND(PRICEPC1(CROP,RUN),2);
PRODUCTION(CROP,RUN)        = ROUND(PRODUCTION(CROP,RUN),0);
PRODUCTDIF(CROP,RUN)        = ROUND(PRODUCTDIF(CROP,RUN),0);
PRODUCTPER(CROP,RUN)        = ROUND(PRODUCTPER(CROP,RUN),2);
PRODUCTPC1(CROP,RUN)        = ROUND(PRODUCTPC1(CROP,RUN),2);
NATIONLAND(CROP,RUN)        = ROUND(NATIONLAND(CROP,RUN),0);
NATLANDABS(CROP,RUN)        = ROUND(NATLANDABS(CROP,RUN),0);
NATLANDPER(CROP,RUN)        = ROUND(NATLANDPER(CROP,RUN),2);
NATLANDP2(CROP,RUN)         = ROUND(NATLANDP2(CROP,RUN),2);
CROPMIXR(SUBREG,CROP,RUN)   = ROUND(CROPMIXR(SUBREG,CROP,RUN),0);
CROPMIXABS(SUBREG,CROP,RUN) = ROUND(CROPMIXABS(SUBREG,CROP,RUN),0);
CROPMIXPCN(SUBREG,CROP,RUN) = ROUND(CROPMIXPCN(SUBREG,CROP,RUN),2);
CROPMIXPC2(SUBREG,CROP,RUN) = ROUND(CROPMIXPC2(SUBREG,CROP,RUN),2);
IRRMIXR(SUBREG,CROP,RUN)    = ROUND(IRRMIXR(SUBREG,CROP,RUN),0);
```

```
IRRMIXABS(SUBREG,CROP,RUN)  = ROUND(IRRMIXABS(SUBREG,CROP,RUN),0);
IRRMIXPCN(SUBREG,CROP,RUN)  = ROUND(IRRMIXPCN(SUBREG,CROP,RUN),2);
DRYMIXR(SUBREG,CROP,RUN)    = ROUND(DRYMIXR(SUBREG,CROP,RUN),0);
DRYMIXABS(SUBREG,CROP,RUN)  = ROUND(DRYMIXABS(SUBREG,CROP,RUN),0);
DRYMIXPCN(SUBREG,CROP,RUN)  = ROUND(DRYMIXPCN(SUBREG,CROP,RUN),2);
*
*
* Setup decimal places and formats for displays
*
*
OPTION SURPLUS    :0:1:1;
OPTION SURABS     :1:1:1;
OPTION SURPERCHN  :2:1:1;
OPTION BALANCE    :0:1:1;

OPTION SRGSURPLUS :0:2:1;
OPTION SRGSURABS  :1:2:1;
OPTION SRGSURPER  :2:1:1;
OPTION SRGSURPER1 :2:1:1;

OPTION REGSURPLUS :0:1:1;
OPTION REGSURABS  :1:1:1;
OPTION REGSURPER  :2:1:1;
OPTION REGSURPER1 :2:1:1;

OPTION PRODUCTION :0:1:1;
OPTION PRODUCTDIF :0:1:1;

OPTION CROPMIXR   :2:2:1;
OPTION CROPMIXABS :2:2:1;
OPTION CROPMIXPCN :2:2:1;
OPTION CROPMIXPC2 :2:2:1;

OPTION NATIONLAND :0:1:1;
OPTION NATLANDABS :0:1:1;
OPTION NATLANDPER :2:1:1;
OPTION NATLANDP2  :2:1:1;

OPTION IRRMIXR    :2:2:1;
OPTION IRRMIXABS  :2:2:1;
OPTION IRRMIXPC2  :2:2:1;

OPTION DRYMIXR    :2:2:1;
OPTION DRYMIXABS  :2:2:1;
OPTION DRYMIXPC2  :2:2:1;

OPTION EROSIONCP  :0:1:1;
OPTION NATINPCOMP :0:1:1;
*
*
```

```
383  * Display items - activate / deactivate items
384  * Note: asterisk in column 1 suppresses print
385  *
386  *
387   DISPLAY SURPLUS;
388   DISPLAY SURABS;
389   DISPLAY SURPERCHN;
390
391  *DISPLAY REGSURPLUS;
392   DISPLAY REGSURABS;
393  *DISPLAY REGSURPER;
394   DISPLAY REGSURPER1;
395
396  *DISPLAY SRGSURPLUS;
397  *DISPLAY SRGSURABS;
398  *DISPLAY SRGSURPER;
399  *DISPLAY SRGSURPER1;
400
401  *DISPLAY CROPMIXR;
402  *DISPLAY CROPMIXABS;
403  *DISPLAY CROPMIXPCN;
404  *DISPLAY CROPMIXPC2;
405
406   DISPLAY PRICE;
407   DISPLAY PRICEDIF;
408  *DISPLAY PRICEPER;
409   DISPLAY PRICEPC1;
410
411   DISPLAY NATIONLAND;
412   DISPLAY NATLANDABS;
413  *DISPLAY NATLANDPER;
414   DISPLAY NATLANDP2 ;
415
416   DISPLAY PRODUCTION;
417   DISPLAY PRODUCTDIF;
418  *DISPLAY PRODUCTPER;
419   DISPLAY PRODUCTPC1;
420
421  *DISPLAY EROSIONCP;
422
423  *DISPLAY IRRMIXR;
424  *DISPLAY IRRMIXABS;
425  *DISPLAY IRRMIXPC2;
426  *DISPLAY DRYMIXR;
427  *DISPLAY DRYMIXABS;
428  *DISPLAY DRYMIXPC2;
```

Table 4.4 ASM Output For Boll Weevil Eradication Study

```
----  63023 PARAMETER SURPLUS      NATIONAL TOTAL SURPLUS IN MILLION

                  BASE  HIGHPLAINS    BELTWIDE

CONSURPLUS     1175235     1175235     1175311
PROSURPLUS       28668       28670       28702
FORSURPLUS      127174      127174      127210
TOTSURPLUS     1331077     1331079     1331222


----  63024 PARAMETER SURABS       ABSOLUTE CHANGE WRT BASE IN MILLION

            HIGHPLAINS    BELTWIDE

CONSURPLUS                    75.6
PROSURPLUS         1.6        33.6
FORSURPLUS                    36.1
TOTSURPLUS         1.6       145.3


----  63025 PARAMETER SURPERCHN    PERCENTAGE CHANGE WRT BASE

            HIGHPLAINS    BELTWIDE

CONSURPLUS                    0.01
PROSURPLUS        0.01        0.12
FORSURPLUS                    0.03
TOTSURPLUS                    0.01


