

# Economic Equilibrium Modeling with GAMS

## *An Introduction to GAMS/MCP and GAMS/MPSGE*

Thomas F. Rutherford

*University of Colorado*

### **Preface, June 1998**

For the past six years I have been meaning to put together a proper user's guide for the GAMS/MPSGE economic modeling language. This is proven to be a difficult task, primarily because I have been too occupied with economics and software to ever think about documentation.<sup>1</sup> The present document is a first attempt to rectify this problem. It consists of five chapters at present, with one more to be added before the end of the summer. The five chapters included here still need work – I want to double or triple the number of exercises here, as I have always found that to be an essential learning device.

It is my intention that this manual will in its final form be able to introduce a second year graduate student working on a correspondence basis to economic equilibrium modeling with GAMS and MPSGE. It seems that there are many such students at universities everywhere, and I always regret that I am unable to provide much guidance for them when I receive the occasional Email or phone call. I hope that this text answer some of their questions.

The book is currently organized as follows. Chapter 1 provides an review of intermediate microeconomics, specifically demand theory and general equilibrium; and it then shows how these ideas can be illustrated using small MPSGE models. Chapter 2 is a somewhat more advanced introduction to general equilibrium analysis with a focus on taxation and public finance. This chapter contains an appendix which describes the MPSGE language in a complete yet compact manner. Chapter 3 concerns cost and expenditure functions, particularly the constant-elasticity of substitution family. The chapter presents two sample programs illustrating how the nested-CES functional form can be calibrated to arbitrary cross-elasticities of substitution in the case of three or four inputs. The final section in this paper collects some definitions and a description of some of the other functional forms which have been widely used in economic equilibrium analysis.

Chapter 4 introduces the mixed complementarity problem in a general format with a range of applications in economics. (This chapter still needs considerable work to produce a set of parallel computational exercise.)

Chapter 5 relates the equilibrium and optimization perspectives on general equilibrium allocations, showing how GAMS can be used to compute the same equilibria in either an optimization or complementarity format. This chapter needs some editing, so that it is clearer how these ideas relate to MPSGE. I will add a few sample programs which compare the various algebraic formulations of the standard model with the corresponding MPSGE models. Of course, this chapter also needs a number of exercises.

---

<sup>1</sup> I will never forget the article in 1982 with those wonderful quotes, “*Real programmers write programs, not documentation.*” and “*Real programmers can write Fortran in any language.*”



The final chapter I wish to add describes how GAMS can be used to compute forward-looking, dynamic general equilibrium models. It builds upon ideas from Chapter 5, comparing the familiar Ramsey model with the complementarity setup.

I will announce the release of the final version on the GAMS list, and I will make note on the GAMS web site. I hope this will happen before the end of summer, 1998.

Demand Theory and General Equilibrium  
*An Intermediate Level Introduction to MPSGE*



## 1. An Overview

This document describes a mathematical programming system for general equilibrium analysis named MPSGE which operates as a subsystem to the mathematical programming language GAMS. MPSGE is a library of function and Jacobian evaluation routines which facilitates the formulation and analysis of AGE models. MPSGE simplifies the modelling process and makes AGE modelling accessible to any economist who is interested in the application of these models. In addition to solving specific modelling problems, the system serves a didactic role as a structured framework in which to think about general equilibrium systems.

MPSGE separates the tasks of model formulation and model solution, thereby freeing model builders from the tedious task of writing model-specific function evaluation subroutines. All features of a particular model are communicated to GAMS/MPSGE through a tabular input format. To use MPSGE, a user must learn the syntax and conventions of this model definition language.

The present paper is intended for students who have completed two semesters of study in microeconomics. The purpose of this presentation is to give students a practical perspective on microeconomic theory. The diligent student who works through all of the examples provided here should be capable of building small models “from scratch” to illustrate basic theory. This is a first step to acquiring a set of useable tools for applied work.

The remainder of this paper is organized as follows. Section 2 provides practical tips on how to get started with GAMS/MPSGE. Section 3 reviews basic ideas from the theory of the consumer. Section 4 introduces the modelling framework with three models illustrating the representation of consumer demand within the MPSGE language. Section 5 reviews the pure exchange model, and Section 5 presents two MPSGE models of exchange. Each of the model-oriented sections present exercises based on the models which give students a chance to work through the material on their own. Section 6 provides solutions to exercises from this paper. Additional introductory examples for self-study can be found in Jim Markusen’s library of MPSGE examples, as well as in the GAMS model library (look for models with names ending in “MGE”).

The level of presentation and diagrammatic exposition adopted here is based on Hal Varian’s undergraduate microeconomics textbook (*Intermediate Microeconomics: A Modern Approach*, Third Edition, W. W. Norton & Company, Inc., 1993).

The ultimate objective of this piece is to remind students of some theory which they have already seen and illustrate how these ideas can be used to build numerical models using GAMS with MPSGE. It is not my intention to provide a graduate level presentation of this material. So far as possible, I have avoided calculus and even algebra. The objective here is to demonstrate that what matters are economic ideas. With the proper tools, it is possible to do concrete economic modeling without a lot of mathematical formalism.

## 2. Getting Started

(i) In order to use GAMS/MPSGE, you need to know how to create and edit text files. There are several methods for doing this. One approach is to use NOTEPAD, the standard text editor under Windows 95. It is also possible to use a text processor such as Microsoft Word as a text editor. If you take this approach, you will need to remember to always save the edited file in a *text* format. A final approach, one which I suggest to graduate students

who are interested in using numerical modelling in their research, is that you take the time to develop some facility with a “real” programmer’s editor like Emacs, Epsilon or Brief. These editors are far more powerful than Notepad, and they are far better suited to computational work than is Microsoft Word.

(ii) You need to have a copy of GAMS with MPSGE to run on your computer. There are versions of this program for PCs as well as for Unix workstations. Copies of the program can be obtained directly from GAMS (*gams@gams.com*), or you may get a copy from someone who already has the program. (Copyright restrictions apply only to GAMS license files.) When you copy the GAMS systems files *without the license file* and the program will only operate in student/demonstration mode. The student version is perfectly adequate for learning about modelling—in fact, it may be better because its dimensionality restrictions prevent the novice model builder from adding unnecessary details.

(iii) You need to install GAMS with MPSGE on your computer. To do this, follow the standard installation procedures.

(iv) You should verify that the system is operational. Connect to a working directory (*never run models from the GAMS system directory!*). Then, extract and run one of the library models, e.g.

```
C:\>MKDIR WORK
C:\>CD WORK
C:\WORK>GAMSLIB SCARFMGE
C:\WORK>GAMS SCARFMGE
```

If the GAMS system is properly installed, these commands will cause GAMS to solve a sequence of models from the `SCARFMGE` sample problem. The output from this process is written to file `SCARFMGE.LST`. If you search for the word “STATUS”. you can verify that all the cases are processed.

There are a number of MPSGE models included in the GAMS library. If you are using a student version of GAMS, you will be able to process some but not all of the library models. The student version of the program limits the number of variables in the model to 100. (I believe that GAMS imposes other limits on use of the student version, but the variable limitation is the most severe constraint.)

Assuming that you have successfully installed the software, let us now proceed to some examples which illustrate both the computing syntax and the underlying economics.

### 3. The Theory of Consumer Demand

A central idea underlying most microeconomic theory is that agents optimize subject to constraints. The optimizing principle applied to consumer choice begins from the notion that agents have preferences over consumption bundles and will always choose the most preferred bundle subject to applicable constraints. To operationalize this theory, three issues which must be addressed: (i) How can we represent preferences? (ii) What is the nature of constraints on consumer choice? and (iii) How can the choice be modelled?

Preferences are relationships between alternative consumption “bundles”. These can be represented graphically using indifference curves, as illustrated in Figure 1. Focusing now

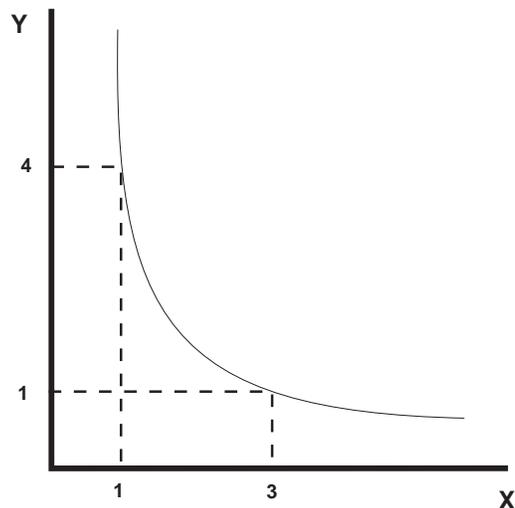


Figure 1. An Indifference Curve

on the preferences of a single consumer, the indifference curve is a line which connects all combinations of two goods  $x$  and  $y$  between which our consumer is indifferent. As this curve is drawn, we have represented an agent with well-behaved preferences: at any allocation, more is better (monotonicity), and averages are preferred to extremes (convexity). Exactly one indifference curve goes through each positive combination of  $x$  and  $y$ . Higher indifference curves lie to the north-east.

If we wish to characterize an agent's preferences, the marginal rate of substitution (MRS) is a useful point of reference. At a given combination of  $x$  and  $y$ , the marginal rate of substitution is the slope of the associated indifference curve. As drawn, the MRS increases in magnitude as we move to the northwest and the MRS decreases as we move to the south east. The intuitive understanding is that the MRS measures the willingness of the consumer to trade off one good for the other. As the consumer has greater amounts of  $x$ , she will be willing to trade more units of  $x$  for each additional unit of  $y$  – this results from convexity.

An ordinal utility function  $U(x, y)$  provides a helpful tool for representing preferences. This is a function which associates a number with each indifference curve. These numbers increase as we move to the northeast, with each successive indifference curve representing bundles which are preferred over the last. The particular number assigned to an indifference curve has no intrinsic meaning. All we know is that if  $U(x_1, y_1) > U(x_2, y_2)$ , then the consumer prefers bundle 1 to bundle 2.

Figure 2 illustrates how it is possible to use a utility function to generate a diagram with the associated indifference curves. This figure illustrates Cobb-Douglas well-behaved preferences which are commonly employed in applied work.

Up to this point, we have we have focused exclusively on the characterization of preferences. Let us now consider the other side of the consumer model – the budget constraint. The simplest approach to characterizing consumer income is to assume that the consumer has a fixed money income which she may spend on any goods. The only constraint on this choice is that the value of the expenditure may not exceed the money income. This is the standard budget constraint:

$$P_x x + P_y y = M.$$

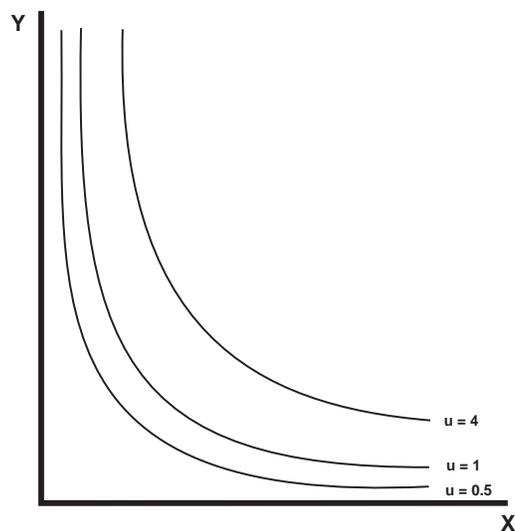


Figure 2. Indifference Curves and Utility Levels



Figure 3. The Budget Set

This equation defines the line depicted graphically in Figure 3. All points inside the budget line are affordable. The consumer faces a choice of which affordable bundle to select.

Within the framework of our theory, the consumer will choose the one combination of  $x$  and  $y$  from the set of affordable bundles which maximizes her utility. This combination of  $x$  and  $y$  will be at the point where the indifference curve is tangent the budget line. This point is called the optimal choice. We see this illustrated in Figure 4.

The standard model of consumer behavior provides a starting point for learning MPSGE. This introduction is “hands on” – I will discuss issues as they arise, assuming that you have access to a computer and can invoke the program and read the output file. You may wish

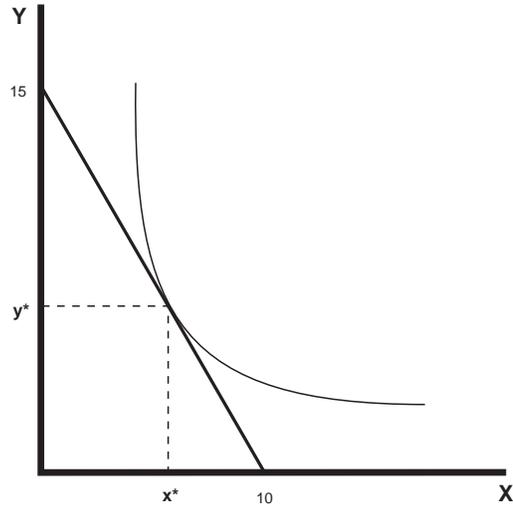


Figure 4. Utility Maximization on the Budget Set

to learn the rudiments of GAMS syntax before starting out, although you may be able to pick up these ideas as they come, depending on your aptitude for computer languages.

#### 4. Modeling Consumer Demand with MPSGE

##### EXAMPLE 1 : EVALUATING A DEMAND FUNCTION

Consider a standard consumer choice problem, one which might appear on a midterm examination in intermediate microeconomics:

$$\max U(x, y) = \ln(x) + 2 \ln(y) \quad \text{s.t.} \quad 1x + 2y = 120$$

where 1 is the exogenous price of  $x$  and 2 is the price of  $y$ .

This type of problem is solved easily using GAMS/MINOS (as a nonlinear program). Strictly speaking, it is not the sort of model for which you would need to use MPSGE. At the same time, this can be an instructive example.

The key issue in this example is learning how to represent utility functions. MPSGE is “non-algebraic” – so function specification depends on an intuitive understanding of the underlying economic structure.

Consider Figure 5 and focus on a single point,  $x = 1$ ,  $y = 1$ . There is an indifference curve through this point, and the marginal rate of substitution (MRS) at this point is simply the slope of this curve. The benchmark MRS does not uniquely determine the underlying preferences.

A utility function is represented in MPSGE by the specification of: (i) benchmark demand quantities, (ii) benchmark demand prices (iii) an elasticity of substitution at the benchmark point. Benchmark quantities determine an anchor point for the set of indifference curves. Benchmark prices fix the slope of the indifference curve at that point, and the elasticity describes the curvature of the indifference curve. Speaking formally, elasticities provide a “second order approximation” of the utility function. To understand the importance of the benchmark elasticity of substitution, consider Figure 6. This figure

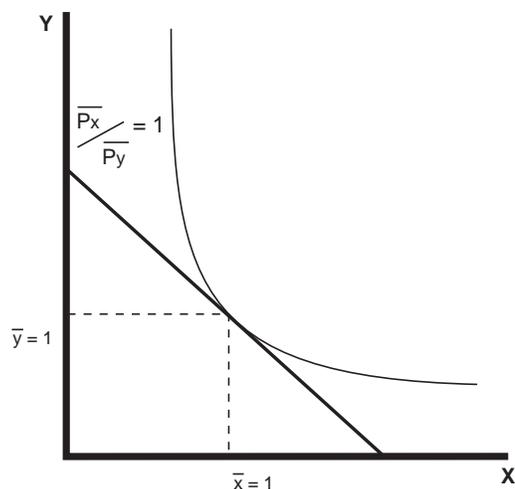


Figure 5. A Calibrated Benchmark

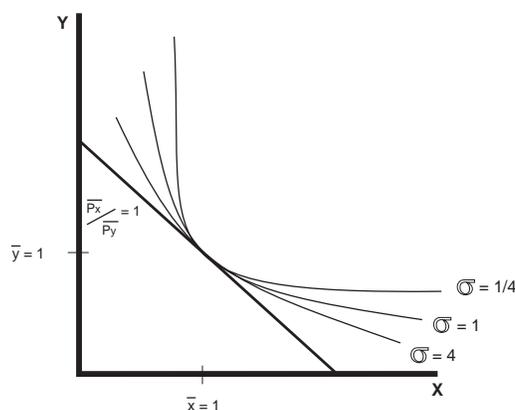


Figure 6. The Elasticity of Substitution

shows three indifference curves all of which share the same benchmark quantities and benchmark prices. They differ only in the elasticities of substitution. The least convex (flattest) curve has the highest elasticity, the most convex curve has the lowest elasticity. (When the elasticity of substitution is 0, the indifference curve is L-shaped with the corner at the benchmark point.)