----  63028 PARAMETER REGSURABS    REGIONAL SURPLUS ABSOLUTE CHANGE IN MILLION

INDEX 1 = HIGHPLAINS

            PROSURPLUS  TOTSURPLUS

SOUTHPLAIN         1.6         1.6
STHCENTRAL         1.6         1.6
```

```
INDEX 1 = BELTWIDE

            CONSURPLUS  PROSURPLUS  TOTSURPLUS

NORTHEAST         17.2         2.8        20.0
LAKESTATES         5.7         6.9        12.6
CORNBELT          11.0        18.4        29.4
NORTHPLAIN         1.7        20.1        21.8
APPALACHIA         7.1         8.4        15.5
SOUTHEAST          8.2         0.4         8.7
DELTASTATE         3.0         9.2        12.2
SOUTHPLAIN         6.4         3.8        10.2
MOUNTAIN           4.2        -6.4        -2.1
PACIFIC           11.1       -30.1       -19.0
EAST              17.7         2.8        20.5
STHEAST           15.6        10.4        26.0
MIDWEST           16.7        25.3        42.0
WEST              13.9       -39.1       -25.2
STHCENTRAL         8.5        11.4        20.0
NORTHERNPL         3.1        22.7        25.9


----  63030 PARAMETER REGSURPER1    PERCENTAGE CHANGE IS HIGHER THAN 0.1

INDEX 1 = BELTWIDE

            PROSURPLUS

NORTHEAST         0.21
LAKESTATES        0.20
CORNBELT          0.24
NORTHPLAIN        0.49
APPALACHIA        0.43
DELTASTATE        0.77
SOUTHPLAIN        0.22
MOUNTAIN         -0.23
PACIFIC          -0.91
EAST              0.20
STHEAST           0.32
MIDWEST           0.23
WEST             -0.79
STHCENTRAL        0.44
NORTHERNPL        0.43
```

```
----  63048 PARAMETER PRICE         MAJOR CROP PRICE IN DOLLARS

                  BASE  HIGHPLAINS    BELTWIDE

COTTON          328.85      328.85      316.11
CORN              2.47        2.47        2.47
SOYBEANS          5.61        5.61        5.62
WHEAT             3.59        3.59        3.60
SORGHUM           2.21        2.21        2.23
RICE             10.61       10.61       10.61
BARLEY            2.56        2.56        2.54
OATS              1.20        1.20        1.22
SILAGE           13.38       13.38       13.41
HAY              72.95       72.95       73.12
SUGARCANE       200.94      200.94      200.70
SUGARBEET       200.94      200.94      200.70
POTATOES          5.66        5.66        5.65
TOMATOFRSH        9.44        9.44        9.54
TOMATOPROC       60.31       60.31       59.59
ORANGEFRSH        5.07        5.07        5.07
ORANGEPROC        5.00        5.00        5.04
GRPFRTFRSH        4.16        4.16        4.18
GRPFRTPROC        3.36        3.36        2.90


----  63049 PARAMETER PRICEDIF      PRICE CHANGE FOR MAJOR CROPS IN DOLLARS

              BELTWIDE

COTTON          -12.74
SOYBEANS          0.01
WHEAT             0.01
SORGHUM           0.01
BARLEY           -0.02
OATS              0.02
SILAGE            0.03
HAY               0.16
SUGARCANE        -0.24
SUGARBEET        -0.24
POTATOES         -0.01
TOMATOFRSH        0.10
TOMATOPROC       -0.72
ORANGEPROC        0.04
GRPFRTFRSH        0.02
GRPFRTPROC       -0.46
```

```
----  63051 PARAMETER PRICEPC1      PERCENTAGE PRICE CHANGE HIGHER THAN 1

              BELTWIDE

COTTON           -3.87
OATS              1.64
TOMATOFRSH        1.09
TOMATOPROC       -1.19
GRPFRTPROC      -13.70


----  63053 PARAMETER NATIONLAND    NATIONAL TOTAL HARVEST ACRE IN 1000

                  BASE  HIGHPLAINS    BELTWIDE

COTTON            9741        9738        9950
CORN             68805       68805       68988
SOYBEANS         55275       55276       55299
WHEAT            63887       63888       63493
SORGHUM          14168       14169       14281
RICE              2784        2784        2783
BARLEY            8647        8647        8620
OATS              6212        6212        6216
SILAGE            7929        7929        7937
HAY              67710       67710       67783
SUGARCANE          565         565         574
SUGARBEET         1179        1179        1176
POTATOES          1415        1415        1415
TOMATOFRSH         138         138         138
TOMATOPROC         304         304         304
ORANGEFRSH         187         187         188
ORANGEPROC         615         615         615
GRPFRTFRSH          84          84          81
GRPFRTPROC          89          89          92


----  63054 PARAMETER NATLANDABS    NATIONAL LAND ABSOLUTE   CHANGE WRT BASE

            HIGHPLAINS    BELTWIDE

COTTON              -3         209
CORN                           183
SOYBEANS             1          23
WHEAT                1        -394
SORGHUM                        113
RICE                            -1
BARLEY                         -28
OATS                             4
SILAGE                           7
HAY                             73
```

```
SUGARCANE                        9
SUGARBEET                       -3
POTATOES                         1
ORANGEFRSH                       1
GRPFRTFRSH                      -3
GRPFRTPROC                       4


----  63056 PARAMETER NATLANDP2     PERCENTAGE CHANGE IS HIGHER THAN 2

              BELTWIDE

COTTON            2.15
GRPFRTFRSH       -3.63
GRPFRTPROC        4.04


----  63058 PARAMETER PRODUCTION    NATIONAL PRODUCTION IN 1000 UNITS

                  BASE  HIGHPLAINS    BELTWIDE

COTTON           14280       14280       14518
CORN           8984782     8984780     8988392
SOYBEANS       2191883     2191883     2191883
WHEAT          2472013     2472021     2458725
SORGHUM         906825      906842      905500
RICE            154675      154675      154675
BARLEY          465159      465161      463699
OATS            325590      325590      325590
SILAGE          118303      118303      118194
HAY             128961      128961      128937
SUGARCANE         2977        2977        3037
SUGARBEET         6374        6373        6359
POTATOES        407839      407839      407839
TOMATOFRSH      146940      146940      145067
TOMATOPROC        8598        8598        8599
ORANGEFRSH       54225       54225       54045
ORANGEPROC      200958      200958      200958
GRPFRTFRSH       30269       30269       30269
GRPFRTPROC       36115       36116       36610
```

```
----  63059 PARAMETER PRODUCTDIF    ABSOLUTE   CHANGE WRT BASE

            HIGHPLAINS    BELTWIDE

COTTON                         238
CORN                -2        3610
WHEAT                8      -13288
SORGHUM             17       -1325
BARLEY               3       -1459
SILAGE               1        -109
HAY                            -24
SUGARCANE                       60
SUGARBEET           -1         -15
TOMATOFRSH                   -1873
TOMATOPROC                       1
ORANGEFRSH                    -180
GRPFRTPROC                     494


----  63061 PARAMETER PRODUCTPC1    PERCENTAGE CHANGE IS HIGHER THAN 1

              BELTWIDE

COTTON            1.67
SUGARCANE         2.00
TOMATOFRSH       -1.27
GRPFRTPROC        1.37
```

Table 4.5 Output For Model Verification of Post-Harvest Loss Reduction Study

```
----  60720 PARAMETER PHLDATA   PHL DATA ASSUMPTIONS BY SCENARIO

                     POTATOES  TOMATOFRSH  ORANGEFRSH  GRPFRTFRSH

TOM1  .TOMATOFRSH                   1.000
TOM3  .TOMATOFRSH                   3.000
TOM5  .TOMATOFRSH                   5.000
POT1  .POTATOES         1.000
POT3  .POTATOES         3.000
POT5  .POTATOES         5.000
CIT1  .ORANGEFRSH                               1.000
CIT1  .GRPFRTFRSH                                           1.000
CIT3  .ORANGEFRSH                               3.000
CIT3  .GRPFRTFRSH                                           3.000
CIT5  .ORANGEFRSH                               5.000
CIT5  .GRPFRTFRSH                                           5.000
JOINT1.POTATOES         1.000
JOINT1.TOMATOFRSH                   1.000
JOINT1.ORANGEFRSH                               1.000
JOINT1.GRPFRTFRSH                                           1.000
JOINT3.POTATOES         3.000
JOINT3.TOMATOFRSH                   3.000
JOINT3.ORANGEFRSH                               3.000
JOINT3.GRPFRTFRSH                                           3.000
JOINT5.POTATOES         5.000
JOINT5.TOMATOFRSH                   5.000
JOINT5.ORANGEFRSH                               5.000
JOINT5.GRPFRTFRSH                                           5.000
```

Table 4.6 Output for Model Verification of Boll Weevil Eradication Study