Let us now consider how the consumer optimization problem can be cast as a general equilibrium model. We do this by adding a single factor of production and two “production” sectors. For concreteness, let the factor of production be called labor with a price  $P_L$ . One production function converts one unit of labor into one unit of  $x$ , the other sector converts 2 units of labor into one unit of  $y$ . Setting the labor endowment equal 120, the market clearance condition for labor reads:

$$1x + 2y = 120$$

which is precisely the budget constraint for the consumer’s problem.

We will now present the program code, a few lines at a time. As part of working through the example, the student should type these lines into a file.

A MPSGE model specification is always listed between `$ONTEXT` and `$OFFTEXT` statements. The first statement within an MPSGE model-description assigns a name to the model. The model name must begin with a letter and must have 10 or fewer characters.

```
$ONTEXT
```

```
$MODEL:DEMAND
```

The model specification begins by declaring variables for the model. In a standard model, there are three types of variables: commodity prices, sectoral activity levels, and consumer incomes. The end of each line may conclude with a “!”, followed by a variable description.

**N.B.** The variables associated with commodities are prices, not quantities. (In this and subsequent models, I use `P` as the first letter for each of the commodity variables to remind us that these variables are prices.)

**N.B.** The variable associated with a consumer is an income level, not a welfare index.

```
$SECTORS:
```

```
  X      ! ACTIVITY LEVEL FOR X = DEMAND FOR GOOD X
  Y      ! ACTIVITY LEVEL FOR Y = DEMAND FOR GOOD Y
```

```
$COMMODITIES:
```

```
  PX     ! PRICE OF X WHICH WILL EQUAL PL
  PY     ! PRICE OF Y WHICH WILL EQUAL 2 PL
  PL     ! PRICE OF THE ARTIFICIAL FACTOR L
```

```
$CONSUMERS:
```

```
  RA     ! REPRESENTATIVE AGENT INCOME
```

Function specifications follow the variable declarations. In this model, our first declarations correspond to the two production sectors. In this model, the production structures are particularly simple. Each of the sectors has one input and one output. In the MPSGE syntax, `I:` denotes an input and `O:` denotes an output. The output quantity coefficients for both sectors are unity (`Q:1`). This means that the level values for  $x$  and  $y$  equal the quantities produced.

The final function specified in the model represents the utility function and endowments for the single consumer. In this function, the `E:` entries correspond to endowments and the `D:` entries are demands. Reference demands, reference prices and the substitution elasticity (`s:1`) characterize preferences.

The demand entries shown here are consistent with a Cobb-Douglas utility function in which the budget share for  $y$  is twice the budget share for  $x$  (i.e. the MRS at (1,1) equals 1/2):

```
$PROD:X
```

```

O:PX  Q:1
I:PL  Q:1

$PROD:Y
O:PY  Q:1
I:PL  Q:2

$DEMAND:RA  s:1
E:PL  Q:120
D:PX  Q:1  P:(1/2)
D:PY  Q:1  P:1

$OFFTEXT

```

The final three statements in this file invoke the MPSGE preprocessor, “generate” and solve the model:

```

$SYSINCLUDE mpsgeset DEMAND

$INCLUDE DEMAND.GEN
SOLVE DEMAND USING MCP;

```

The preprocessor invocation (“\$SYSINCLUDE mpsgeset”) should be placed immediately following the \$OFFTEXT block containing the model description. The model generator code, DEMAND.GEN, is produced by the previous statement and must be included immediately before the SOLVE statement.

At this point, the reader should take the time to type the example into a file and execute the program with GAMS/MPSGE.

This is possibly the first GAMS model which some readers have solved, so it is worth looking through the listing file in some detail. After running the solver, we examine the listing file. I typically begin my assessment of a model’s solution by searching for “STATUS”. For this model, we have the following:

```

          S O L V E      S U M M A R Y

MODEL    DEMAND
TYPE     MCP
SOLVER   PATH                FROM LINE  263

**** SOLVER STATUS      1 NORMAL COMPLETION
**** MODEL STATUS      1 OPTIMAL

RESOURCE USAGE, LIMIT      1.432      1000.000
ITERATION COUNT, LIMIT     5          1000
EVALUATION ERRORS         0           0

```

Work space allocated                    --        4.86 Mb

Default price normalization using income for RA

This information is largely self-explanatory. The most important items are the **SOLVER STATUS** and **MODEL STATUS** indicators. When the solver status is 1 and the model status is 1, the system has returned an equilibrium.

For small models such as this, the limits on resource usage (time) and solver iterations have no effect. You can modify these values by adding statements “*model.RESLIM = number of cpu seconds;*” and “*model.ITERLIM = number of iterations;*” before the **SOLVE** statement.

The work space allocation for MPSGE models is determined by the number of variables in the model. It is possible to exogenously specify the work space allocation by assigning `model.WORKSPACE = xx`; where `xx` is the desired number of megabytes.

The final message, “Default price normalization...” reminds the user that an Arrow-Debreu general equilibrium model determines only relative prices. In such an equilibrium, the absolute scaling of prices is indeterminate. (I.e., if  $(p^*, M^*)$  are a set of equilibrium prices and income levels, then  $(2p^*, 2M^*)$  is also a solution, etc.)

It is common practice in economics to address the normalization issue through the specification of a numeraire commodity. You can do this for an MPSGE model by “fixing” a price, with a statement like: “`PX.FX = 1;`” entered following the model declaration (`$$SYSINCLUDE mpsgeset`) but prior to the solver invocation. When any price or income level is fixed, MPSGE recognizes that a numeraire has been specified and does no automatic normalization.

Following some output from the solver (PATH in this case), the listing file provides a complete report of equilibrium values. With MPSGE models, the equation listings are superfluous. The variable listings provide all the relevant information.

For this model, the solution listing appears as follows:

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR X	.	40.000	+INF	.
---- VAR Y	.	40.000	+INF	.
---- VAR PX	.	1.000	+INF	.
---- VAR PY	.	2.000	+INF	.
---- VAR PL	.	1.000	+INF	.
---- VAR RA	.	120.000	+INF	.

X	SUPPLY AND DEMAND OF GOOD X
Y	SUPPLY AND DEMAND OF GOOD Y
PX	PRICE OF X WHICH WILL EQUAL CX * PL
PY	PRICE OF Y WHICH WILL EQUAL CY * PL
PL	PRICE OF THE ARTIFICIAL FACTOR L
RA	REPRESENTATIVE AGENT INCOME

The **LOWER** and **UPPER** columns report variable bounds applied in the model. In these columns, zero is represented by “.” and infinity is represented by “+INF”. The **LEVEL** column reports the solution value returned by the algorithm. Here we see that the equilibrium price of  $x$  is 1 and the price of  $y$  is 2, as determined by the specification of labor inputs. The **MARGINAL** column reports the net balance of a variable’s *associated equation*. In the case of an activity level, the **MARGINAL** entry is  $-1 \times$  the activity’s net profit. For a commodity price, **MARGINAL** reports the net market excess supply. For a consumer, this column reports the income balance.

#### EXERCISES FOR EXAMPLE 1:

- (i) The utility function calibration point is arbitrary. Here, we have selected  $x=y=1$  as the reference quantity. Revise the program to use a different calibration point where  $x=2$  and  $y=1$ , where  $MRS(2,1) = 1/4$ . (Remember to modify both the **Q:** and **P:** fields.) Rerun the model to demonstrate that this does not change the result.
- (ii) Increase the price of  $x$  from 1 to 2 by changing the **Q:** coefficient for **PL** in sector **X** from 1 to 2. What happens to the demand for  $x$ ? Explain why a change in the price of  $x$  is represented by a change in the **Q:** field for sector **X**.
- (iii) Compute an equilibrium in which commodity  $y$  is defined as the numeraire.

#### EXAMPLE 2 : EVALUATING THE MARGINAL RATE OF SUBSTITUTION

This example further explores the representation of demand functions with MPSGE. It sets up a trivial equilibrium model with two goods and one consumer which returns the marginal rate of substitution of good  $x$  for good  $y$  at a given level of demand.

The underlying utility function is:  $U(x,y) = \ln(x) + 4 \ln(y)$  When  $x = y = 1$ , the marginal rate of substitution of  $x$  for  $y$  is  $1/4$ . We use this information to calibrate the demand function, specifying the ratio of the reference prices of  $x$  to  $y$  equal to  $1/4$ .

In an equilibrium, final demand always equals endowments for both goods, because these are the only sources of demand and supply. The model as parameterized demonstrates that if we set endowments for this model equal to the demand function calibration point, the model equilibrium price ratio equals the benchmark MRS.

This program begins with some GAMS statements in which three scalar parameters are declared. These parameters will be used in the place of numbers within the MPSGE model. The syntax for these GAMS statements is introduced in Chapter 2 of the GAMS manual. In short, we declare  $x$ ,  $y$  and MRS as scalar parameters and initialize the first two of these to unity. The MRS parameter is assigned a value following the solution of the model.

```
SCALAR
    X      QUANTITY OF X FOR WHICH THE MRS IS TO BE EVALUATED /1/
    Y      QUANTITY OF Y FOR WHICH THE MRS IS TO BE EVALUATED /1/
    MRS    COMPUTED MARGINAL RATE OF SUBSTITUTION;
```

The remainder of the MPSGE program is, in fact, simpler than Example 1.

```
$ONTEXT

$MODEL:MRSCAL

$COMMODITIES:
    PX      ! PRICE INDEX FOR GOOD X
    PY      ! PRICE INDEX FOR GOOD Y

$CONSUMERS:
    RA      ! REPRESENTATIVE AGENT

$DEMAND:RA  s:1
    D:PX    Q:1    P:(1/4)
    D:PY    Q:1    P:1
    E:PX    Q:X
    E:PY    Q:Y

$OFFTEXT

$SYSINCLUDE mpsgeset MRSCAL

$INCLUDE MRSCAL.GEN
SOLVE MRSCAL USING MCP;
```

Following the solution, we compute a function of the solution values, the ratio of the price of  $x$  to the price of  $y$ . We do this using the GAMS syntax which references the equilibrium level values of the PX and PY and storing this result in the scalar MRS. This scalar value is then displayed in the listing file with 8 digits:

```
MRS = PX.L / PY.L;
OPTION MRS:8;
DISPLAY MRS;
```

## EXERCISES FOR EXAMPLE 2:

(a) Show that the demand function is homothetic by uniform scaling of the  $x$  and  $y$  endowments. The resulting MRS should remain unchanged.

(b) Modify the demand function calibration point so that the reference prices of both  $x$  and  $y$  equal unity (hint: the marginal rate of substitution is:

$$MRS = x/(4y).$$

## EXAMPLE 3 : GOODS DEMAND, LEISURE DEMAND AND LABOR SUPPLY

This model investigates the labor-leisure decision. A single consumer is endowed with labor which is either supplied to the market or “repurchased” as leisure. The consumer utility function over market goods ( $x$  and  $y$ ) and leisure is Cobb-Douglas:

$$U(x, y, L) = \ln(x) + \ln(y) + \ln(L)$$

Goods  $x$  and  $y$  may only be purchased using funds obtained from labor sales. This constraint is written:

$$x + y = \phi LS$$

where goods  $x$  and  $y$  both have a price of unity at base year productivity and  $\phi$  is an index of labor productivity ( $\phi$  is denoted PHI in the program). An increase in productivity is equivalent to a proportional decrease in the prices of  $x$  and  $y$ .

The model declaration is as follows:

```
SCALAR          PHI          AGGREGATE LABOR PRODUCTIVITY /1/,
                CX          COST OF X AT BASE YEAR PRODUCTIVITY /1/,
                CY          COST OF Y AT BASE YEAR PRODUCTIVITY /1/;

$ONTEXT

$MODEL:LSUPPLY

$SECTORS:
    X          ! SUPPLY=DEMAND FOR X
    Y          ! SUPPLY=DEMAND FOR Y
    LS         ! LABOR SUPPLY

$COMMODITIES:
    PX         ! MARKET PRICE OF GOOD X
    PY         ! MARKET PRICE OF GOOD Y
    PL         ! MARKET WAGE
```

```

        PLS      ! CONSUMER VALUE OF LEISURE

$CONSUMERS:
        RA      ! REPRESENTATIVE AGENT

$PROD:LS
        O:PL    Q:PHI
        I:PLS   Q:1

$PROD:X
        O:PX    Q:1
        I:PL    Q:CX

$PROD:Y
        O:PY    Q:1
        I:PL    Q:CY

$DEMAND:RA  s:1
        E:PLS   Q:120
        D:PLS   Q:1     P:1
        D:PX    Q:1     P:1
        D:PY    Q:1     P:1

$OFFTEXT
$SYSINCLUDE mpsgeset LSUPPLY

$INCLUDE LSUPPLY.GEN
SOLVE LSUPPLY USING MCP;

```

We can use this model to evaluate the wage elasticity of labor supply. In the initial equilibrium (computed in the last statement) the demands for  $x$ ,  $y$  and  $L$  all equal 40. A subsequent assignment to PHI (below) increases labor productivity. After computing a new equilibrium, we can use the change in labor supply to determine the wage elasticity of labor supply, an important parameter in labor market studies.

It should be emphasized that the elasticity of labor supply should be an input rather than an output of a general equilibrium model – this is a parameter for which econometric estimates can be obtained.

Here is how the programming works. First, we declare some scalar parameters which we will use for reporting, then save the “benchmark” labor supply in LSO:

```

SCALAR
        LSO    REFERENCE LEVEL OF LABOR SUPPLY
        ELS    ELASTICITY OF LABOR SUPPLY WRT REAL WAGE;

LSO = LS.L;

```

Next, we modify the value of scalar **PHI**, increasing labor productivity by 1%. Because this is a neoclassical model, this change is equivalent to increasing the real wage by 1%. We need to recompute equilibrium prices after having changed the **PHI** value:

```
PHI = 1.01;
$INCLUDE LSUPPLY.GEN
SOLVE LSUPPLY USING MCP;
```

We use this solution to compute and report the elasticity of labor supply as the percentage change in the **LS** activity:

```
ELS = 100 * (LS.L - LS0) / LS0;
DISPLAY ELS;
```

As the model is currently constructed, the wage elasticity of labor supply equals zero. This is because the utility function is Cobb-Douglas over goods and leisure, and the consumer's only source of income is labor. As the real wage rises, this increases both the demand for goods (labor supply) and the demand for leisure. These effect exactly balance out and the supply of labor is unchanged.

(a) One way in which the labor supply elasticity might differ from zero in a model with Cobb-Douglas final demand is if there were income from some other source. Let the consumer be endowed with good  $x$  in addition to labor. What  $x$  endowment is consistent with a labor supply elasticity equal to 0.15?

*Hint:* Let  $\eta$  be the uncompensated labor supply elasticity. Algebraic derivation leads to the following formula:

$$\eta = (\theta_\ell P_x E_X) / (P_L L(1 - \theta_\ell) - \theta_\ell P_x E_X)$$

where:

$\theta_\ell$  is the value share of leisure,

$P_x$  is the price of  $X$ ,

$E_X$  is the endowment of commodity  $X$ ,

$P_L$  is the price of labor,

$L$  is the endowment of labor.