```
----  60720 PARAMETER BWEDATA    BWE DATA BY SCENARIO

INDEX 1 = HIGHPLAINS

                          DRYLAND       IRRIG

TXHIPLAINS.COTTON           0.250       0.250
TXHIPLAINS.CHEMICALCO      -1.100      -1.100

INDEX 1 = BELTWIDE

                          DRYLAND       IRRIG

ALABAMA   .COTTON           1.950       1.950
ALABAMA   .CHEMICALCO     -19.100     -19.100
ARKANSAS  .COTTON           1.230       1.230
ARKANSAS  .CHEMICALCO     -17.400     -17.400
LOUISIANA .COTTON           3.510       3.510
LOUISIANA .CHEMICALCO     -14.700     -14.700
MISSISSIPP.COTTON           2.280       2.280
MISSISSIPP.CHEMICALCO      -8.700      -8.700
MISSOURI  .COTTON           2.090       2.090
MISSOURI  .CHEMICALCO     -23.400     -23.400
OKLAHOMA  .COTTON           2.510       2.510
OKLAHOMA  .CHEMICALCO     -27.400     -27.400
TENNESSEE .COTTON           6.120
TENNESSEE .CHEMICALCO     -48.900
TXHIPLAINS.COTTON           0.250       0.250
TXHIPLAINS.CHEMICALCO      -1.100      -1.100
TXROLINGPL.COTTON           3.470       3.470
TXROLINGPL.CHEMICALCO     -23.400     -23.400
TXCNTBLACK.COTTON           4.240       4.240
TXCNTBLACK.CHEMICALCO     -15.000     -15.000
TXEAST    .COTTON           4.350       4.350
TXEAST    .CHEMICALCO     -35.500     -35.500
TXEDPLAT  .COTTON           5.890       5.890
TXEDPLAT  .CHEMICALCO     -44.200     -44.200
TXCOASTBE .COTTON           4.290
TXCOASTBE .CHEMICALCO     -29.700
TXSOUTH   .COTTON           3.290       3.290
TXSOUTH   .CHEMICALCO     -38.400     -38.400
TXTRANSPEC.COTTON                       1.560
TXTRANSPEC.CHEMICALCO                 -30.400
```

5. BIBLIOGRAPHY

Brooke, A., D. Kendrick, and A. Meeraus. GAMS: A User's Guide. Boyd and Fraser Publishers, Version 2.25, 1993.

GAMS Development Corporation. "Sensitivity Analysis with GAMS/CPLEX and GAMS/OSL." Washington, DC, 1993.
Web page: http://www.gams.com/docs/sensitiv.htm

GAMS Development Corporation. "Guide to the 'Put' Writing Facility." Washington, DC, 1990.
Web page: http://www.gams.com/docs/putt.htm

McCarl, B.A. "GAMSCHK USER DOCUMENTATION: A System for Examining the Structure and Solution Properties of Linear Programming Problems Solved using GAMS." Working Documentation, Department of Agricultural Economics, Texas A&M University, 1994.

McCarl, B.A. “GAMSBAS Users Guide.” 1996.
Web page: http://agrinet.tamu.edu/mccarl/softdown.htm

McCarl, B.A. “GAMSCHK Users Guide.” 1996.
Web page: http://agrinet.tamu.edu/mccarl/softdown.htm

McCarl, B.A. “Installation Instructions for GAMSBAS and GAMSCHK.” 1996.
Web page: http://agrinet.tamu.edu/mccarl/softdown.htm

McCarl, B.A., A. Meeraus, W.I. Nayda, R. Ramen, and P. Steacy. “So Your GAMS Model Didn’t Work Right: A Guide to Model Repair.” Draft book, Department of Agricultural Economics, Texas A&M University, 1996.

McCarl, B.A., C.C. Chang, J.D. Atwood, and W.I. Nayda. “Documentation of the ASM: The U.S. Agricultural Sector Model.” Draft book, Department of Agricultural Economics, Texas A&M University, 1995.

McCarl, B.A., T.H. Spreen. “Applied Mathematical Programming Using Algebraic Systems.” Course Notes for Agricultural Economics 641, Texas A&M University, College Station, TX, 1996.

Appendix A Table of Major GAMS Syntax Items

A number of GAMS features are quite commonly used in the document above. Here we present a very brief glossary of these with a short definition of the purpose, a place in the ASM structure where this item is used and then references to the supporting documents giving the location where the material is covered.

Item	Purpose	Example of Usage in ASM[1]	Location in Reference where Discussion of Item can be Found[2]
SET	Defines indexes in algebraic model	Table 2.1 Lines 1-5	BKM Chapter 4 MS Chapter 5.1.1
Parameter	Defines indexed data items either with explicit values or to be calculated	Table 2.1 Lines 7-15 (value) Table 3.1 Lines 61-66 (calculations)	BKM Chapter 5,6 MS Chapter 5.1.3
Scalar	Defines a single unindexed data item	Section 3.2.1 Example	BKM Chapter 5 MS Chapter 5.1.3
=	Used in calculating parameter values in assignment statements	Section 3.2.1	BKM Chapter 6 MS Chapter 5.1.2,5.1.3,5.1.4
Table	Defines multidimensional indexed items and include values	Table 2.1 Lines 17-28	BKM Chapter 5 MS Chapter 5.1.3
Variables	Defines unknowns in optimization model without sign restriction	Table 2.1 line 32	BKM Chapter 7 MS Chapter 5.1.5

[1] A reference to Table and Line refers to a place in this manual. A filename refers to a ASM file.

[2] MS stands for McCarl and Spreen.
ME stands for McCarl et al.
BKM stands for Brooke, Kendrick and Meeraus

Item	Purpose	Example of Usage in ASM[1]	Location in Reference where Discussion of Item can be Found[2]
Positive Variables	Defines nonnegative unknowns in optimization model	Table 2.1 Line 31	BKM Chapter 7 MS Chapter 5.1.5
Equations	Names and dimensions constraints in optimization model	Table 2.1 Lines 34-36	BKM Chapter 8 MS Chapter 5.1.6
..	Used in specifying algebraic statement of programming model constraints and objective function	Table 2.1 Lines 38-44	BKM Chapter 8 MS Chapter 5.1.7
$ONTEXT $OFFTEXT	Causes entries in between to be treated as comment	Table 4.2 Lines 104,108	BKM page 42
*	Causes a line to be treated as comment when placed in column 1	Table 4.2 Lines 49-52	BKM page 42
.UP, .LO	Used to enter upper and lower bounds on variables	File ASMCALRN Lines 141,144	BKM Chapter 7 MS Chapter 5.1.8
.SCALE	Used to enter scaling for variables and equations	File FASCALE Line 385	ME Chapter 10

[1] A reference to Table and Line refers to a place in this manual. A filename refers to a ASM file.

[2] MS stands for McCarl and Spreen.
ME stands for McCarl et al.
BKM stands for Brooke, Kendrick and Meeraus

Item	Purpose	Example of Usage in ASM[1]	Location in Reference where Discussion of Item can be Found[2]
.L	Used to reference optimal value for variable	Table 3.1 Lines 64-66	BKM pg. 122-123
.m	Used to reference optimal value for a dual variable (shadow price)	Table 3.1 Lines 64-66	BKM page 31
**** $Number	GAMS marker for compilation errors		BKM pg. 122-123 ME Chapter 6
Display	Mechanism with which GAMS displays output	Table 3.1 Line 68	BKM Chapter 13
Option Preceding Display	Method to control decimals and format of display	Table 4.3 Line 174-203	BKM pg. 145-8
$INCLUDE	Mechanism for including files in a GAMS code	Table 4.1 Line 233	BKM pg. 273
$BATINCLUDE	Mechanism for including a file with arguments		BKM pg. 273
$	Conditional in setup of programming model equation or calculation	Table 3.1 Line 43 Section 6.3.1 step 5 Section 3.2	BKM pg 72, 92-5 ME Chapter 12

[1] A reference to Table and Line refers to a place in this manual. A filename refers to a ASM file.