(b) A second way to calibrate the labor supply elasticity is to change the utility function. We can do this by changing the **s:1** to **s:SIGMA**, where **SIGMA** is a scalar value representing the benchmark elasticity of substitution between  $x$ ,  $y$  and  $L$  in final demand. Modify the program to include **SIGMA** as a scalar, and find the value for **SIGMA** consistent with a labor supply elasticity equal to 0.15.

*Hint:* Let  $\eta$  be the uncompensated labor supply elasticity. The algebraic derivation of  $\eta$  using a CES utility function, leads to the following formula:

$$\eta = \frac{LEIS}{LSUP}(\sigma - 1)(1 - \theta_\ell)$$

where:

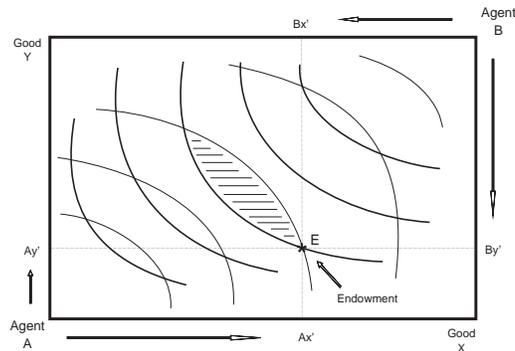


Figure 7. The Edgeworth-Bowley Box

$\theta_\ell$  is the value share of leisure (Sh1),

$LEIS$  is the benchmark demand for leisure,

$LSUP$  is the benchmark supply of labor, and

$\sigma$  is the compensated labor supply elasticity.

## 5. The Pure Exchange Model

Partial equilibrium analysis forms the basis of most economics courses at the undergraduate level. In these models we focus on price, supply and demand for a single commodity. The partial equilibrium approach neglects indirect effects, through which changes in the demand or supply for one good may influence the market for another good.

In the previous section, we focused on the choices of a single consumer. In the present section, we will explore the implications of interactions between many consumers with heterogeneous preferences. Furthermore, the analysis will explore the potentially important interaction between market prices and income which are determined jointly in a general equilibrium.

The most widely-used graphical framework for multi-agent exchange equilibrium analysis is the Edgeworth-Bowley box as illustrated in Figure 7. In this diagram we model the following economy:

Two types of consumers, denoted  $A$  and  $B$ . We consider  $A$  and  $B$  to each represent many households, each with the same endowments and preferences. (This assumption justifies an assumption of perfectly competitive, price-taking behavior.) There are two commodities in the model, denoted  $x$  and  $y$ . Each consumer has fixed endowments of both goods. The horizontal axis measures the total world endowment of good  $x$ . The vertical axis measures the total world endowment of good  $y$ . Any point in the box then represents an allocation of goods between the two agents. The agent  $A$  allocation is measured with respect to the lower left origin. The agent  $B$  allocation is measured with respect to the upper right origin.

Each agent has a given initial endowment, here denoted point  $E$ . Furthermore, we assume that there is no possibility for trade. The indifference curves through point  $E$  therefore represent autarchy welfare levels.

The key idea in this model is that trade can improve both agents' welfare. One agent gives some amount good  $x$  to the other in return for an amount of good  $y$ . The *terms of trade*, the rate of exchange between  $x$  and  $y$ , is determined by the model. The model illustrates a number of important properties of market economies:

- (i) Trade is mutually beneficial. So long as the transactions are voluntary, neither  $H$  nor  $F$  will be hurt by engaging in trade.
- (ii) Market prices can be used to guide the economy to a Pareto-efficient allocation, a state of affairs in which further mutually-beneficial trades are not possible.
- (iii) There is no guarantee that the gains from trade will be “fairly distributed” across consumers. A competitive equilibrium may produce a significant welfare increase for one consumer while have negligible impact on the other.
- (iv) There are multiple Pareto-efficient allocations, typically only one of which is a competitive equilibrium. We can use this model to demonstrate that the issues of efficiency and equity can be separated when there is the possibility of lump-sum income transfers between agents.

#### EXAMPLE 4: A 2X2 EXCHANGE MODEL

In this program, we examine the simple two good, two agent model of exchange equilibrium. The world endowments for goods  $x$  and  $y$  are both equal to 1. Six parameters are used to parameterize the model. These are declared as scalars at the top of the program:

```
SCALAR  XA          AGENT A ENDOWMENT OF X ( 0 < XA < 1)    /0.2/
        YA          AGENT A ENDOWMENT OF Y ( 0 < YA < 1)    /0.8/
        THETA_A     AGENT A DEMAND SHARE PARAMETER FOR X    /0.5/
        THETA_B     AGENT B DEMAND SHARE PARAMETER FOR X    /0.8/
        SIGMA_A     AGENT A ELASTICITY PARAMETER            /2.0/
        SIGMA_B     AGENT B ELASTICITY PARAMETER            /0.5/;
```

This model is actually simpler than the models presented above because we have no need for production. There are simply two commodities and two consumers. The consumers differ in terms of commodity endowments and preferences. The competitive equilibrium prices are such that supply equals demand for both goods and both agents spend an amount equal to their endowment income.

This model illustrates how to use computed function coefficients. See, for example,  $Q:(1-THETA\_A)$  in the  $\$DEMAND:A$  block. Any numeric input field in an MPSGE model may be “computed”, provided that the algebraic expression is enclosed within parentheses and legitimate GAMS code.

This model specification uses the default values for reference prices in the demand function blocks. When  $P: value$  is not specified in a  $D:,I:$  or  $O:$  record,  $P:1$  is assumed.

```
$ONTEXT
```

```

$MODEL: EXCHANGE

$COMMODITIES:
    PX      ! EXCHANGE PRICE OF GOOD X
    PY      ! EXCHANGE PRICE OF GOOD Y

$CONSUMERS:
    A       ! CONSUMER A
    B       ! CONSUMER B

$DEMAND: A  s: SIGMA_A
    E: PX   Q: XA
    E: PY   Q: YA
    D: PX   Q: THETA_A
    D: PY   Q: (1-THETA_A)

$DEMAND: B  s: SIGMA_B
    E: PX   Q: (1-XA)
    E: PY   Q: (1-YA)
    D: PX   Q: THETA_B
    D: PY   Q: (1-THETA_B)

$OFFTEXT
$SYSINCLUDE mpsgeset EXCHANGE

$INCLUDE EXCHANGE.GEN
SOLVE EXCHANGE USING MCP;

SCALAR
    PRATIO  EQUILIBRIUM PRICE OF X IN TERMS OF Y,
    IRATIO  EQUILIBRIUM RATIO OF CONSUMER INCOMES;

PRATIO = PX.L / PY.L;
IRATIO = A.L / B.L;

DISPLAY IRATIO, PRATIO;

```

The foregoing sets up the model and computes the competitive equilibrium. After GAMS returns from the solver, we declare and compute some report values.

Absolute levels of income and price returned from a general equilibrium model are not meaningful because a model determines only relative prices. For this reason, we report equilibrium income and price levels in relative terms.

In the final step, we compute an alternative efficient equilibrium, one in which the income levels for A and B are equal. The purpose of this exercise is to demonstrate the second welfare theorem. When incomes are both fixed, the equilibrium remains efficient, but the connection between market prices and endowment income is eliminated.

In GAMS/MPSGE, a variable may be fixed using an assignment statement such as: “*variable.fx = value;*”, as illustrated in this model:

```
A.FX = 1;
B.FX = 1;
$INCLUDE EXCHANGE.GEN
SOLVE EXCHANGE USING MCP;

SCALAR      TRANSFER  IMPLIED TRANSFER FROM A TO B AS A PERCENTAGE OF INCOME;

TRANSFER = 100 * ( A.L - PX.L * XA - PY.L * YA ); PRATIO = PX.L / PY.L;
IRATIO = A.L / B.L;

DISPLAY TRANSFER, PRATIO, IRATIO;
```

EXERCISES FOR EXAMPLE 4:

- (i) Set up a separate models which computes the autarchy price ratios for consumers A and B. (You can use one of the earlier models as a starting point.)
- (ii) Determine parameter values for which the endowment point is the equilibrium point.
- (iii) Set up a series of computations from which you can sketch the efficiency locus. Draw the Edgeworth box diagram which is consistent with these values.

## 6. Import Tariffs and Market Power

The exchange model provides a remarkably useful tool for analyzing issues related to international trade. Applied models of international trade typically include production technologies, but these are omitted in this treatment for the sake of simplicity. In this model, we will consider a generalization of the earlier  $2 \times 2$  exchange model. In this extension, we introduce independent markets for consumers *A* and *B* and trade activities which deliver goods from one market to the other.

The set of input parameters largely the same as in the previous example. Two new parameters are ad-valorem tariffs which apply on imports to each of the regions.

```
SCALAR  XA          AGENT A ENDOWMENT OF X ( 0 < XA < 1)      /0.2/
        YA          AGENT A ENDOWMENT OF Y ( 0 < YA < 1)      /0.8/
        THETA_A     AGENT A DEMAND SHARE PARAMETER FOR X      /0.4/
        THETA_B     AGENT B DEMAND SHARE PARAMETER FOR X      /0.6/
        SIGMA_A     AGENT A ELASTICITY PARAMETER              /1.0/
        SIGMA_B     AGENT B ELASTICITY PARAMETER              /1.0/,
        T_A         AD-VALOREM TARIFF ON IMPORTS TO AGENT A  /0.10/
        T_B         AD-VALOREM TARIFF ON IMPORTS TO AGENT B  /0.10/;
```

The program differs from the previous example in several respects. First, we introduce a separate commodity price for each agent. In the absence of tariffs, these prices are identical.

A second difference is that in this model trade activities deliver goods from one agent to the other. These are denoted  $M\{good\}\{agent\}$  for imports of *good* to *agent*. There are four flows which may be operated in only one direction (the activity levels are non-negative). In terms of initial endowments and preferences, this model has exactly the same economic structure as the previous model.

\$ONTEXT

\$MODEL:TARIFFS

\$SECTORS:

```

MXA      ! TRADE IN X FROM B TO A
MXB      ! TRADE IN X FROM A TO B
MYA      ! TRADE IN Y FROM B TO A
MYB      ! TRADE IN Y FROM A TO B

```

\$COMMODITIES:

```

PXA      ! PRICE OF GOOD X FOR AGENT A
PYA      ! PRICE OF GOOD Y FOR AGENT A
PXB      ! PRICE OF GOOD X FOR AGENT B
PYB      ! PRICE OF GOOD Y FOR AGENT B

```

\$CONSUMERS:

```

A        ! CONSUMER A
B        ! CONSUMER B

```

\$DEMAND:A s:SIGMA\_A

```

E:PXA    Q:XA
E:PYA    Q:YA
D:PXA    Q:THETA_A
D:PYA    Q:(1-THETA_A)

```

\$DEMAND:B s:SIGMA\_B

```

E:PXB    Q:(1-XA)
E:PYB    Q:(1-YA)
D:PXB    Q:THETA_B
D:PYB    Q:(1-THETA_B)

```

The trade activities each have one input and one output. They simply deliver a good (*X* or *Y*) from one agent's market to the other's market. The new syntax presented here is specification of an ad-valorem tax. Adding a tax requires two new fields. The first is "A:" which specifies the tax agent, a consumer who collects the tax revenue as part of income. The second is "T:" which specifies the ad-valorem tax rate.

**N.B.** In an MPSGE model taxes may be applied on production inputs and outputs, but taxes may not be applied to final demand.

The tax applies on a *net basis* on inputs. For example, if we consider the MxA sector, the price of one unit of input is given by:  $Px_B(1 + T_A)$ , where  $Px_B$  is the net of tax price of a unit of  $x$  in the agent  $B$  market and  $T_A$  is the ad-valorem tariff rate.

```

$PROD:MXA
  O:PX A Q:1
  I:PX B Q:1  A:A  T:T_A

$PROD:MXB
  O:PX B Q:1
  I:PX A Q:1  A:B  T:T_B

$PROD:MYA
  O:PY A Q:1
  I:PY B Q:1  A:A  T:T_A

$PROD:MYB
  O:PY B Q:1
  I:PY A Q:1  A:B  T:T_B

```

The final portions of the file introduces MPSGE report variables. In this case, report variables are used to recover a Hicksian money-metric welfare index for each of the agents. In the REPORT block, a V: field designates a variable name which must be distinct, and a W: field indicates that the variable is to return a welfare index for the specified consumer.

We first compute the tariff-ridden equilibrium which defines benchmark welfare levels. After this calculation, we set all tariffs to zero and compute the free-trade equilibrium. Using welfare indices from the counterfactual and welfare levels from the benchmark, we are able to report the change in welfare associated with the removal of tariff distortions.

```

$REPORT:
  V:WA  W:A
  V:WB  W:B

$OFFTEXT
$SYSINCLUDE mpsgeset TARIFFS

$INCLUDE TARIFFS.GEN
SOLVE TARIFFS USING MCP;

SCALAR
  WAO  BENCHMARK WELFARE INDEX FOR AGENT A
  WBO  BENCHMARK WELFARE INDEX FOR AGENT B;

```

```

WAO = WA.L;
WBO = WB.L;
T_A = 0;
T_B = 0;

```

```

$INCLUDE TARIFFS.GEN
SOLVE TARIFFS USING MCP;

```

```

SCALAR

```

```

    EVA    HICKSIAN EQUIVALENT VARIATION FOR AGENT A
    EVB    HICKSIAN EQUIVALENT VARIATION FOR AGENT B;

```

```

EVA = 100 * (WA.L-WAO)/WAO;
EVB = 100 * (WB.L-WBO)/WBO;
DISPLAY EVA, EVB;

```

EXERCISES FOR EXAMPLE 5:

- (i) Find the *optimal tariff* in this model for agent A, assuming that agent B does not retaliate and leaves her tariff rate at the benchmark level.
- (ii) Insert the endowment and preference parameters from the previous problem, retaining the same benchmark tariff rates. Does free trade benefit both countries? If not, why not?

## 7. Solutions to Exercises

SOLUTION TO EXAMPLE 1

```

$title: Solution to Example 1

```

```

*      Declare the benchmark values as scalars to allow for changes when
*      doing the exercises:

```

```

SCALAR

```

```

    XO      LEVEL OF X AT BENCHMARK
    YO      LEVEL OF Y AT BENCHMARK
    PXO     PX AT BENCHMARK
    PYO     PY AT BENCHMARK
    LXO     LABOR IN SECTOR X;

```

```

XO = 1;
YO = 1;
PXO = (1/2);

```

```

PYO = 1;
LXO = 1;

$ONTEXT

$MODEL:DEMAND

$SECTORS:
    X      ! ACTIVITY LEVEL FOR X = DEMAND FOR GOOD X
    Y      ! ACTIVITY LEVEL FOR Y = DEMAND FOR GOOD Y

$COMMODITIES:
    PX     ! PRICE OF X WHICH WILL EQUAL PL
    PY     ! PRICE OF Y WHICH WILL EQUAL 2 PL
    PL     ! PRICE OF FACTOR L

$CONSUMERS:
    RA     ! REPRESENTATIVE AGENT INCOME

$PROD:X
    O:PX   Q:XO
    I:PL   Q:LXO

$PROD:Y
    O:PY   Q:YO
    I:PL   Q:2

$DEMAND:RA    s:1
    E:PL      Q:120
    D:PX      Q:XO      P:PXO
    D:PY      Q:YO      P:PYO

$OFFTEXT

$SYSINCLUDE mpsgeset DEMAND

$INCLUDE DEMAND.GEN
SOLVE DEMAND USING MCP;

```

The solution listing appears as follows:

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR X	.	40.000	+INF	.
---- VAR Y	.	40.000	+INF	.
---- VAR PX	.	1.000	+INF	.
---- VAR PY	.	2.000	+INF	.
---- VAR PL	.	1.000	+INF	.

```

---- VAR RA          .          120.000    +INF          .

X          ACTIVITY LEVEL FOR X = DEMAND FOR GOOD X
Y          ACTIVITY LEVEL FOR Y = DEMAND FOR GOOD Y
PX         PRICE OF X WHICH WILL EQUAL PL
PY         PRICE OF Y WHICH WILL EQUAL 2 PL
PL         PRICE OF FACTOR L
RA         REPRESENTATIVE AGENT INCOME

```

Exercise (1.i): Use a different calibration point where  $x = 2$ ,  $y = 1$ ,  $MRS(2, 1) = 1/4$ . Rerun the model to show that this does not change the results.

```

XO = 2;
YO = 1;
PXO = (1/4);
PYO = 1;

$INCLUDE DEMAND.GEN
SOLVE DEMAND USING MCP;

```

Solution listing for exercise (1.i): Different calibration point.

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR X	.	40.000	+INF	.
---- VAR Y	.	40.000	+INF	.
---- VAR PX	.	0.500	+INF	.
---- VAR PY	.	2.000	+INF	.
---- VAR PL	.	1.000	+INF	.
---- VAR RA	.	120.000	+INF	.

Exercise (1.ii): Increase the price of  $x$  from 1 to 2, by changing the Q: coefficient for PL in sector  $x$  from 1 to 2.

```

LXO = 2;

$INCLUDE DEMAND.GEN
SOLVE DEMAND USING MCP;

```

The solution listing appears as follows:

	LOWER	LEVEL	UPPER	MARGINAL
--	-------	-------	-------	----------

```

---- VAR X      .      20.000    +INF      .
---- VAR Y      .      40.000    +INF      .
---- VAR PX     .       1.000    +INF      .
---- VAR PY     .       2.000    +INF      .
---- VAR PL     .       1.000    +INF      .
---- VAR RA     .     120.000    +INF      .

```

Exercise (1.iii): Compute an equilibrium in which commodity  $Y$  is defined as the numeraire.

This is done by fixing the value of  $P_y$  equal to one. The MPSGE function evaluation program detects that a price has been fixed, and it then does not apply any price normalization.

```

PY.FX = 1;
$INCLUDE DEMAND.GEN
SOLVE DEMAND USING MCP;

```

Solution listing for exercise (1.iii): Commodity  $Y$  as numeraire.

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR X	.	20.000	+INF	.
---- VAR Y	.	40.000	+INF	.
---- VAR PX	.	0.500	+INF	.
---- VAR PY	1.000	1.000	1.000	EPS
---- VAR PL	.	0.500	+INF	.
---- VAR RA	.	60.000	+INF	.

SOLUTION TO EXAMPLE 2

```
$TITLE: Solution to Example 2
```

```
*      Declare the benchmarck prices as scalars to allow for changes in the
*      counterfactual models.
```

```
SCALAR
```

```

X      QUANTITY OF X FOR WHICH THE MRS IS TO BE EVALUATED
Y      QUANTITY OF Y FOR WHICH THE MRS IS TO BE EVALUATED
MRS    COMPUTED MARGINAL RATE OF SUBSTITUTION
PX0    PRICE OF X
PY0    PRICE OF Y;

```

```

X = 1;
Y = 1;
PX0 = 1/4;

```

PYO = 1;

\$ONTEXT

\$MODEL:MRSCAL

\$COMMODITIES:

PX ! PRICE INDEX FOR GOOD X  
 PY ! PRICE INDEX FOR GOOD Y

\$CONSUMERS:

RA ! REPRESENTATIVE AGENT

\$DEMAND:RA s:1

D:PX Q:1 P:PX0  
 D:PY Q:1 P:PY0  
 E:PX Q:X  
 E:PY Q:Y

\$OFFTEXT

\$SYSINCLUDE mpsgeset MRSCAL

\$INCLUDE MRSCAL.GEN

SOLVE MRSCAL USING MCP;

\* Compute the MRS using the solution values:

MRS = PX.L / PY.L;

\* Display MRS with 8 decimals:

OPTION MRS:8;

DISPLAY MRS;

The solution listing appears as follows:

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR PX	.	0.400	+INF	.
---- VAR PY	.	1.600	+INF	.
---- VAR RA	.	2.000	+INF	.

PX PRICE INDEX FOR GOOD X

PY PRICE INDEX FOR GOOD Y

RA REPRESENTATIVE AGENT

---- 191 PARAMETER MRS = 0.25000000 COMPUTED MARGINAL

## RATE OF SUBSTITUTION

Exercise (2.i): Show that the demand function is homothetic by uniform scaling of the  $X$  and  $Y$  endowments. The resulting  $MRS$  should remain unchanged.

```
X = 20;
Y = 20;
```

```
$INCLUDE MRSCAL.GEN
SOLVE MRSCAL USING MCP;
```

```
*      Compute the MRS using the new solution values:
```

```
MRS = PX.L / PY.L;
```

```
DISPLAY MRS;
```

The solution to exercise (1.i):

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR PX	.	0.400	+INF	.
---- VAR PY	.	1.600	+INF	.
---- VAR RA	.	40.000	+INF	.

---- 255 PARAMETER MRS = 0.25000000 COMPUTED MARGINAL  
RATE OF SUBSTITUTION

Exercise (2.ii): Modify the demand function calibration point so that the reference price of both  $X$  and  $Y$  equal unity.

```
PX0 = 1;
PY0 = 1;
```

```
$INCLUDE MRSCAL.GEN
SOLVE MRSCAL USING MCP;
```

```
*      Compute the MRS using solution values:
```

```
MRS = PX.L / PY.L;
```

```
DISPLAY MRS;
```

Solution to exercise (2.ii):

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR PX	.	1.000	+INF	.
---- VAR PY	.	1.000	+INF	.
---- VAR RA	.	40.000	+INF	.

---- 319 PARAMETER MRS = 1.00000000 COMPUTED MARGINAL  
RATE OF SUBSTITUTION

### SOLUTION TO EXAMPLE 3

\$TITLE: Solution to Example 3

\* Declare as scalars EXO and SIGMA to change their values in the  
\* counterfactual models

SCALAR

PHI AGGREGATE LABOR PRODUCTIVITY /1/  
CX COST OF X AT BASE YEAR PRODUCTIVITY /1/  
CY COST OF Y AT BASE YEAR PRODUCTIVITY /1/  
EXO ENDOWMENT OF X AT BENCHMARK /0/  
SIGMA ELASTICITY OF SUBSTITUTION IN CONSUMPTION /1/;

\$ONTEXT

\$MODEL: LSUPPLY

\$SECTORS:

X ! SUPPLY = DEMAND FOR X  
Y ! SUPPLY = DEMAND FOR Y  
LS ! LABOR SUPPLY

\$COMMODITIES:

PX ! MARKET PRICE OF GOOD X  
PY ! MARKET PRICE OF GOOD Y  
PL ! MARKET WAGE  
PLS ! CONSUMER VALUE OF LEISURE

\$CONSUMERS:

RA ! REPRESENTATIVE AGENT

\$PROD:LS

O:PL Q:PHI

```

      I:PLS   Q:1

$PROD:X
      O:PX   Q:1
      I:PL   Q:CX

$PROD:Y
      O:PY   Q:1
      I:PL   Q:CY

*      Add endowment of X (EXO) which in benchmark is 0, so we can change
*      its value in exercise (a).

$DEMAND:RA  s:sigma
      E:PLS   Q:120
      E:PX    Q:EXO
      D:PLS   Q:1   P:1
      D:PX    Q:1   P:1
      D:PY    Q:1   P:1

$OFFTEXT
$SYSINCLUDE mpsgeset LSUPPLY
$INCLUDE LSUPPLY.GEN
SOLVE LSUPPLY USING MCP;

*      Save the solution value of labor supply with PHI=1, to compare with
*      the labor supply in the counterfactual model with PHI = 1.01.

SCALAR
      LSO      REFERENCE LEVEL OF LABOR SUPPLY
      ELS      ELASTICITY OF LABOR SUPPLY WRT REAL WAGE;

LSO = LS.L;

*      Counterfactual Model:  Modify Labor Productivity.

PHI = 1.01;

$INCLUDE LSUPPLY.GEN
SOLVE LSUPPLY USING MCP;

*      Compute the labor supply elasticity:

ELS = round(100 * (LS.L - LSO) / LSO);
DISPLAY ELS;

```

Exercise (3.i): To get a labor supply elasticity different than zero in a model with Cobb-Douglas final demand, include income from some other source different than labor income. Let the consumer be endowed with good  $X$  in addition to labor. Find the  $X$  endowment consistent with a labor supply elasticity of 0.15.

Solving for  $E_X$  in the expression for  $\eta$  given in the hint, we have that:

$$E_X = (\eta/(1 + \eta))L((1 - \theta_\ell)/\theta_\ell) * (P_L/P_x)$$

To compute  $E_X$  in this exercise, we have the benchmark values:  $L = 120$ ,  $\theta_\ell = 1/3$  (from the solution of the model at benchmark:  $LS = 80$ ). We are given:  $\eta = 0.15$ ,  $P_\ell = 1.01$  (from the change in PHI).

\$ontext

\$offtext

\* Compute EX0:

```
SCALARS      ETA      Elasticity of labor supply /0.15/
              L        Endowment of labor /120/
              Prl      Price of labor /1.01/
              Prx      Price of commodity X /1.0/
              Shl      Value share of leisure;
```

Shl = 1/3;

\* X endowment consistent with ETA = 0.15:

```
EX0 = (ETA/(1+ETA)) * L * ((1-Shl)/Shl) * (Prl/Prx);
```

DISPLAY EX0;

\* Return to the benchmark labor productivity:

PHI = 1;

```
$INCLUDE LSUPPLY.GEN
SOLVE LSUPPLY USING MCP;
```

The solution listing appears as follows:

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR X	.	40.000	+INF	.
---- VAR Y	.	40.000	+INF	.
---- VAR LS	.	80.000	+INF	.
---- VAR PX	.	1.000	+INF	.

```

---- VAR PY      .      1.000    +INF      .
---- VAR PL      .      1.000    +INF      .
---- VAR PLS     .      1.000    +INF      .
---- VAR RA      .     120.000    +INF      .

```

```

X      SUPPLY = DEMAND FOR X
Y      SUPPLY = DEMAND FOR Y
LS     LABOR SUPPLY
PX     MARKET PRICE OF GOOD X
PY     MARKET PRICE OF GOOD Y
PL     MARKET WAGE
PLS    CONSUMER VALUE OF LEISURE
RA     REPRESENTATIVE AGENT

```

\* Save the value of labor supply with PHI = 1;

```
LS0 = LS.L;
```

\* Counterfactual model:

```
PHI = 1.01;
```

```
$INCLUDE LSUPPLY.GEN
SOLVE LSUPPLY USING MCP;
```

\* Compute the labor supply elasticity with the solution values:

```
ELS = 100 * (LS.L - LS0) / LS0;
DISPLAY ELS;
```

Counterfactual model: PHI=1.01:

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR X	.	19.322	+INF	.
---- VAR Y	.	50.939	+INF	.
---- VAR LS	.	69.565	+INF	.
---- VAR PX	.	0.990	+INF	.
---- VAR PY	.	0.990	+INF	.
---- VAR PL	.	0.990	+INF	.
---- VAR PLS	.	1.000	+INF	.
---- VAR RA	.	151.304	+INF	.
---- 646 PARAMETER ELS =		0.150	ELASTICITY OF LABOR	

## SUPPLY WRT REAL WAGE

Exercise (3.ii) Alternatively, change the utility function. Let us change `s:1` to `s:SIGMA`, where `SIGMA` is a scalar value representing the benchmark elasticity of substitution between  $X$ ,  $Y$  and  $L$  in final demand. Find the value for `SIGMA` consistent with a labor supply elasticity of 0.15

Solving for  $\sigma$  in the expression for  $\eta$  given in the hint, we have that:

$$\sigma = \eta \frac{LSUP}{LEIS} \frac{1}{1 - \theta_\ell} + 1$$

To compute  $\sigma$  in this exercise we are given:  $\eta = 0.15$ ,  $\theta_\ell = 1/3$ , change in `PHI` from 1.0 to 1.01, thus with  $\eta = 0.15$  we have `LSUP` = 80.12 and `LEIS` = 39.88.

\* Compute `SIGMA`:

```
SCALAR  LSUP      Labor supply /80.12/
        LEIS      Demand for leisure /39.88/;
```

\* Value of `SIGMA` consistent with `ELS` = 0.15:

```
SIGMA =  ETA * (LSUP/LEIS) * (1/(1-Shl)) + 1;
DISPLAY SIGMA;
```

\* Return to the benchmark value of the endowment of  $X$  and labor prod:

```
EXO = 0;
PHI = 1;
```

```
$INCLUDE LSUPPLY.GEN
SOLVE LSUPPLY USING MCP;
```

Listing output:

```
----- 679 PARAMETER SIGMA = 1.452 ELASTICITY OF
                                         SUBSTITUTION IN
                                         CONSUMPTION

          LOWER    LEVEL    UPPER    MARGINAL
----- VAR X          .    40.000    +INF          .
----- VAR Y          .    40.000    +INF          .
----- VAR LS         .    80.000    +INF          .
----- VAR PX         .     1.000    +INF          .
----- VAR PY         .     1.000    +INF          .
----- VAR PL         .     1.000    +INF          .
```

```

---- VAR PLS      .          1.000    +INF      .
---- VAR RA       .          120.000   +INF      .

```

Continuing, we compute an equilibrium to evaluate the labor supply elasticity:

```
*      Save the value of labor supply with PHI = 1;
```

```
LS0 = LS.L;
```

```
*      Counterfactual model:
```

```
PHI = 1.01;
```

```
$INCLUDE LSUPPLY.GEN
SOLVE LSUPPLY USING MCP;
```

```
*      Compute the labor supply elasticity:
```

```
ELS = 100 * (LS.L - LS0) / LS0;
DISPLAY ELS;
```

Model output is as follows:

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR X	.	40.461	+INF	.
---- VAR Y	.	40.461	+INF	.
---- VAR LS	.	80.120	+INF	.
---- VAR PX	.	0.990	+INF	.
---- VAR PY	.	0.990	+INF	.
---- VAR PL	.	0.990	+INF	.
---- VAR PLS	.	1.000	+INF	.
---- VAR RA	.	120.000	+INF	.

---- 888 PARAMETER ELS = 0.150 ELASTICITY OF LABOR

Counterfactual model (Labor productivity = 1.01):

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR X	.	40.400	+INF	.
---- VAR Y	.	40.400	+INF	.
---- VAR LS	.	80.000	+INF	.
---- VAR PX	.	0.990	+INF	.
---- VAR PY	.	0.990	+INF	.

```

---- VAR PL      .          0.990    +INF      .
---- VAR PLS     .          1.000    +INF      .
---- VAR RA      .         120.000    +INF      .

```

```

---- 402 PARAMETER ELS = 0.000 ELASTICITY OF LABOR
      SUPPLY WRT REAL WAGE

```

Exercise (a): Consumer is endowed with good  $X$  and labor ( $\text{PHI}=1.0$ ):

```

---- 442 PARAMETER EXO = 31.617 ENDOWMENT OF X AT
      BENCHMARK

```

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR X	.	18.922	+INF	.
---- VAR Y	.	50.539	+INF	.
---- VAR LS	.	69.461	+INF	.
---- VAR PX	.	0.998	+INF	.
---- VAR PY	.	0.998	+INF	.
---- VAR PL	.	0.998	+INF	.
---- VAR PLS	.	0.998	+INF	.
---- VAR RA	.	151.304	+INF	.