[2] MS stands for McCarl and Spreen.
ME stands for McCarl et al.
BKM stands for Brooke, Kendrick and Meeraus

Item	Purpose	Example of Usage in ASM[1]	Location in Reference where Discussion of Item can be Found[2]
LOOP	Syntax which allows repeated execution of a code segment	Table 3.1 Line 87	BKM Chapter 12 pg. 278
IF	Syntax which allows conditional execution of a code segment	File ASMCALRN Lines 233-244	BKM pg. 283-4
PUT	Alternative output choice. Allows spreadsheet communication		BKM pg. 275-281
SOLVE	Syntax which causes model solution to occur	Table 2.1 Line 47	BKM Chapter 9
MODEL	Syntax which tells what equations are in optimization model	Table 2.1 Line 46	BKM Chapter 9

.

[1] A reference to Table and Line refers to a place in this manual. A filename refers to a ASM file.

[2] MS stands for McCarl and Spreen.
ME stands for McCarl et al.
BKM stands for Brooke, Kendrick and Meeraus

Appendix B Selected List of Input Parameters - Alphabetical Order

Input	Primary Input	Brief Description of Item Contents
Item Name	file	
AUMSSUP	demand.96	SUBREGIONAL AUMS SUPPLY
CCCBUDDATA	crop.96	CROP BUDGET DATA
CRPSTUFF	crp.96	MISCELLANEOUS CROP INFORMATION
EROSION	erosion.96	EROSION DATA
FARMPROD	fpdata.92	FARM PROGRAM DATA
FEDBEEF	natmix.96	FED STEER AND HEIFERS BY REGION (1975-1992)
FPPART	fpdata.92	FARM PROGRAM PARTICIPATION RATES
HEIFCV	natmix.96	HEIFER CALVES BY REGION (1975-1992)
HEIFYEAR	natmix.96	HEIFER YEARLINGS BY REGION (1975-1992)
INPUTPRICE	demand.96	NATIONAL INPUT PRICES
IRRMIX63	irrmix.96	IRRIGATED CROP ACREAGE BY REGION (1982,1987,1992)
LABORSUP	demand.96	REGIONAL LABOR SUPPLY
LANDAVAIL	demand.96	LAND AVAILABILITY BY CLASS
LANDSUPPL	demand.96	REGIONAL LAND SUPPLY DATA
LBUDDATA	beeffeed	LIVESTOCK BUDGET DATA
LIVESTK	natmix.96	LIVESTOCK TYPES IN VARIOUS SUBREGIONS
MIXDATA	mix.96	HISTORICAL CROP MIX BY REGION
NATMIXDATA	natmix.96	PRIMARY PRODUCT MIXDATA
PDEMAND	demand.96	PRIMARY COMMODITY DOMESTIC DEMAND DATA
PEXPORT	demand.96	PRIMARY COMMODITY EXPORT DEMAND DATA
PIMPORT	demand.96	PRIMARY COMMODITY IMPORT DEMAND DATA
POPULATION	demand.96	POPULATION BY SUBREGION FOR WELFARE ACCOUNTING
SDEMAND	demand.96	SECONDARY COMMODITY DOMESTIC DEMAND DATA
SEXPORT	demand.96	SECONDARY COMMODITY EXPORT DEMAND DATA
SIMPORT	demand.96	SECONDARY COMMODITY IMPORT DEMAND DATA
SKIPINPUT	asmcalsu	SKIP INPUT CALCULATION
SOWSLAUGHT	natmix.96	POUNDS OF CULL SOWS SLAUGHTERED BY REGION (1975 - 199
STEERCV	natmix.96	STEER CALVES (1975-1992)
STEERYEAR	natmix.96	STEER YEARLINGS (1975-1992)
WATERSUP	demand.96	SUBREGIONAL WATER SUPPLY DATA

Appendix C Selected List of Potential Output Parameters - Alphabetical Order

Output Item Name	Primary Defining File	Brief Description of Item Contents
ACRECOMP	asmcompr	REGIONAL HARVESTED ACRE COMPARISON
AUMSSUM	asmrept	AUMS USE SUMMARY
BALANCE	runreport	PRIMARY PRODUCT DEMAND AND SUPPLY BALANCE
BALANCEP	asmrept	PRIMARY PRODUCT SUPPLY DEMAND BALANCE
BALANCES	asmrept	SECONDARY SUPPLY DEMAND BALANCE
CBALANCEP	asmcompr	PRODUCTION BALANCE SHEET
CBALANCES	asmcompr	SECONDARY PRODUCTS BALANCE SHEET
CFWELFARE	pestloop	DETAIL ON FOREIGN SURPLUS
COSTC	asmcalrn	SUM OF INPUT AND COST FOR CROPS
COSTL	asmcalrn	SUM OF INPUT AND COST FOR LIVESTOCK
CPCONSUR	pestloop	DETAIL ON DOMESTIC CONSUMER SURPLUS
CROPMIXABS	runreport	ABSOLUTE CHANGE WRT BASE
CROPMIXPCN	runreport	PERCENTAGE CHANGE WRT BASE
CROPMIXR	runreport	SUBREGIONAL TOTAL HARVEST ACRE IN 1000
CROPPRCOMP	asmcompr	REGIONAL CROP PRODUCTION COMPARISON
CROPSUBREG	asmrept	SUBREGIONAL CROP PRODUCTION REPORT
DRYCOMP	asmcompr	DRYLAND CROP ACREAGE
DRYMIXABS	runreport	ABSOLUTE CHANGE WRT BASE
DRYMIXPCN	runreport	PERCENTAGE CHANGE WRT BASE
DRYMIXR	runreport	SUBREGIONAL DRYLAND ACRE IN 1000 ACRES
DRYREGCOMP	asmcompr	DRYLAND REGIONAL ACREAGE
EROSIONCP	asmcompr	NATIONAL EROSION COMPARISON
EXPORTCOMP	asmcompr	EXPORT COMPARISON
FCONSCOMP	asmcompr	FOREIGN SURPLUS COMPARISON
GOVCCC	asmrept	GOVERNMENT CCC LOAN COST SUMMARY
GOVCOMP	asmcompr	GOVERNMENT PROGRAM PAYMENT COMPARISON
GOVDEF	asmrept	GOVERNMENT DEFICIENCY PAYMENT SUMMARY
GROSSRV	asmrept	GROSS REVENUE REPORT BY COMMODITY
HARVEST	asmrept	NATIONAL HARVESTED ACREAGE REPORT
HARVESTREG	asmrept	REGIONAL HARVESTED ACREAGE REPORT
HARVESTSUB	asmrept	SUBREGIONAL HARVESTED ACREAGE REPORT
IACR	asmrept	SUBREGIONAL CROP ACREAGE
IACRSUM	asmrept	TOTAL SUBREGIONAL CROP ACREAGE
IMPORTCOMP	asmcompr	IMPORT QUANTITY COMPARISON
INPCOMP	asmcompr	RESOURCE USE RESULTS COMPARISON
INPVALCOMP	asmcompr	AVERAGE VARIABLE FACTOR PRICES

IRRCOMP	asmcompr	IRRIGATED CROP ACREAGE COMPARISON
IRRMIXABS	runreport	ABSOLUTE CHANGE WRT BASE
IRRMIXPCN	runreport	PERCENTAGE CHANGE WRT BASE
IRRMIXR	runreport	SUBREGIONAL IRRIGATION ACRE IN 1000
IRRREGCOMP	asmcompr	IRRIGATED REGIONAL ACREAGE COMPARISON
LABORSUM	asmrept	LABOR USE SUMMARY
LANDSUM	asmrept	LAND USE SUMMARY
LIVEABS	runreport	ABSOLUTE CHANGE WRT BASE
LIVECOMP	asmcompr	REGIONAL LIVESTOCK PRODUCTION COMPARISON
LIVEPCNT	runreport	PERCENTAGE CHANGE WRT BASE
LVSTKRATIO	natmix.96	PROPORTION OF LIVESTOCK IN A REGION
LVSTKTOT	natmix.96	TOTAL LIVESTOCK IN A COMPOSITE REGION
NATINPCOMP	asmcompr	NATIONAL IN PUT USE RESULTS COMPARISON
NATINPUSE	asmrept	NATIONAL IN PUT USE SUMMARY
NATIONLAND	runreport	NATIONAL TOTAL HARVEST ACRE IN 1000
NATLANDABS	runreport	NATIONAL LAND ABSOLUTE CHANGE WRT BASE
NATLANDPER	runreport	NATIONAL LAND PERCENTAGE CHANGE WRT BASE
OTAEXPORT	asmcompr	NATIONAL EXPORT VALUE IN MIL $
PCONSCOMP	asmcompr	CONSUMER SURPLUS COMPARISON
PCONSUR	asmrept	DOMESTIC CONSUMER SURPLUS
PRICECOMP	asmcompr	COMMODITY PRICE COMPARISON
PRICEDIF	runreport	PRICE CHANGE FOR MAJOR CROPS IN DOLLARS
PRICEPER	runreport	PERCENTAGE PRICE CHANGE FOR MAJOR CROPS
PRICSDIF	runreport	PRIMARY C.S. DIFFERENCE WITH BASE
PRIMARYCS	runreport	PRIMARY C.S.
PROCSUM	asmrept	PROCESSING SUMMARY
PRODNCOMP	asmcompr	COMMODITY PRODUCTION COMPARISON
PRODUCTDIF	runreport	ABSOLUTE CHANGE WRT BASE
PRODUCTION	runreport	NATIONAL PRODUCTION IN 1000 UNITS
PRODUCTPER	runreport	PERCENTAGE CHANGE WRT BASE
REGCOMP	asmcompr	REGIONAL INPUT COMPARISON
REROSION	asmcompr	REGIONAL EROSION
REVCOMP	asmcompr	GROSS REVENUE COMPARISON
RLABORSUM	asmrept	REGIONAL LABOR USAGE
RNATINPUT	asmcompr	REGIONAL INPUT USE RESULT COMPARISON
SURABS	runreport	ABSOLUTE CHANGE WRT BASE IN MILLION
SURCOMP	asmcompr	WELFARE COMPARISON IN BILLION$
SURPERCHN	runreport	PERCENTAGE CHANGE WRT BASE
SURPLUS	runreport	NATIONAL TOTAL SURPLUS IN MILLION
TXREPORT	asmrept	TEXAS INPUT USE AND PRODUCER SURPLUS
WATERSUM	asmrept	WATER USE SUMMARY

WATERSUP	pestloop	SUBREGIONAL WATER SUPPLY DATA
WELSUM	asmrept	SOCIAL WELFARE SUMMARY REPORT

Appendix D   List of All Items Used in ASM And Reference Locations