#### SOLUTION TO EXAMPLE 4

\$TITLE: Solution to Example 4

SCALAR	XA	AGENT A ENDOWMENT OF X ( $0 < XA < 1$ )	/0.2/
	YA	AGENT A ENDOWMENT OF Y ( $0 < YA < 1$ )	/0.8/
	THETA_A	AGENT A DEMAND SHARE PARAMETER FOR X	/0.5/
	THETA_B	AGENT B DEMAND SHARE PARAMETER FOR X	/0.8/
	SIGMA_A	AGENT A ELASTICITY PARAMETER	/2.0/
	SIGMA_B	AGENT B ELASTICITY PARAMETER	/0.5/;

\$ONTEXT

\$MODEL: EXCHANGE

\$COMMODITIES:

```

PX      ! EXCHANGE PRICE OF GOOD X
PY      ! EXCHANGE PRICE OF GOOD Y

```

\$CONSUMERS:

```

A      ! CONSUMER A
B      ! CONSUMER B

```

```

$DEMAND:A s:SIGMA_A
    E:PX    Q:XA
    E:PY    Q:YA
    D:PX    Q:THETA_A
    D:PY    Q:(1-THETA_A)

$DEMAND:B s:SIGMA_B
    E:PX    Q:(1-XA)
    E:PY    Q:(1-YA)
    D:PX    Q:THETA_B
    D:PY    Q:(1-THETA_B)

*      Recover the solution value of the demands of both consumers to
*      construct the "efficiency locus" in exercise (c):
$REPORT:
    V:XAD   D:PX   DEMAND:A
    V:YAD   D:PY   DEMAND:A
    V:XBD   D:PX   DEMAND:B
    V:YBD   D:PY   DEMAND:B

$OFFTEXT
$SYSINCLUDE mpsgeset EXCHANGE

$INCLUDE EXCHANGE.GEN
SOLVE EXCHANGE USING MCP;

*      Compute the price and income ratios:
SCALAR
    PRATIO EQUILIBRIUM PRICE OF X IN TERMS OF Y
    IRATIO EQUILIBRIUM RATIO OF INCOMES;

PRATIO = PX.L / PY.L;
IRATIO = A.L / B.L;
DISPLAY IRATIO, PRATIO;

```

Solution listing and other displayed output for exercise 4:

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR PX	.	1.546	+INF	.
---- VAR PY	.	0.864	+INF	.
---- VAR A	.	1.000	+INF	.
---- VAR B	.	1.409	+INF	.
---- VAR XAD	.	0.232	+INF	.
---- VAR YAD	.	0.743	+INF	.
---- VAR XBD	.	0.768	+INF	.
---- VAR YBD	.	0.257	+INF	.

PX            EXCHANGE PRICE OF GOOD X

```

PY          EXCHANGE PRICE OF GOOD Y
A           CONSUMER A
B           CONSUMER B
XAD
YAD
XBD
YBD

```

```

----- 281 PARAMETER IRATIO = 0.709 EQUILIBRIUM RATIO OF
        CONSUMER INCOMES

```

```

        PARAMETER PRATIO = 1.790 EQUILIBRIUM PRICE OF
X IN TERMS OF Y

```

Compute an alternative efficient equilibrium where the income levels for *A* and *B* are equal, to demonstrate that when incomes are both fixed, the equilibrium remains efficient, but the connection between market prices and endowment income is eliminated:

```

*      Fix the income variables using: variable.FX = value:

```

```

A.FX = 1;
B.FX = 1;

```

```

$INCLUDE EXCHANGE.GEN
SOLVE EXCHANGE USING MCP;

```

```

SCALAR TRANSFER      IMPLIED TRANSFER FROM A TO B AS A PERCENTAGE OF INCOME;

```

```

TRANSFER = 100 * (A.L - (PX.L * XA + PY.L * YA));
PRATIO = PX.L / PY.L;
IRATIO = A.L / B.L;

```

```

DISPLAY TRANSFER, PRATIO, IRATIO;

```

Listing file output is as follows:

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR PX	.	1.223	+INF	.
---- VAR PY	.	0.777	+INF	.
---- VAR A	1.000	1.000	1.000	EPS
---- VAR B	1.000	1.000	1.000	EPS
---- VAR XAD	.	0.318	+INF	.
---- VAR YAD	.	0.786	+INF	.
---- VAR XBD	.	0.682	+INF	.

```

---- VAR YBD          .          0.214      +INF          .

----   396 PARAMETER TRANSFER = 13.351 IMPLIED TRANSFER
FROM A TO B AS A
PERCENTAGE OF INCOME
      PARAMETER PRATIO   = 1.572 EQUILIBRIUM PRICE OF
      X IN TERMS OF Y
      PARAMETER IRATIO   = 1.000 EQUILIBRIUM RATIO OF
CONSUMER INCOMES

```

Exercise (4.i): Set up separate models which compute autarchy price ratios for consumers A and B.

\* Remove the fixed values of incomes:

```

A.LO = 0;
A.UP = +INF;
B.LO = 0;
B.UP = +INF;

```

\$ONTEXT

\$MODEL: AUTAR\_A

\$COMMODITIES:

```

      PX      ! AUTAR_A PRICE OF GOOD X
      PY      ! AUTAR_A PRICE OF GOOD Y

```

\$CONSUMERS:

```

      A      ! CONSUMER A

```

\$DEMAND: A s: SIGMA\_A

```

      E: PX    Q: XA
      E: PY    Q: YA
      D: PX    Q: THETA_A
      D: PY    Q: (1-THETA_A)

```

\$OFFTEXT

\$SYSINCLUDE mpsgeset AUTAR\_A

\$INCLUDE AUTAR\_A.GEN

SOLVE AUTAR\_A USING MCP;

Solution listing for agent A autarchy equilibrium:

```

          LOWER      LEVEL      UPPER      MARGINAL
---- VAR PX          .          1.667      +INF          .

```

```

---- VAR PY          .          0.833    +INF      .
---- VAR A           .          1.000    +INF      .

```

Store autarchy equilibrium price ratio for agent *A*, then specify the autarchy equilibrium for agent *B* and compute a second equilibrium:

\* Compute the price ratio:

```

PARAMETER          PRICER          PRICE RATIO OF THE AUTARCHY MODELS;
PRICER("AUT_A", "PRATIO") = PX.L / PY.L;

```

\$ONTEXT

\$MODEL: AUTAR\_B

\$COMMODITIES:

```

      PX      ! AUTAR_B PRICE OF GOOD X
      PY      ! AUTAR_B PRICE OF GOOD Y

```

\$CONSUMERS:

```

      B      ! CONSUMER B

```

\$DEMAND: B s: SIGMA\_B

```

      E: PX    Q: (1-XA)

```

```

      E: PY    Q: (1-YA)

```

```

      D: PX    Q: THETA_B

```

```

      D: PY    Q: (1-THETA_B)

```

\$OFFTEXT

\$SYSINCLUDE mpsgeset AUTAR\_B

\$INCLUDE AUTAR\_B.GEN

SOLVE AUTAR\_B USING MCP;

\* Compute the price ratio:

```

PRICER("AUT_B", "PRATIO") = PX.L / PY.L;
DISPLAY PRICER;

```

Listing file output with a solution for model AUTAR\_B and a comparison report of autarchy price ratios:

```

                LOWER    LEVEL    UPPER    MARGINAL
---- VAR PX          .      1.000    +INF      .

```

```

---- VAR PY          .          1.000    +INF      .
---- VAR B           .          1.000    +INF      .

----      746 PARAMETER PRICER          PRICE RATIO OF THE AUTARCHY MODELS

          PRATIO
AUT_A     2.000
AUT_B     1.000

```

Exercise (4.ii): Determine parameter values for which the endowment point is equilibrium point.

\* Change preferences of A to be the same as his/her endowment:

```
THETA_A = 0.2;
```

```
$INCLUDE EXCHANGE.GEN
SOLVE EXCHANGE USING MCP;
```

\* Compute the price and income ratios:

```
PRATIO = PX.L / PY.L;
IRATIO = A.L / B.L;
DISPLAY PRATIO, IRATIO;
```

Listing file output for exercise 4.ii solution:

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR PX	.	1.000	+INF	.
---- VAR PY	.	1.000	+INF	.
---- VAR A	.	1.000	+INF	.
---- VAR B	.	1.000	+INF	.
---- VAR XAD	.	0.200	+INF	.
---- VAR YAD	.	0.800	+INF	.
---- VAR XBD	.	0.800	+INF	.
---- VAR YBD	.	0.200	+INF	.

```

----      856 PARAMETER PRATIO = 1.000 EQUILIBRIUM PRICE OF
          X IN TERMS OF Y
          PARAMETER IRATIO = 1.000 EQUILIBRIUM RATIO OF
          CONSUMER INCOMES

```

Exercise (4.iii): Set up a series of computations from which you can sketch the efficiency locus.

```

*      Return to the original value of the parameters:

THETA_A = 0.5;

*      Loop using different values of endowments of X and extract the solution
*      value of the demands:

SET          SC          SCENARIOS /SC1*SC7/;

PARAMETER    XAVALUE(SC)  VALUE SHARE OF "A" ENDOWMENT OF X
              /SC1 0,    SC2 0.1, SC3 0.3, SC4 0.5,
              SC5 0.7, SC6 0.9, SC7 1/
              DEMAND(SC,*) DEMAND BY SCENARIO;

LOOP (SC,
*      Install a XA value for the current scenario:
      XA = XAVALUE(SC);

$INCLUDE EXCHANGE.GEN
SOLVE EXCHANGE USING MCP;

*      Extract the solution value of the demands:
DEMAND(SC, "XADEM") = XAD.L;
DEMAND(SC, "YADEM") = YAD.L;
DEMAND(SC, "XBDEM") = XBD.L;
DEMAND(SC, "YBDEM") = YBD.L;
);

DISPLAY XAVALUE, DEMAND;

```

Listings from the comparison (solution listings have been suppressed):

```

----- 984 PARAMETER XAVALUE  VALUE SHARE OF "A" ENDOWMENT OF X
SC2 0.100, SC3 0.300, SC4 0.500, SC5 0.700, SC6 0.900, SC7 1.000

```

```

----- 984 PARAMETER DEMAND          DEMAND BY SCENARIO

          XADEM          YADEM          XBDEM          YBDEM
SC1      0.068          0.598          0.932          0.402
SC2      0.151          0.690          0.849          0.310
SC3      0.311          0.783          0.689          0.217
SC4      0.466          0.845          0.534          0.155
SC5      0.620          0.896          0.380          0.104

```

SC6	0.773	0.940	0.227	0.060
SC7	0.849	0.961	0.151	0.039

## SOLUTION TO EXAMPLE 5

\$TITLE: Solution to Example 5

```

SCALAR  XA          AGENT A ENDOWMENT OF X (0 < XA < 1)    /0.2/
        YA          AGENT A ENDOWMENT OF Y (0 < YA < 1)    /0.8/
        THETA_A     AGENT A DEMAND SHARE PARAMETER FOR X    /0.4/
        THETA_B     AGENT B DEMAND SHARE PARAMETER FOR X    /0.6/
        SIGMA_A     AGENT A ELASTICITY PARAMETER            /1/
        SIGMA_B     AGENT B ELASTICITY PARAMETER            /1/
        T_A         AD VALOREM TARIFF ON IMPORTS TO AGENT A /0.10/
        T_B         AD VALOREM TARIFF ON IMPORTS TO AGENT B /0.10/;

```

\$ONTEXT

\$MODEL:TARIFFS

\$SECTORS:

```

MXA    ! TRADE IN X FROM B TO A
MXB    ! TRADE IN X FROM A TO B
MYA    ! TRADE IN Y FROM B TO A
MYB    ! TRADE IN Y FROM A TO B

```

\$COMMODITIES:

```

PXA    ! PRICE OF GOOD X FOR AGENT A
PYA    ! PRICE OF GOOD Y FOR AGENT A
PXB    ! PRICE OF GOOD X FOR AGENT B
PYB    ! PRICE OF GOOD Y FOR AGENT B

```

\$CONSUMERS:

```

A      ! CONSUMER A
B      ! CONSUMER B

```

```

$DEMAND:A      s: SIGMA_A
                E: PXA   Q: XA
                E: PYA   Q: YA
                D: PXA   Q: THETA_A
                D: PYA   Q: (1-THETA_A)

```

```

$DEMAND:B      s: SIGMA_B
                E: PXB   Q: (1-XA)
                E: PYB   Q: (1-YA)

```

D:PXB Q:THETA\_B  
 D:PYB Q:(1-THETA\_B)

\* Trade activities

\$PROD:MXA

O:PX A Q:1  
 I:PX B Q:1 A:A T:T\_A

\$PROD:MXB

O:PX B Q:1  
 I:PX A Q:1 A:B T:T\_B

\$PROD:MYA

O:PY A Q:1  
 I:PY B Q:1 A:A T:T\_A

\$PROD:MYB

O:PY B Q:1  
 I:PY A Q:1 A:B T:T\_B

\* Recover a Hicksian money-metric welfare index:

\$REPORT:

V:WA W:A  
 V:WB W:B

\$OFFTEXT

\$SYSINCLUDE mpsgeset TARIFFS

\$INCLUDE TARIFFS.GEN

SOLVE TARIFFS USING MCP;

The solution listing for this model appears as follows:

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR MXA	.	0.177	+INF	.
---- VAR MXB	.	.	+INF	0.243
---- VAR MYA	.	.	+INF	0.243
---- VAR MYB	.	0.177	+INF	.
---- VAR PXA	.	1.272	+INF	.
---- VAR PYA	.	1.156	+INF	.
---- VAR PXB	.	1.156	+INF	.
---- VAR PYB	.	1.272	+INF	.
---- VAR A	.	1.200	+INF	.
---- VAR B	.	1.200	+INF	.
---- VAR WA	.	0.999	+INF	.
---- VAR WB	.	0.999	+INF	.

```

MXA      TRADE IN X FROM B TO A
MXB      TRADE IN X FROM A TO B
MYA      TRADE IN Y FROM B TO A
MYB      TRADE IN Y FROM A TO B
PXA      PRICE OF GOOD X FOR AGENT A
PYA      PRICE OF GOOD Y FOR AGENT A
PXB      PRICE OF GOOD X FOR AGENT B
PYB      PRICE OF GOOD Y FOR AGENT B
A        CONSUMER A
B        CONSUMER B
WA
WB

```

Now compute a counterfactual equilibrium with zero tariffs in both countries:

```

SCALAR  WAO      BENCHMARK WELFARE INDEX FOR AGENT A
        WBO      BENCHMARK WELFARE INDEX FOR AGENT B;

WAO = WA.L;
WBO = WB.L;

*      Counterfactual model: Free Trade (remove tariffs)
T_A = 0;
T_B = 0;

$INCLUDE TARIFFS.GEN
SOLVE TARIFFS USING MCP;

*      Compute Hicksian equivalent variation:

SCALAR  EV_A     HICKSIAN EQUIVALENT VARIATION FOR AGENT A
        EV_B     HICKSIAN EQUIVALENT VARIATION FOR AGENT B;

EV_A = 100 * (WA.L - WAO)/WAO;
EV_B = 100 * (WB.L - WBO)/WBO;
DISPLAY EV_A, EV_B;

```

The solution listing for the Global Free Trade scenario are as follows:

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR MXA	.	0.200	+INF	.
---- VAR MXB	.	.	+INF	EPS
---- VAR MYA	.	.	+INF	EPS
---- VAR MYB	.	0.200	+INF	.
---- VAR PXA	.	1.179	+INF	.
---- VAR PYA	.	1.179	+INF	.

```

---- VAR PXB          .          1.179    +INF      .
---- VAR PYB          .          1.179    +INF      .
---- VAR A            .          1.179    +INF      .
---- VAR B            .          1.179    +INF      .
---- VAR WA           .          1.000    +INF      .
---- VAR WB           .          1.000    +INF      .

----      543 PARAMETER EV_A    = 0.108 HICKSIAN EQUIVALENT
      VARIATION FOR AGENT A
      PARAMETER EV_B    = 0.108 HICKSIAN EQUIVALENT
      VARIATION FOR AGENT B

```

Exercise (5.i): Find an "optimal tariff" in this model for agent *A*, assuming that Agent *B* does not retaliate and leaves her tariff rate at the benchmark level.

```

*      Return to original tariffs

T_A = 0.1;
T_B = 0.1;

*      Loop using different tariff rates and extract welfare index to
*      compute the Hicksian equivalent variation:

SET      SC      Scenarios /SC1*SC7/;

PARAMETER TVALUE(SC)      TARIFF VALUE FOR SC
                          /SC1 0,   SC2 0.2, SC3 0.4, SC4 0.6,
                          SC5 0.8, SC6 1.0, SC7 1.2/
                          SUMMARY(SC,*) HICKSIAN EQUIVALENT VARIATION BY SCENARIO;

LOOP(SC,

*      Install a tariff rate imposed by Agent A in the current scenario:
*      (T_B remains at the benchmark level).

      T_A = TVALUE(SC);

$INCLUDE TARIFFS.GEN
SOLVE TARIFFS USING MCP;

*      Extract welfare index and compute Hicksian EV:
SUMMARY(SC,"EV_A") = 100 * (WA.L - WAO)/WAO;
SUMMARY(SC,"EV_B") = 100 * (WB.L - WBO)/WBO;
);

OPTION TVALUE:2;
DISPLAY TVALUE, SUMMARY;

```

Suppressing solution listings, the following summary report is generated:

```

----- 720 PARAMETER TVALUE          TARIFF VALUE FOR SC
SC2 0.20, SC3 0.40, SC4 0.60, SC5 0.80, SC6 1.00, SC7 1.20

```

```

----- 720 PARAMETER SUMMARY HICKSIAN EV BY SCENARIO

```

	EV_A	EV_B
SC1	-0.911	1.073
SC2	0.697	-0.939
SC3	1.614	-2.497
SC4	2.075	-3.730
SC5	2.226	-4.721
SC6	2.160	-5.528
SC7	1.939	-6.193

Exercise (5.ii): Insert the endowment and preference parameters from the previous problem, retaining the same "benchmark" tariff rates.