Due to the complex multiple file structure of ASM, a user looking for a particular data item might need to go on a hunting expedition. However, a program was recently written which maps out the incidence of parameters (i.e., data items) and sets in the ASM file structure.[13] The resultant output appears below. The output contains a table for sets (D.1), one for parameters (D.2), one for equations (D.3), one for variables (D.4), and one for models (D.5). The columns in the tables are defined as follows:

ITEM NAME   This is the name of the item in the GAMS listing. For example, in Table D.1, the first set is named AGPRODCT.

DECLARED   This tells where the set, parameter, equation, or variable statement appears.

DEFINED   Tells where values are entered into the item. This indicates which items have exogenous set labels or numerical data specified, and in which files they appear.

ASSIGNED   Indicates the files in which values are computed into this item.

CONTROL   Indicates where a set is used to sum over various items or define an equation.

REF   Indicates the ASM files where an item is used in calculations or display statements.

For example, in Table D.1 the set COST is declared in the SETS file, and there it is given a value. Then it is used as a control in ASMCALSU, ASMMODEL, and ASMCALRN, and it is referenced in the same files. If one then wishes to find out where a data item is used, one can use this table to find the ASM files where the data item is used, and then use a text editor to locate the data item within each ASM file.7

[13] When this program is ready for public release it will be made available to ASM users.

Table D.1 Files Where Actions on SETS Appear

ITEM NAME	DECLARED	DEFINED	ASSIGNED	CONTROL	REF
AGPRODCT	reptsets	reptsets			reptsets
ALLI	sets	sets		sets	sets
				fpdata.92	fpdata.92
				asmcheck	proc.96
				asmcalsu	beeffeed
				allofit	crop.96
				mixit	asmcheck
				reptsets	asmcalsu
				fascale	allofit
				pestloop	mixit
					reptsets
					asmmodel
					fascale
					pestloop
ALLRUNS	pestloop	pestloop		pestloop	pestloop
ALLSUBREG	sets	sets			sets
					crp.96
					reptsets
					pestloop
ANIMAL	sets	sets		natmix.96	sets

				asmcalsu	beeffeed
				asmmodel	natmix.96
				fascale	asmcalsu
				asmcalrn	reptsets
					asmmodel
					fascale
					asmcalrn
AUMSIT	reptsets	reptsets			reptsets
AUMSITEM	sets	sets		pestloop	demand.96
					pestloop
BALITEM	reptsets	reptsets		runreport	reptsets
					pestloop
					runreport
C	sets	sets		asmsolve	sets
					asmsolve
CHANGE	pestloop	pestloop		pestloop	pestloop
COMPOSITE	sets	sets		natmix.96	sets
				crp.96	natmix.96
					crp.96
COST	sets	sets		asmcalsu	asmcalsu

				asmmodel	asmmodel
				asmcalrn	asmcalrn
CROP	sets	sets		asmcheck	sets
				asmcalsu	crop.96
				allofit	mix.96
				mixit	irrmix.96
				asmmodel	asmcheck
				fascale	asmcalsu
				asmcalrn	allofit
				def	mixit
				pestloop	erosion.96
				runreport	reptsets
					asmmodel
					fascale
					asmcalrn
					pestloop
					runreport
CRPINFO	crp.96	crp.96		crp.96	crp.96
CRPMIXALT	sets	sets		asmcheck	mix.96
				mixit	irrmix.96
				asmmodel	asmcheck
				fascale	mixit

				asmcalrn	asmmodel
					asmcalrn
					pestloop
CTECH	sets	sets		crop.96	crop.96
				asmcheck	asmcheck
				asmcalsu	asmcalsu
				allofit	allofit
				mixit	mixit
				asmmodel	reptsets
				fascale	asmmodel
				asmcalrn	fascale
				pestloop	asmcalrn
				runreport	pestloop
					runreport
ENTER	reptsets	reptsets			reptsets
ESOURCE	erosion.96	erosion.96			erosion.96
FARMPRO	fpdata.92	fpdata.92		asmcalsu	fpdata.92
					asmcalsu
GROSSITEM	reptsets	reptsets			reptsets
HARVLAND	reptsets	reptsets			reptsets

INPITEM	pestloop	pestloop			pestloop
INPUT	sets	sets		sets asmcalsu asmmodel fascale asmcalrn	demand.96 asmcalsu reptsets asmmodel fascale asmcalrn pestloop
ITEMCCC	reptsets	reptsets			reptsets
ITEMCS	reptsets	reptsets			reptsets pestloop
ITEMDEF	reptsets	reptsets			reptsets pestloop
ITEMFOR	reptsets	reptsets			reptsets pestloop
ITER	sets	sets		asmsolve	sets asmmodel asmsolve
LABORITEM	sets	sets		pestloop	demand.96 pestloop

LABRITEM	reptsets	reptsets			reptsets
LANDITEM	sets	sets		pestloop	demand.96 pestloop
LANDITM	reptsets	reptsets			reptsets
LANDTYPE	sets	sets		sets asmcalsu asmmodel fascale asmcalrn pestloop	demand.96 asmcalsu reptsets asmmodel fascale asmcalrn pestloop
LIVESTOCK	sets	sets			reptsets pestloop runreport
LIVETECH	sets	sets		natmix.96 asmcheck asmcalsu asmmodel fascale asmcalrn	beeffeed natmix.96 asmcheck asmcalsu asmmodel fascale

					asmcalrn
LREGION	sets	sets		asmmodel fascale asmcalrn	asmmodel asmcalrn
MAPPING	sets	sets		asmcalsu	asmcalsu asmmodel
MAPTHEM	sets	sets		natmix.96 crp.96	natmix.96 crp.96
MIXITEM	mixit	mixit			mixit
MIXTYPE	sets	sets			sets reptsets
NATMIXALT	sets	sets		natmix.96 asmmodel fascale asmcalrn	natmix.96 asmmodel asmcalrn
PRIMARY	sets	sets		sets natmix.96 asmcalsu reptsets asmmodel	sets demand.96 natmix.96 asmcalsu reptsets

				fascale asmcalrn pestloop runreport	asmmodel fascale asmcalrn pestloop runreport
PROCESSALT	sets	sets		asmcalsu allofit asmmodel fascale asmcalrn	sets proc.96 asmcalsu allofit reptsets asmmodel fascale asmcalrn
PROCITEM	reptsets	reptsets			reptsets
PROXY	mixit	mixit		mixit	
PROXY2	mixit	mixit		mixit	
REGION	asmcalsu		asmcalsu	asmcalsu	asmcalsu
REGIONS	sets	sets		asmcalsu pestloop	sets demand.96 asmcalsu reptsets

					pestloop
REGITEM	pestloop	pestloop			pestloop
REPITEM	pestloop	pestloop			pestloop
RUN	pestloop	pestloop		pestloop	pestloop
				runreport	runreport
SDITEM	sets	sets		pestloop	demand.96
					pestloop
SECONDARY	sets	sets		sets	demand.96
				asmcalsu	asmcalsu
				reptsets	reptsets
				asmmodel	asmmodel
				fascale	fascale
				asmcalrn	asmcalrn
				pestloop	pestloop
SEX	natmix.96	natmix.96		natmix.96	natmix.96
SORGFEED	sets	sets		allofit	
STEPS	asmseper	asmseper		asmmodel	asmseper
				fascale	asmmodel
SUBREG	sets	sets		crop.96	sets

		natmix.96	demand.96
		asmcheck	fpdata.92
		asmcalsu	beeffeed
		acreirrs	crop.96
		allofit	mix.96
		mixit	irrmix.96
		crp.96	allofit
		asmmodel	natmix.96
		fascale	asmcheck
		asmcalrn	asmcalsu
		pestloop	acreirrs
		runreport	mixit
			crp.96
			erosion.96
			reptsets
			asmmodel
			fascale
			asmcalrn
			pestloop
			runreport
SUBREG2	allofit	natmix.96	natmix.96
		mixit	asmcheck
			mixit

SUBYEAR	asmcheck	asmcheck		asmcheck	asmcheck
SURITEM	pestloop	pestloop			pestloop
TECH	sets	sets		crop.96 asmcheck asmcalsu allofit mixit asmmodel fascale asmcalrn pestloop runreport	crop.96 asmcheck asmcalsu allofit mixit asmmodel fascale asmcalrn pestloop runreport
TECHNOLOGY	pestloop	pestloop		pestloop	pestloop
TYPESOF	reptsets	reptsets			reptsets
WATERITEM	sets	sets		pestloop	demand.96 pestloop
WATRITEM	reptsets	reptsets			reptsets
WELITEM	reptsets	reptsets			reptsets
WTECH	sets	sets		crop.96	crop.96