\* Use parameter values from Example 4:

```

XA = 0.2;
YA = 0.8;
THETA_A = 0.5;
THETA_B = 0.8;
SIGMA_A = 2.0;
SIGMA_B = 0.5;

```

\* Benchmark tariff rates from Example 5:

```

T_A = 0.1;
T_B = 0.1;

```

```

$INCLUDE TARIFFS.GEN
SOLVE tariffs USING MCP;

```

Equilibrium values for benchmark tariffs:

	LOWER	LEVEL	UPPER	MARGINAL
---- VAR MXA	.	0.024	+INF	.
---- VAR MXB	.	.	+INF	0.396
---- VAR MYA	.	.	+INF	0.236

```

---- VAR MYB      .      0.040    +INF      .
---- VAR PXA      .      2.076    +INF      .
---- VAR PYA      .      1.126    +INF      .
---- VAR PXB      .      1.887    +INF      .
---- VAR PYB      .      1.239    +INF      .
---- VAR A        .      1.320    +INF      .
---- VAR B        .      1.762    +INF      .
---- VAR WA       .      0.904    +INF      .
---- VAR WB       .      1.009    +INF      .

```

Compute free trade equilibrium and report equivalent variations in income:

```
*      Counterfactual model: Free Trade (remove tariffs)
```

```
T_A = 0;
```

```
T_B = 0;
```

```
$INCLUDE TARIFFS.GEN
```

```
SOLVE TARIFFS USING MCP;
```

```
EV_A = 100 * (WA.L - WAO)/WAO;
```

```
EV_B = 100 * (WB.L - WBO)/WBO;
```

```
DISPLAY EV_A, EV_B;
```

Free trade calculations with equivalent variation report:

```

          LOWER      LEVEL      UPPER      MARGINAL
---- VAR MXA      .      0.032    +INF      .
---- VAR MXB      .      .      +INF      2.220E-16
---- VAR MYA      .      .      +INF      EPS
---- VAR MYB      .      0.057    +INF      .
---- VAR PXA      .      1.928    +INF      .
---- VAR PYA      .      1.077    +INF      .
---- VAR PXB      .      1.928    +INF      .
---- VAR PYB      .      1.077    +INF      .
---- VAR A        .      1.247    +INF      .
---- VAR B        .      1.758    +INF      .
---- VAR WA       .      0.902    +INF      .
---- VAR WB       .      1.011    +INF      .

---- 1030 PARAMETER EV_A      =      -9.657 HICKSIAN EQUIVALENT
          VARIATION FOR AGENT A
          PARAMETER EV_B      =      1.242 HICKSIAN EQUIVALENT
          VARIATION FOR AGENT B

```

# Applied General Equilibrium Modeling with MPSGE as a GAMS Subsystem

*An Overview of the Modeling Framework and Syntax*

**Abstract.** This paper describes a programming environment for economic equilibrium analysis. The system introduces the Mathematical Programming System for General Equilibrium analysis (MPSGE, Rutherford 1987) within the Generalized Algebraic Modelling System (GAMS, Brooke, Kendrick and Meeraus (1988)). This arrangement exploits GAMS' set-oriented algebraic syntax for data manipulation and report writing. The system based on the tabular MPSGE input format provides a compact, non-algebraic representation of a model's nonlinear equations. This paper provides an overview of the modelling environment and three worked examples in tax policy analysis.

**Keywords:** Applied General Equilibrium

**JEL codes:** C68



## 1. INTRODUCTION

This paper introduces a programming language for economic equilibrium modelling. The paper presents the motivation for the system, the programming syntax, and three small scale examples. A library of larger models are provided with the program. The purpose of the paper is to provide a concise introduction to the modelling environment.

MPSGE is a language for concise representation of Arrow-Debreu economic equilibrium models (Rutherford, 1987). The name stands for “mathematical programming system for general equilibrium”. MPSGE provides a short-hand representation for the complicated systems of nonlinear inequalities which underly general equilibrium models. The MPSGE framework is based on nested constant elasticity of substitution utility functions and production functions. The data requirements for a model include share and elasticity parameters, endowments, and tax rates for all the consumers and production sectors included in the model. These may or may not be calibrated from a consistent benchmark equilibrium dataset.

GAMS, the “Generalized Algebraic Modelling System”, is a modeling language which was originally developed for linear, nonlinear and integer programming. This language was developed over 20 years ago by Alex Meeraus when he was working at the World Bank. (See Brooke, Kendrick and Meeraus 1988.) Since that time, GAMS has been widely applied for large-scale economic and operations research modeling projects.

Prior to their marriage, MPSGE and GAMS embodied different design philosophies. MPSGE was (and is) appropriate for a specific class of nonlinear equations, while GAMS is capable of representing any system of algebraic equations. While GAMS is applicable in several disciplines, MPSGE is only applicable in the analysis of economic equilibrium models. The expert knowledge embodied in MPSGE is of particular use to economists who are interested in the insights provided by formal models but who are unable to devote many hours to programming. MPSGE provides a structured framework for novice modellers. When used by experts, MPSGE reduces the setup cost of producing an operational model and the cost of testing alternative specifications.

In contrast, the GAMS modelling language is designed for managing large datasets. The use of sets and detached-coefficient matrix notation makes the GAMS environment very nice for both developing balanced benchmark datasets and for writing solution reports. GAMS’ main disadvantage for economic applications concerns the specification of the model structure. Economic equilibrium models, particularly those based on complicated functions such as nested constant-elasticity-of-substitution (CES), are easier to understand at an abstract level than they are to specify in detail, and the translation of a model from input data into algebraic relations can be a tedious and error-prone undertaking.

The interface between GAMS and MPSGE combines the strengths of both programs. The system uses GAMS as the “front end” and “back end” to MPSGE, facilitating data handling and report writing. The language employs an extended MPSGE syntax based on GAMS sets, so that model specification is concise. In addition, the system includes two large-scale solvers, MILES (Rutherford, 1993) and PATH (Ferris and Dirkse, 1993), which may be used interchangeably. The availability of two algorithms greatly enhances robustness and reliability.

The GAMS/MPSGE interface has been commercially available since 1993 and there are a number of published applications based on the software. Appendix A provides a

selected set of papers based on models programmed with GAMS/MPSGE. The range of issues which have been addressed testifies to the flexibility of the modeling environment.

Of course, GAMS/MPSGE is not the only software available for formulating and solving CGE models. GEMPACK (General Equilibrium Modelling PACKage, Harrison and Pearson (1996), Harrison, Pearson, Powell and Small (1994)) is a suite of general-purpose economic modelling software for equilibrium models represented as a mixture of linearized and level equations.

The remainder of this paper is organized as follows. Section 2 introduces MPSGE input syntax and the GAMS interface using a small two-sector model of taxation. Section 3 extends the 2x2 model to illustrate how the software is used to perform equal-yield (differential) tax policy analysis and to analyze tax reform in a model with endogenous taxation. Section 4 provides a brief summary and conclusion. The paper introduces language features largely through example. Details on language syntax and program structure are provided in two appendices. Appendix B provides a complete statement of MPSGE syntax. Appendix C provides an overview of the modeling environment and the structure of GAMS input files which employ MPSGE. Appendix D provides an algebraic specification of the three models considered in the paper.

If readers unfamiliar with GAMS wish to go through the examples, it would be helpful to install a demonstration system (see <http://www.gams.com> for details), then retrieve and process the library file THREEMGE which contains three MPSGE models (HARBERGER, SHOVEN and SAMUELSON). Once the GAMS system has been installed, two commands to retrieve and run the models used in this paper:

```
gamslib threemge
gams threemge
```

After having processed this file, print the listing file (THREEMGE.LST) for reference.

## 2. THE MATHEMATICAL FORMULATION

Mathiesen (1985) demonstrated that an Arrow-Debreu general economic equilibrium model could be formulated and efficiently solved as a complementarity problem. Mathiesen's formulation may be posed in terms of three sets of "central variables":

$\mathbf{p}$  = a non-negative  $n$ -vector of commodity prices including all final goods, intermediate goods and primary factors of production;

$\mathbf{y}$  = a non-negative  $m$ -vector of activity levels for constant returns to scale production sectors in the economy; and

$\mathbf{M}$  = an  $h$ -vector of income levels, one for each "household" in the model, including any government entities.

An equilibrium in these variables satisfies a system of three classes of nonlinear inequalities.

*Zero Profit*

The first class of constraint requires that in equilibrium no producer earns an “excess” profit, i.e. the value of inputs per unit activity must be equal to or greater than the value of outputs. This can be written in compact form as:

$$-\Pi_j(p) = C_j(p) - R_j(p) \geq 0 \quad \forall j$$

where  $\Pi_j(p)$  is the unit profit function, the difference between unit revenue and unit cost, defined as:

$$C_j(p) \equiv \min\left\{\sum_i p_i x_i \mid f_j(x) = 1\right\}$$

and

$$R_j(p) \equiv \max\left\{\sum_i p_i y_i \mid g_j(y) = 1\right\}$$

where  $f$  and  $g$  are the associated production functions characterizing feasible input and output. For example, if we have:

$$f(x) = \phi \prod_i x_i^{\alpha_i} \quad \sum_i \alpha_i = 1, \quad \alpha_i \geq 0$$

and

$$g(y) = \psi \max_i \frac{y_i}{\beta_i} \quad \beta_i \geq 0$$

then the dual functions will be:

$$C(p) = \frac{1}{\phi} \prod_i \left(\frac{p_i}{\alpha_i}\right)^{\alpha_i}$$

and

$$R(p) = \sum_i \beta_i p_i$$

*Market Clearance*

The second class of equilibrium conditions is that at equilibrium prices and activity levels, the supply of any commodity must balance or exceed excess demand by consumers. We can express these conditions as:

$$\sum_j y_j \frac{\partial \Pi_j(p)}{\partial p_i} + \sum_h \omega_{ih} \geq \sum_h d_{ih}(p, M_h)$$

in which the first sum, by Shepard’s lemma, represents the net supply of good  $i$  by the constant-returns to scale production sectors, the second sum represents the aggregate initial endowment of good  $i$  by households, and the sum on the right-hand-side represents aggregate final demand for good  $i$  by households, given market prices  $p$  and household income levels  $M$ .

Final demand are derived from budget-constrained utility maximization:

$$d_{ih}(p, M_h) = \operatorname{argmax} \{U_h(x) \mid \sum_i p_i x_i = M_h\}$$

in which  $U_h$  is the utility function for household  $h$ .

#### *Income Balance*

The third condition is that at an equilibrium, the value of each agent's income must equal the value of factor endowments:

$$M_h = \sum_i p_i \omega_{ih}$$

We always work with utility functions which exhibit non-satiation, so Walras' law will always hold:

$$\sum_i p_i d_{ih} = M_h = \sum_i p_i \omega_{ih}$$

Aggregating market clearance conditions using equilibrium prices and the zero profit conditions using equilibrium activity levels, it then follows that:

$$\sum_j y_j \Pi_j(p) = 0$$

or

$$y_j \Pi_j(p) = 0 \quad \forall j$$

Furthermore, it follows that:

$$p_i \left( \sum_j y_j \frac{\partial \Pi_j(p)}{\partial p_i} + \sum_h \omega_{ih} - \sum_h d_{ih}(p, M_h) \right) = 0 \quad \forall i$$

In other words, complementary slackness is a *feature* of the equilibrium allocation even though it is not imposed as an equilibrium condition, *per se*. This means that in equilibrium, any production activity which is operated makes zero profit and any production activity which earns a negative net return is idle. Likewise, any commodity which commands a positive price has a balance between aggregate supply and demand, and any commodity in excess supply has an equilibrium price of zero.

### 3. A SMALL EXAMPLE

This section of the paper introduces MPSGE model building using a two- good, two-factor (2x2) example. This is addressed to readers who may be unfamiliar with GAMS and/or the original (scalar) MPSGE syntax. The discussion provides some details on the formulation and specification of one small model from the MPSGE library. Subsequently, two extensions are presented, one which illustrates equal yield constraints and another which introduces a pure public good. These examples are by no means exhaustive of the

classes of equilibrium structures which can be investigated using the software, but they do provide a starting point for new users.

The structure of MPSGE model HARBERGER is “generic” Arrow-Debreu with taxes. Households obtain income by supplying factors of production to industry or collecting tax revenue. This income is then allocated between alternative goods in order to maximize welfare subject to the budget constraint.

Firms operate subject to constant returns to scale, selecting factor inputs in order to minimize cost subject to technological constraints. For an algebraic description of a model closely related to this one, see Shoven and Whalley (1984). The present model differs in two respects from the Shoven-Whalley example. First, in this model there are intermediate inputs to production while in the Shoven-Whalley model goods are produced using only value-added. Second, this model incorporates a labor-leisure choice so that the excess burden of factor taxes here incorporates the disincentive to work associated with a lower net wage.

### 3.1. BENCHMARK DATA

Figure 1 presents most of the input data for a two good, two factor, closed economy model. This is an economy in which, initially, taxes are levied only on capital inputs to production. We treat tax revenue as though it were returned lump-sum to the households.

	Sectors		Consumers		
	X	Y	OWNERS	WORKERS	GOVT
PX	100	-20	-30	-50	
PY	-10	80	-40	-30	
PK	-20	-40	60		
PL	-50	-10		60*	
TK	-20	-10			30
TRN			10	20	-30

\* 60 = labor endowment net leisure demand= 100 - 40

*Figure 1.* The Benchmark Social Accounts

The input data is presented in the form of a balanced matrix, the entries in which represent the value of economic transactions in a given period (typically one year). Social accounting matrices (SAMs) can be quite detailed in their representation of an economy, and they are also quite flexible. All sorts of inter-account taxes, subsidies and transfers can be represented through an appropriate definition of the accounts.