```
asmcheck     asmcheck

asmcalsu     asmcalsu

allofit      allofit

asmmodel     mixit

fascale      asmmodel

asmcalrn     fascale

pestloop     asmcalrn

runreport    pestloop

             runreport
```

Table D.2 Files Where Actions on PARAMETERS Appear

ITEM NAME	DECLARED	DEFINED	ASSIGNED	REF
ADJUST	pestloop		pestloop	pestloop
ASCALE	fascale	fascale	fascale	fascale
AUMSSUP	demand.96	demand.96		asmcalsu asmmodel asmcalrn pestloop
BADFEED	natmix.96		natmix.96	natmix.96
BADHY	natmix.96		natmix.96	natmix.96
BADPIG	natmix.96		natmix.96	natmix.96
BADSC	natmix.96		natmix.96	natmix.96
BADSY	natmix.96		natmix.96	natmix.96
BALANCE	runreport		runreport	runreport
BSCALE	fascale		fascale	fascale
BWEADJ	pestloop	pestloop		pestloop
CBALANCEP	pestloop			runreport
CBUDDATA	allofit		allofit mixit pestloop	allofit mixit asmmodel fascale asmcalrn pestloop
CCCBUDDATA	crop.96	crop.96	crop.96 asmcheck asmcalsu	crop.96 asmcheck asmcalsu allofit
CONVERGE	sets	sets	asmsolve	asmsolve

COSTC	asmcalsu		asmcalrn	asmmodel
COSTL	asmcalsu		asmcalrn	asmmodel
CROPMIXABS	runreport		runreport	runreport
CROPMIXPC2	runreport		runreport	runreport
CROPMIXPCN	runreport		runreport	runreport
CROPMIXR	runreport		runreport	runreport
CRPSTUFF	crp.96	crp.96	crp.96	crp.96
CSCALE	fascale		fascale	fascale
DRYMIXABS	runreport		runreport	runreport
DRYMIXPC2	runreport		runreport	runreport
DRYMIXPCN	runreport		runreport	runreport
DRYMIXR	runreport		runreport	runreport
EROSIONCP	pestloop			runreport
FARMPROD	fpdata.92	fpdata.92	fpdata.92 asmcalsu def asmsolve pestloop	fpdata.92 asmcalsu asmmodel asmsolve pestloop
FEDBEEF	natmix.96	natmix.96		natmix.96
FPPART	fpdata.92	fpdata.92		asmcalsu asmmodel
GOODDBUD	mixit		mixit	mixit
GOODIBUD	mixit		mixit	mixit
GROSSRV	reptsets			pestloop
HEIFCV	natmix.96	natmix.96	natmix.96	natmix.96
HEIFYEAR	natmix.96	natmix.96	natmix.96	natmix.96

IACR	pestloop			runreport
INPUTPRICE	demand.96	demand.96		asmcalsu asmmodel asmcalrn
IRRMIX63	irrmix.96	irrmix.96	mixit	mixit asmmodel asmcalrn
IRRMIXABS	runreport		runreport	runreport
IRRMIXPC2	runreport		runreport	runreport
IRRMIXPCN	runreport		runreport	runreport
IRRMIXR	runreport		runreport	runreport
ISCALE	fascale	fascale	fascale	fascale
LABORSUP	demand.96	demand.96		asmcalsu asmmodel asmcalrn pestloop
LANDAVAIL	demand.96	demand.96	crp.96	asmmodel
LANDSUPPL	demand.96	demand.96		asmcalsu asmmodel asmcalrn pestloop
LBUDDATA	beeffeed	beeffeed	asmcheck asmcalsu	natmix.96 asmcheck asmcalsu asmmodel fascale asmcalrn
LIM	asmmodel		asmmodel	asmsolve
LIVEMIX	sets		asmmodel	reptsets
LIVESTK	natmix.96	natmix.96		natmix.96
LIVESUM	natmix.96		natmix.96	natmix.96

LVSTKRATIO	natmix.96		natmix.96	natmix.96
LVSTKTOT	natmix.96		natmix.96	natmix.96
MIXADJUST	mixit		mixit	mixit
MIXCHECK	asmcheck		asmcheck	asmcheck
MIXDATA	mix.96	mix.96	asmcheck mixit	asmcheck mixit asmmodel asmcalrn
MSCALE	fascale		fascale	fascale
NATINPCOMP	pestloop			runreport
NATINPUSE	reptsets			reptsets
NATIONLAND	runreport		runreport	runreport
NATLANDABS	runreport		runreport	runreport
NATLANDP2	runreport		runreport	runreport
NATLANDPER	runreport		runreport	runreport
NATMIXDATA	natmix.96	natmix.96	natmix.96	natmix.96 asmmodel asmcalrn
NATMIXMAX	natmix.96		natmix.96	natmix.96
NEEDDBUD	mixit		mixit	mixit
NEEDIBUD	mixit		mixit	mixit
NEEDIRRMIX	asmmodel		asmmodel	asmmodel
PCONSCOMP	pestloop			runreport
PDEMAND	demand.96	demand.96	asmcalsu allofit asmcalrn	asmcalsu asmmodel asmcalrn pestloop
PEXPORT	demand.96	demand.96	asmcalrn	asmmodel

				asmcalrn pestloop
PIMPORT	demand.96	demand.96		asmmodel asmcalrn pestloop
PRICE	runreport		runreport	runreport
PRICEDIF	runreport		runreport	runreport
PRICEPC1	runreport		runreport	runreport
PRICEPER	runreport		runreport	runreport
PRIMARYCS	runreport		runreport	runreport
PROCBUD	proc.96	proc.96	asmcalsu allofit	asmcalsu allofit asmmodel asmcalrn
PRODUCTDIF	runreport		runreport	runreport
PRODUCTION	runreport		runreport	runreport
PRODUCTPC1	runreport		runreport	runreport
PRODUCTPER	runreport		runreport	runreport
PROFITL	asmcalsu		asmcalsu	asmcalsu
PROFITPR	asmcalsu		asmcalsu	asmcalsu
PSCALE	fascale	fascale	fascale	fascale
QINC	asmseper	asmseper		asmmodel
REGMIX	natmix.96		natmix.96	natmix.96
REGPCNT	natmix.96		natmix.96	natmix.96
RESULT	sets		asmsolve	asmsolve
SAVEHORSE	asmcheck		asmcheck	asmcheck
SCALE	sets		sets	fascale

			fascale	asmcalrn asmsolve
SCALLIVE	sets	sets	sets fascale	fascale asmcalrn
SCALMIX	sets	sets	sets fascale	fascale asmcalrn
SCALOBJ	sets	sets	sets fascale	asmsolve
SCALPROC	sets	sets	sets fascale	fascale asmcalrn
SCALPROD	sets	sets	sets fascale	fascale asmcalrn pestloop
SCBUDDATA	pestloop		pestloop	pestloop
SDEMAND	demand.96	demand.96	asmcalrn	asmcalsu asmmodel asmcalrn pestloop
SEPAG	asmseper	asmseper	asmseper asmmodel	asmmodel fascale
SEXPORT	demand.96	demand.96	asmcalrn	asmmodel asmcalrn pestloop
SIMPORT	demand.96	demand.96		asmmodel asmcalrn pestloop
SKIPINPUT	asmcalsu	asmcalsu	pestloop	asmcalrn
SOWSLAUGHT	natmix.96	natmix.96	natmix.96	natmix.96
SSCALE	fascale	fascale	fascale	fascale
STEERCV	natmix.96	natmix.96	natmix.96	natmix.96
STEERYEAR	natmix.96	natmix.96	natmix.96	natmix.96

SURABS	runreport		runreport	runreport
SURCOMP	pestloop			runreport
SURPERCHN	runreport		runreport	runreport
SURPLUS	runreport		runreport	runreport
TOL	sets	sets	asmmodel asmsolve	asmsolve
TOLER	sets		asmsolve	asmsolve
WATERSUP	demand.96	demand.96		asmcalsu asmmodel asmcalrn pestloop
YESIRRMIX	asmmodel	asmmodel		asmmodel

Table D.3 Files Where Actions on EQUATIONS Appear

ITEM NAME	DECLARED	DEFINED	ASSIGNED	REF
ARTIFICIAL	asmmodel	asmmodel	fascale	asmmodel
AUMSCONVEX	asmmodel	asmmodel		asmmodel
AUMSIDENT	asmmodel	asmmodel		asmmodel
AUMSR	asmmodel	asmmodel	fascale	asmmodel
DEMPCONVEX	asmmodel	asmmodel		asmmodel
DEMSCONVEX	asmmodel	asmmodel		asmmodel
DIVERT	asmmodel	asmmodel	fascale	asmmodel
EXPPCONVEX	asmmodel	asmmodel		asmmodel
EXPSCONVEX	asmmodel	asmmodel		asmmodel
FAMILYLIM	asmmodel	asmmodel	fascale	asmmodel
FIX	asmmodel	asmmodel	fascale	asmmodel
FRMPROG	asmmodel	asmmodel	fascale	asmmodel asmsolve
HIRELIM	asmmodel	asmmodel	fascale	asmmodel
IMIXREG	asmmodel	asmmodel		asmmodel
IMIXREGTOT	asmmodel	asmmodel		asmmodel
IMPPCONVEX	asmmodel	asmmodel		asmmodel
IMPSCONVEX	asmmodel	asmmodel		asmmodel
LABOR	asmmodel	asmmodel	fascale	asmmodel
LABRCONVEX	asmmodel	asmmodel		asmmodel
LABRIDENT	asmmodel	asmmodel		asmmodel
LAND	asmmodel	asmmodel	fascale	asmmodel

LANDCONVEX	asmmodel	asmmodel		asmmodel
LANDIDENT	asmmodel	asmmodel		asmmodel
MAXLAND	asmmodel	asmmodel	fascale	asmmodel
MIXNAT	asmmodel	asmmodel	fascale	asmmodel
MIXREG	asmmodel	asmmodel	fascale	asmmodel
MIXREGTOT	asmmodel	asmmodel	fascale	asmmodel
OBJT	asmmodel	asmmodel	fascale	asmmodel
P092	asmmodel	asmmodel	fascale	asmmodel
P5092	asmmodel	asmmodel	fascale	asmmodel
PDEMIDENT	asmmodel	asmmodel		asmmodel
PEXPIDENT	asmmodel	asmmodel		asmmodel
PIMPIDENT	asmmodel	asmmodel		asmmodel
PRIMARYBAL	asmmodel	asmmodel	fascale	asmmodel asmsolve
PUBAUMS	asmmodel	asmmodel	fascale	asmmodel
SDEMIDENT	asmmodel	asmmodel		asmmodel
SECONDBAL	asmmodel	asmmodel	fascale	asmmodel
SEXPIDENT	asmmodel	asmmodel		asmmodel
SIMPIDENT	asmmodel	asmmodel		asmmodel
UNHARVEST	asmmodel	asmmodel	fascale	asmmodel
WATERR	asmmodel	asmmodel	fascale	asmmodel
WATRCONVEX	asmmodel	asmmodel		asmmodel
WATRIDENT	asmmodel	asmmodel		asmmodel

Table D.4 Files Where Actions on VARIABLES Appear

ITEM NAME	DECLARED	DEFINED	ASSIGNED	REF
ARTIF	asmmodel		fascale	asmmodel
ARTS	asmmodel			asmmodel
AUMSPRIV	asmmodel		asmmodel fascale asmcalrn	asmmodel
AUMSPRIVS	asmmodel		fascale	asmmodel
AUMSPUB	asmmodel		fascale	asmmodel
CCCLOANP	asmmodel		fascale	asmmodel
CCCLOANS	asmmodel		fascale	asmmodel
CROPBUDGET	asmmodel		fascale asmcalrn	asmmodel pestloop
CSPS	asmmodel		fascale	asmmodel asmsolve
DEFPRODN	asmmodel		fascale	asmmodel
DEMANDP	asmmodel		asmmodel fascale asmcalrn	asmmodel
DEMANDPS	asmmodel		fascale	asmmodel
DEMANDS	asmmodel		asmmodel fascale asmcalrn	asmmodel
DEMANDSS	asmmodel		fascale	asmmodel
DIVPRODN	asmmodel		fascale	asmmodel
EXPORTP	asmmodel		asmmodel fascale asmcalrn	asmmodel
EXPORTPS	asmmodel		fascale	asmmodel

EXPORTS	asmmodel	asmmodel fascale asmcalrn	asmmodel
EXPORTSS	asmmodel	fascale	asmmodel
FAMILY	asmmodel	fascale	asmmodel
HIRED	asmmodel	asmmodel fascale asmcalrn	asmmodel
HIREDS	asmmodel	fascale	asmmodel
IMPORTP	asmmodel	asmmodel fascale asmcalrn	asmmodel
IMPORTPS	asmmodel	fascale	asmmodel
IMPORTS	asmmodel	asmmodel fascale asmcalrn	asmmodel
IMPORTSS	asmmodel	fascale	asmmodel
ITWID	asmmodel	asmcalrn	asmmodel
LANDSUPPLS	asmmodel	fascale	asmmodel
LANDSUPPLY	asmmodel	asmmodel asmcalrn	asmmodel
LVSTBUDGET	asmmodel	fascale asmcalrn	asmmodel
MIXI	asmmodel		asmmodel
MIXR	asmmodel	fascale	asmmodel
NATMIX	asmmodel	fascale	asmmodel
PRDN092	asmmodel	fascale	asmmodel
PRDN5092	asmmodel	fascale	asmmodel

PROCESS	asmmodel	fascale asmcalrn	asmmodel
TOLR	asmmodel	fascale asmcalrn	asmmodel
TWID	asmmodel	fascale asmcalrn	asmmodel
UNHARV	asmmodel	fascale	asmmodel
WATERFIX	asmmodel	fascale	asmmodel
WATERVAR	asmmodel	asmmodel fascale asmcalrn	asmmodel
WATERVARS	asmmodel	fascale	asmmodel

Table D.5 Files Where Actions on MODELS Appear

ITEM NAME	DECLARED	DEFINED	ASSIGNED	REF
SECTOR	asmmodel	asmmodel	asmmodel asmsolve	asmsolve