Traditionally, a SAM is square with an exact correspondence between rows and columns. (For an introduction, see Pyatt and Round, “Social Accounting Matrices: A Basis for Planning”, The World Bank, 1985.) The numbers which appear in a conventional SAM are typically positive, apart from very special circumstances, whereas the rectangular

SAM displayed in Figure 1 follows a sign convention wherein supplies or receipts are represented by positive numbers and demands or payments are represented by negative numbers. Internal consistency of a rectangular SAM implies that row sums and column sums are zero. This means that supply equals demand for all goods and factors, tax payments equal tax receipts, there are no excess profits in production, the value of each household expenditure equals the value of factor income plus transfers, and the value of government tax revenue equals the value of transfers to households.

With simple MPSGE models, it is convenient to use a *rectangular* SAM format. This format emphasizes how the MPSGE program structure is connected to the benchmark data. In the rectangular SAM, we have one row for every market (traded commodity), one column for every production sector, and one column for every consumer. Benchmark equilibrium is reflected in the row and column sums. Columns corresponding to sectors sum to zero reflecting zero excess profit (the value of output equals the cost of inputs). Columns corresponding to consumers sum to zero indicating income balance. Rows sum to zero indicating market clearance.

In the present model, there are four markets, for goods X and Y and factors L and K; two production sectors (X and Y) and three consumers (OWNERS, WORKERS and GOVT).

### 3.2. DATA ENTRY IN GAMS

Consider a generalized version of the model in which the set of production sectors be denoted  $S$  (here,  $S = \{X, Y\}$ ). Let the set of goods be  $G$ . Production sectors are mapped one-to-one with the goods, so we see that sets  $S$  and  $G$  are in fact the same set. Let  $F$  denote the set of primary factors (here,  $F = \{L, K\}$ ), and let  $H$  denote the set of households (here  $H = \{OWNER, WORKER\}$ ).

Now that we have identified the underlying sets, we may interpret the input matrix as a set of parameters with which we can easily specify the benchmark equilibrium. (See Figure 2.) It is quite common to begin a general equilibrium modelling project with a large input-output Figure or social accounting matrix which may then be mapped onto a number of submatrices, each of which is dimensioned according to the underlying sets used in the model.

It should be emphasized at this point that there is an enormous difference between Figures 1 and 2. While Figure 1 concisely represents an equilibrium for a single two-sector model, Figure 2 represents an arbitrary set of commodities, factors, production sectors and households. *Scalability* is an enormous advantage of the GAMS/MPSGE modeling environment. Model implementation and debugging can be conducted on a tiny prototype, and then the same computer code can be employed with a large-scale dataset.

The GAMS specification of benchmark data is presented in Figure 3 which begins with a statement of the underlying sets ( $G$ ,  $F$ ,  $H$ ). The statement “`ALIAS (S,G);`” simply says that  $S$  and  $G$  both reference  $\{X, Y\}$ . Thereafter follows the social accounting data table and declarations for the various submatrices. The parameters `ELAS()` and `ESUB()` are elasticities (“free parameters”) which can be chosen independently from the benchmark accounts. The parameters `TF` and `PF` are calibrated tax and reference price arrays which are computed given benchmark factor and tax payments. (In this model, average and marginal tax rates are not distinguished, so the benchmark marginal tax rate is simply the tax payment divided by the net factor income.)

	Sectors	Consumers	
	(S)	Households(H)	Government
Goods Markets (G):	A(G,S)-B(G,S)	-C(G,H)	
Factor Markets (F):	-FD(F,S)	E(F,H)-D(F,H)	
Capital taxes:	-T("K",S)		GREV
Transfers:		TRN(H)	-GREV

Figure 2. Mapping from Benchmark Social Accounts into Parameters

*	SECTION (i)	DATA SPECIFICATION AND BENCHMARKING			
SETS	G	GOODS AND SECTORS /X, Y/,			
	F	PRIMARY FACTORS /K, L/,			
	H	HOUSEHOLDS /OWNER, WORKER/;			
ALIAS (S,G);					
TABLE SAM(*,*) SOCIAL ACCOUNTING MATRIX					
	X	Y	OWNER	WORKER	GOVT
X	100	-20	-30	-50	
Y	-10	80	-40	-30	
K	-20	-40	60		
L	-50	-10		60	
TK	-20	-10			30
TRN			10	20	-30

Figure 3a. Set and Data Specifications for the 2x2 Model Harberger

A general equilibrium model determines only relative prices. For purposes of reporting or constructing value-indices, we use a Laspeyres quantity index,  $\text{THETA}(G)$ , the elements of which correspond to shares of aggregate consumer expenditure in the benchmark period.

### 3.3. MODEL SPECIFICATION

The MPSGE description of this model is shown in Figure 4. Declarations following the  $\$MODEL$  statement indicate that the model involves one class of production activities ( $AL(S)$ ), three classes of commodities ( $P(G)$ ,  $W(F)$  and  $PT$ ), and two types of consumers, private consumers ( $RA(H)$ ), and a government "consumer" ( $GOVT$ ).

One  $\$PROD$ : block describes the single class of production activities, and two  $\$DEMAND$ : blocks characterize endowments and preferences for the two classes of consumers.

## PARAMETER

A(S)	BENCHMARK OUTPUT
B(G,S)	USE MATRIX (GOODS INPUTS BY SECTOR)
C(G,H)	HOUSEHOLD DEMAND
FD(F,S)	FACTOR DEMAND BY SECTOR
E(F,H)	FACTOR ENDOWMENTS
D(F,H)	FACTOR DEMAND BY HOUSEHOLDS
T(F,S)	TAX PAYMENT BY FACTOR BY SECTOR
TRN(H)	TRANSFER REVENUE
ELAS(S)	ELASTICITY OF SUBSTITUTION IN PRODUCTION
ESUB(H)	ELASTICITY OF SUBSTITUTION IN DEMAND
GREV	BENCHMARK GOVERNMENT REVENUE
TF(F,S)	FACTOR TAX RATE
PF(F,S)	BENCHMARK FACTOR PRICES GROSS OF TAX
THETA(G)	WEIGHTS IN NUMERAIRE PRICE INDEX
WBAR(H)	BENCHMARK WELFARE INDEX;

Figure 3b. Parameter Declarations for the 2x2 Model Harberger

\* EXTRACT DATA FROM THE SOCIAL ACCOUNTING MATRIX:

```

A(S)  = SAM(S,S);           B(G,S) = MAX(0, -SAM(G,S));
C(G,H) = -SAM(G,H);        FD(F,S) = -SAM(F,S);
E(F,H) = SAM(F,H);        D(F,H) = 0;
T("K",S) = -SAM("TK",S);  TRN(H) = SAM("TRN",H);

```

\* INSTALL "FREE" ELASTICITY PARAMETERS:

```

E("L","WORKER") = 100;    D("L","WORKER") = 40;
ELAS(S) = 1;             ESUB(H) = 0.5;

```

\* INSTALL FUNCTIONS OF BENCHMARK VALUES:

```

GREV      = SUM(H, TRN(H));
TF(F,S) = T(F,S) / FD(F,S);
PF(F,S) = 1 + TF(F,S);
THETA(G) = SUM(H, C(G,H));
THETA(G) = THETA(G) / SUM(S, THETA(S));
WBAR(H) = SUM(G, C(G,H)) + SUM(F, D(F,H));

```

Figure 3c. Parameter Assignments for the 2x2 Model Harberger

Consider the records associated with production sector  $AL(S)$ . The entries on the first line of a  $\$PROD:$  block are elasticity values. The “s:0” field indicates that the top-level elasticity of substitution between inputs is zero (Leontief). The entry “va:ELAS(S)” indicates that inputs identified as belonging to the “va:” aggregate trade off with an elasticity of substitution  $ELAS(S)$  (at the second level of the production function). In these production functions, the primary factors ( $W(F)$ ) are identified as entering in the va: aggregate.

The nesting assignments within an MPSGE model can be easily modified by the assignment of inputs to aggregates, as has been done in this model using the va: nest. This flexibility is very helpful when testing robustness with respect to changes in the elasticity structure of a model. (A note on syntax: nesting identifiers are limited to four characters.)

The records within a  $\$PROD:$  block begin with “O:” or “I:”. An “O:” indicates an output and an “I:” represents an input. In both types of records, “Q:” is a “quantity field” indicating a reference input or output level of the named commodity. A “P:” signifies a reference price field. This price is measured as a user cost, gross of applicable taxes. The default values for reference price and reference quantity are both unity (i.e., a value of 1 is installed if a P: or Q: field is missing).

The va: and T: fields in a  $\$PROD:$  block indicate tax agent and ad-valorem tax rate, respectively. The tax agent is specified before the tax rate. A single input or output coefficient may have two or more taxes applied. Consumers are treated symmetrically, and there is thus no restriction on the consumer to whom the tax is paid. Typically, however, one consumer is associated with the government.

To better understand the relationship between reference prices and base year tax rates, consider inputs of  $W.K$  to sector  $AL.X$  in this model. The benchmark payment to capital in the  $X$  sector is 20 and the tax payment is 20. Hence the ad-valorem tax rate in the benchmark equilibrium is 100% ( T:1), and the reference price of capital, market price of unity times  $(1 + 100\%)$ , is 2 (P:2). If in a counterfactual experiment the tax rate on capital inputs to sector  $X$  is altered, this will change the T: field but it will not change the P: field. Q: and P: characterize a reference equilibrium point, and these are therefore unaffected by subsequent changes in the exogenous parameters.

It is important to remember that the  $\$PROD:AL(S)$  block represents as many individual production functions as there are elements in set  $S$  (two in this case). Within the  $\$PROD:AL(S)$  block, inputs refer to sets  $G$  and  $F$ , while the output coefficient,  $O:P(S)$ , refers only to set  $S$ . Sets referenced within a commodity name in an I: or O: field may be sets which are “controlled” by the sets referenced in the function itself, in which case only a scalar entry is produced. In  $\$PROD:AL(S)$  there are only outputs of commodity  $S$  in sector  $S$ .

The first line of a  $\$DEMAND$  block also contains fields (e.g., s:, gds:, c1: etc.) which represent elasticities of substitution. The subsequent records may begin with either an E: field or a D: field. These, respectively, represent commodity endowments and demands. In the demand fields, the P: and Q: entries are interpreted as reference price and reference quantity, analogous to the input fields in a  $\$PROD$  block. Ad-valorem taxes may not be applied on final demands, so that if consumption taxes are to be applied in a model they must be levied on production activities upstream of the final demand.

The benchmark values for all activity levels and prices are equal to the default value of unity, and therefore we are able to specify values in the Q: fields directly from the benchmark data. An equivalent model could be specified in which the benchmark activity levels for  $AL(S)$  equal, for example,  $A(S,S)$ . This would then require rescaling the input and output coefficients for those sectors, and it would not necessarily be helpful, because

```

*      SECTION (ii)      MPSGE MODEL DECLARATION

$ONTEXT

$MODEL:HARBERGER

$SECTORS:
      AL(S)

$COMMODITIES:
      P(G)  W(F)  PT

$CONSUMERS:
      RA(H)  GOVT

$PROD:AL(S)  s:0  va:ELAS(S)
      O:P(S)   Q:A(S)
      I:P(G)   Q:B(G,S)
      I:W(F)   Q:FD(F,S)  P:PF(F,S)  A:GOVT  T:TF(F,S)  va:

$DEMAND:RA(H)  s:1  gds:ESUB(H)
      D:P(G)   Q:C(G,H)  gds:
      D:W(F)   Q:D(F,H)
      E:W(F)   Q:E(F,H)
      E:PT     Q:TRN(H)

$DEMAND:GOVT
      D:PT     Q:GREV

$REPORT:
      V:CD(G,H)      D:P(G)      DEMAND:RA(H)
      V:DF(F,H)      D:W(F)      DEMAND:RA(H)
      V:EMPLOY(S)    I:W("L")    PROD:AL(S)
      V:WLF(H)      W:RA(H)

$OFFTEXT

*      Invoke the preprocessor to declare the model for GAMS:

$SYSINCLUDE mpsgeset HARBERGER

```

Figure 4a. MPSGE Model Declaration

```

* -----
* SECTION (iii) BENCHMARK REPLICATION

HARBERGER.ITERLIM = 0;
$INCLUDE HARBERGER.GEN
SOLVE HARBERGER USING MCP;
ABORT$(ABS(HARBERGER.OBJVAL) GT 1.E-4)
  "*** HARBERGER benchmark does not calibrate.";
HARBERGER.ITERLIM = 1000;

```

Figure 4b. MPSGE Model Specification and Benchmark Replication

in a scaled model it is more difficult to verify consistency of the benchmark accounts and MPSGE input file. Furthermore, for numerical reasons it is advisable to scale equilibrium values for the central variables to be close to unity.

Government transfers to households are accomplished through the use of an “artificial commodity” (PT). The government is identified as the agent who receives all tax revenue (see the A:GOVT entry in both of the \$PROD: blocks). Commodity PT is the only commodity on which GOVT spends this income, hence government tax revenue is divided between the two households in proportion to their endowments of the artificial good. In order to scale units so that the benchmark price of PT is unity, the \$30 of government tax revenue chases 10 units of PT assigned to OWNER and 20 units assigned to WORKER. (See values for TRN(H) in Figure 3.)

The \$REPORT section of the input file requests the solution system to return values for inputs, outputs, final demands or welfare indices at the equilibrium. Only those items which are requested will be written to the solution file. Each record in the report block begins with a V: (variable name) field. These names must be distinct from all other names in the model. The second field of the report record must have one of the labels I:, O: or D: followed by a commodity name, or the label W: followed by a consumer name. The third field’s label must be “PROD:” in an I: or O: record, and it must be “DEMAND:” if it is a D: record.

An algebraic formulation of the Harberger model is provided in Appendix D for the interested reader.

### 3.4. MPSGE FORMULATION: KEY IDEAS

There are two points regarding the MPSGE function format which are important yet easily misunderstood by new users:

(i) *The elasticities together with the reference quantities and reference prices of inputs and outputs completely characterize the underlying nested CES functions.* No other data fields in the \$PROD: block alters the technology. If, for example, a tax rate changes as part of a counter-factual experiment, this has no effect on the reference price. The value in the P: field depends on the benchmark value of the T: field if the model has been calibrated, but subsequent changes in T: do not change the underlying technology.

(ii) *Tax rates are interpreted differently for inputs and outputs.* The tax rate on inputs is specified on a net basis, while the tax rate on outputs is specified on a gross basis. That

is, the user cost of an input with market price  $p$  subject to an ad-valorem tax at rate  $t$  is  $p(1+t)$ , while the user cost of an output subject to an ad-valorem tax at rate  $t$  is  $p(1-t)$ . (A tax increases the producer cost of inputs and decreases the producer value of outputs.)

MPSGE provides a limited number of economic components with which complex models can be constructed. There are some models which lie outside the MPSGE domain, but in many cases it is possible to recast the equilibrium structure in order to produce an MPSGE model which is logically equivalent to the original model - usually after having introduced some sort of artificial commodity or consumer. In the present model, the use of commodity PT to allocate government revenue between households provides a fairly typical example of how this can be done. In the process of making such a transformation, one often gains a meaningful economic insight.

### 3.5. THE SOLUTION LISTING

The detailed solution listing for model **HARBERGER** is shown in Figure 5. The standard GAMS report facilities display solution values. Central variables are always either fixed (upper = lower), or they are non-negative (lower bound = 0, upper bound = +INF). The **MARGINAL** column in the solution report returns the value of the associated slack variable. Complementarity implies that in equilibrium, either the level value of a variable will be positive or the marginal value will be positive, but not both.

The output file (not shown) also provides details on the computational process. For an explanation of these statistics, see Rutherford (1993).

### 3.6. COMPUTING COUNTER-FACTUAL SCENARIOS

Figure 6 presents the GAMS code for specification and solution of three counterfactual equilibria. In these experiments, the nonuniform system of capital taxes from the benchmark is replaced by three alternative uniform factor tax structures: a tax on labor, a tax on capital, and a tax on both labor and capital. In each case, the tax rate is chosen to replace the benchmark tax revenue at benchmark prices and demand (ignoring induced substitution effects). Following each solution, the equilibrium values for tax revenue, welfare (Hicksian equivalent variation), employment, prices and output are stored in parameter **REPORT**.

## 4. ALTERNATIVE MODELS

The “standard” MPSGE model is based on fixed endowments and tax rates, but many empirical models do not fit into this structure. For example, in the model **HARBERGER**, the level of each replacement tax was specified to be consistent with “equal yield”, but as a result of the endogenous response of prices and quantities, the resulting tax revenues differed significantly from the benchmark levels. For example, when the capital tax is replaced by a uniform labor tax at a rate which, in the absence of labor supply response, produces “equal yield”, we find that tax revenue in fact declines by 39%. In order to perform differential (equal yield) tax policy analysis, it is therefore necessary to accommodate the endogenous determination of tax rates as part of the equilibrium computation. This is one of many possible uses of “auxiliary variables” in MPSGE.

```

---- VAR AL

      LOWER      LEVEL      UPPER      MARGINAL
X      .          1.000      +INF      .
Y      .          1.000      +INF      .

---- VAR P

      LOWER      LEVEL      UPPER      MARGINAL
X      .          1.000      +INF      .
Y      .          1.000      +INF      .

---- VAR W

      LOWER      LEVEL      UPPER      MARGINAL
K      .          1.000      +INF      .
L      .          1.000      +INF      .

---- VAR PT

              LOWER      LEVEL      UPPER      MARGINAL
              .          1.000      +INF      .

---- VAR RA

      LOWER      LEVEL      UPPER      MARGINAL
OWNER      .          70.000      +INF      .
WORKER     .          120.000      +INF      .

---- VAR GOVT

              LOWER      LEVEL      UPPER      MARGINAL
              .          30.000      +INF      .

```

Figure 5a. GAMS Solution Listing for Model HARBERGER

#### 4.1. TAX ANALYSIS WITH EQUAL YIELD

Figure 7 presents the MPSGE model definition for test problem SHOVEN. This model is equivalent to the HARBERGER, apart from the addition of an auxiliary variable TAU.

Within MPSGE, auxiliary variables can either represent price-adjustment instruments (endogenous taxes) or they can represent a quantity-adjustment instruments (endowment rations). In model SHOVEN, TAU is used to proportionally scale factor taxes in order to achieve a target level of government revenue. The auxiliary variable first appears in the \$PROD:AL(S) block, following the declaration of a tax agent. There are two fields associated with an endogenous tax. The first field (N:) gives the name of the auxiliary variable which will scale the tax rate. The second field (M:) specifies the multiplier. If the M: field is omitted, the multiplier assumes a default value of unity. If the value in the M: field is zero, the tax does not apply.

---- VAR CD

	LOWER	LEVEL	UPPER	MARGINAL
X.OWNER	.	30.000	+INF	.
X.WORKER	.	50.000	+INF	.
Y.OWNER	.	40.000	+INF	.
Y.WORKER	.	30.000	+INF	.

---- VAR DF

	LOWER	LEVEL	UPPER	MARGINAL
K.OWNER	.	.	+INF	EPS
K.WORKER	.	.	+INF	EPS
L.OWNER	.	.	+INF	EPS
L.WORKER	.	40.000	+INF	.

---- VAR EMPLOY

	LOWER	LEVEL	UPPER	MARGINAL
X	.	50.000	+INF	.
Y	.	10.000	+INF	.

---- VAR WLF

	LOWER	LEVEL	UPPER	MARGINAL
OWNER	.	1.000	+INF	.
WORKER	.	1.000	+INF	.

```

**** REPORT SUMMARY :      0   NONOPT
                           0  INFEASIBLE
                           0  UNBOUNDED
                           0   ERRORS

```

Figure 5b. GAMS Solution Listing for Model HARBERGER (cont.)

The auxiliary variable **TAU** also appears at the bottom of the file where it labels an associated equation. An auxiliary variable may or may not appear in its associated constraint. The constraint associated with **TAU** is based on a price index defined by **THETA(G)**. The equilibrium level of **TAU** is selected to provide level of tax revenue such that the value of transfers to households is held constant. (Endowments of the commodity **PT** are fixed, so when the value of **PT** is fixed, then so too are the value of transfers from **GOVT** to each of the households.)

**SHOVEN** illustrates how an auxiliary variable can be interpreted as a tax instrument. In the MPSGE syntax, auxiliary variables may also be employed to endogenously determine commodity endowments. There is no restrictions on how a particular auxiliary variable is to

```

* -----
* SECTION (iv) COUNTER-FACTUAL SPECIFICATION AND SOLUTION:

SET SC COUNTERFACTUAL SCENARIOS TO BE COMPUTED /
    UNIF_L  UNIFORM TAX ON LABOR,
    UNIF_K  UNIFORM TAX ON CAPITAL,
    UNIF_VA UNIFORM VALUE-ADDED TAX/

PARAMETER      TAXRATE(F,S,SC) COUNTERFACTUAL TAX RATES,
                REPORT(*,*,*,SC) SOLUTION REPORT - % CHANGES,
                PINDEX          PRICE DEFLATOR;

* SPECIFY COUNTER-FACTUAL TAX RATES TO ACHIEVE CETERIS
* PARIBUS BALANCED BUDGET:

TAXRATE("L",S,"UNIF_L") = GREV / SUM(G, FD("L",G));
TAXRATE("K",S,"UNIF_K") = GREV / SUM(G, FD("K",G));
TAXRATE("L",S,"UNIF_VA") = GREV / SUM((F,G), FD(F,G));
TAXRATE("K",S,"UNIF_VA") = GREV / SUM((F,G), FD(F,G));

```

*Figure 6a.* Specification Counter-Factual Scenarios

be interpreted. A single variable could conceivably serve simultaneously as an endogenous tax as well as a endowment ratio, although this would be rather unusual.

An algebraic formulation of the Shoven model is provided in Appendix D for the interested reader.

#### 4.2. PUBLIC GOODS AND ENDOGENOUS TAXATION

Consider a final extension of the 2x2 model in which tax revenue funds a pure public good. Model SAMUELSON presented in Figure 8. This model illustrates one of several ways that public goods can be modelled in MPSGE. Here the level of public provision is determined by a Samuelson-condition equating the sum of individual marginal rates of substitution (marginal benefit) with the marginal rate of transformation (marginal cost). Unlike the equal yield formulation, the tax revenues collected by GOVT are not returned lump-sum but are instead used to finance provision of a pure public good. This representation of government has not been widely adopted in the CGE literature, perhaps because of the difficulties involved in specifying preferences for public goods.

The relevant characteristic of a pure public good entering final demand is that each consumer “owns” the same quantity. Agents’ attitudes toward public goods differ, and because there is no market, agents’ valuations of the public good will also differ. In an MPSGE model, the separate valuations are accommodated through the introduction of “personalized” markets for public good - one market for each consumer. In the model, consumer expenditure encompasses both private and public “purchases”, and consumer income encompasses both private and public “endowments”. An individual is endowed

```

LOOP(SC,
*      INSTALL TAX RATES FOR THIS COUNTERFACTUAL:
      TF(F,S) = TAXRATE(F,S,SC);

$INCLUDE HARBERGER.GEN
SOLVE HARBERGER USING MCP;

*      -----
*      SECTION (v) REPORT WRITING:
*
*      REPORT SOME RESULTS:

      PINDEX = SUM(G, P.L(G) * THETA(G));

      REPORT("REVENUE","_",SC) = 100 * (PT.L/PINDEX - 1);
      REPORT("TAXRATE","_",SC) =
          100 * SMAX((F,S), TAXRATE(F,S,SC));
      REPORT("WELFARE",H,SC) = 100 * (WLF.L(H) - 1);
      REPORT("EMPLOY",S,SC) = 100 * (EMPLOY.L(S)/FD("L",S) - 1);
      REPORT("PRICE",G,SC) = 100 * (P.L(G)/PINDEX - 1);
      REPORT("PRICE",F,SC) = 100 * (W.L(F)/PINDEX - 1);
      REPORT("OUTPUT",S,SC) = 100 * (AL.L(S) - 1);
);
DISPLAY REPORT;

```

*Figure 6b.* Solution and Reporting

with a quantity of her own version of the public good determined by the level of public expenditures. An increase in taxes, to the extent that it increases tax revenue, will increase the level of public provision.

In this model, the structure of relative factor taxes is exogenous but the aggregate level of taxes is not. Tax rates are scaled up or down so that the sum of individual valuations of the public good (the marginal benefit) equals the cost of supply of the public good (the direct marginal cost).

Consider features of model **SAMUELSON** which do not appear in **SHOVEN**:

(i) There are new commodities **PG** and **VG(H)**. The first of these represents the direct marginal cost of public output from sector **GP**, a Leontief technology which converts private goods inputs into the public good. For the **SAMUELSON** structure, all government revenues apply to purchases of the public good (observe that the only good demanded by consumer **GOVT** is **PG**). The prices **VG(H)** represent the individual consumer valuations of the public good. Commodity **VG(H)** appears only in the endowments and demands of consumer **RA(H)**. The endowment record for **VG(H)** includes a quantity **V(H)** which is the benchmark valuation of the public good by agent **H**.

```

$ONTEXT

$MODEL: SHOVEN

$SECTORS:
    AL(S)

$COMMODITIES:
    P(G)  W(F)  PT

$CONSUMERS:
    RA(H)  GOVT

$AUXILIARY:
    TAU

$REPORT:
    V:CD(G,H)      D:P(G)      DEMAND:RA(H)
    V:DF(F,H)      D:W(F)      DEMAND:RA(H)
    V:EMPLOY(S)    I:W("L")    PROD:AL(S)
    V:WLF(H)       W:RA(H)

$PROD:AL(S)  s:0  va:ELAS(S)
    O:P(G)      Q:A(G,S)
    I:P(G)      Q:B(G,S)
    I:W(F)      Q:FD(F,S)  P:PF(F,S)
+   A:GOVT  N:TAU$TF(F,S)  M:TF(F,S)$TF(F,S)  va:

$DEMAND:RA(H)  s:1  gds:ESUB(H)
    D:P(G)      Q:C(G,H)  gds:
    D:W(F)      Q:D(F,H)
    E:W(F)      Q:E(F,H)
    E:PT        Q:TRN(H)

$DEMAND:GOVT
    D:PT        Q:GREV

$CONSTRAINT:TAU
    PT =E= SUM(G, THETA(G) * P(G));

$OFFTEXT

```

Figure 7. Differential Tax Policy Analysis

(ii) There are two auxiliary variables. **TAU** has the same interpretation as in the **SHOVEN**, determining the aggregate tax level. Auxiliary variable **LGP** is a rationing instrument representing an index of the “level of public goods provision”, scaled to unity in the benchmark. Consumer **RA(H)** thus is endowed with a quantity of **VG(H)** given by  $V(H) * LGP$ .

(iii) The constraint for **TAU** in **SAMUELSON** differs from the **TAU** constraint in **SHOVEN**. Here the constraint represents the Samuelson condition, equating the marginal cost ( $PG * GREV$ ) and the sum of individuals’ marginal benefit ( $\sum(H, VG(H) * V(H))$ ). The constraint for **LGP** simply assigns **LGP** equal to the sector **GP** activity level. (The **LGP** variable and constraint are only needed because the **R:** field only accepts auxiliary variables.)

An algebraic formulation of the Samuelson model is provided in Appendix D for the interested reader.

#### 4.3. COMPARING MODEL RESULTS

Although the foregoing discussion has focused on the nuances of MPSGE model syntax, but there are many interesting economic questions which can be addressed using even small-scale models such as the ones described here. Consider the output listing from parameter **REPORT** is displayed in Figure 9. It is perhaps surprising to note that none of the uniform tax structures represents a Pareto-superior choice compared to the benchmark tax structure. Furthermore, from the standpoint of aggregate welfare (“**WELFARE.TOTAL**” = income-weighted sum of individual **EV**’s), only the uniform capital tax represents an improvement.

### 5. SUMMARY

This paper has provided an introduction to a new GAMS subsystem for applied general modeling. This extension of GAMS accomodates a tabular representation of highly non-linear cost and expenditure functions through which model specification is concise and transparent. The paper has presented three small examples which illustrate the programming environment and its application to traditional economic issues in public finance for which applied general equilibrium analysis is a standard tool.

### References

- Ballard and Fullerton. Distortionary taxes and the provision of public goods. *Journal of Economic Perspectives*, 6(3), 1992.
- Tony Brooke, David Kendrick, and Alex Meeraus. *GAMS: A User’s Guide*. The Scientific Press, Redwood City, California, 1988.
- Thomas F. Rutherford. Miles: A mixed inequality and nonlinear equation solver. <http://www.gams.com/docs/pdf/miles.PDF>, 1993.
- Thomas F. Rutherford. *Applied General Equilibrium Modeling*. PhD thesis, Stanford University, 1987.
- J. Shoven and J. Whalley. Applied general equilibrium models of taxation and international trade: Introduction and survey. *Journal of Economic Literature*, 22:1007–1051, 1984.
- G. Thompson and S. Thore. *Computational Economics*. Scientific Press, Redwood City, California, 1992.

```

$ONTEXT

$MODEL: SAMUELSON

$SECTORS:
    AL(S)  GP

$COMMODITIES:
    P(G)  W(F)  PG  VG(H)

$CONSUMERS:
    RA(H)  GOVT

$AUXILIARY:
    TAU  LGP

$REPORT:
    V:CD(G,H)      D:P(G)      DEMAND:RA(H)
    V:DF(F,H)      D:W(F)      DEMAND:RA(H)
    V:EMPLOY(S)    I:W("L")    PROD:AL(S)
    V:WLF(H)       W:RA(H)

$PROD:AL(S)  s:0  va:ELAS(S)
    O:P(G)      Q:A(G,S)
    I:P(G)      Q:B(G,S)
    I:W(F)      Q:FD(F,S)  P:PF(F,S)
+   A:GOVT  N:TAU$TF(F,S)  M:TF(F,S)$TF(F,S)  va:

$PROD:GP  s:0
    O:PG      Q:GREV
    I:P(G)    Q:GD(G)

$DEMAND:RA(H)  s:1  gds:ESUB(H)
    D:P(G)      Q:C(G,H)  gds:
    D:W(F)      Q:D(F,H)
    D:VG(H)     Q:V(H)
    E:VG(H)     Q:V(H)  R:LGP
    E:W(F)     Q:E(F,H)

$DEMAND:GOVT
    D:PG      Q:GREV

$CONSTRAINT:TAU
    GREV * PG =E= SUM(H, V(H) * VG(H));

$CONSTRAINT:LGP
    LGP =E= GP;

$OFFTEXT

```

Figure 8. Endogenous Determination of Tax Revenue

## INDEX 1 = HARBERGER

	UNIF_K	UNIF_L	UNIF_VA
REVENUE. _	3.9	-38.9	-0.8
TAXRATE. _	50.0	50.0	25.0
WELFARE. OWNER	1.9	42.4	18.5
WELFARE. WORKER	-0.1	-26.8	-10.9
WELFARE. TOTAL	0.6	-1.3	-3.48143E-2
EMPLOY .X	-5.3	-6.9	-8.4
EMPLOY .Y	20.5	34.4	22.1
PRICE .X	-10.4	-11.2	-10.3
PRICE .Y	11.8	12.8	11.8
PRICE .K	3.9	59.5	24.5
PRICE .L	-4.7	-38.9	-23.5
OUTPUT .X	3.6	-1.0	0.4
OUTPUT .Y	-3.7	2.0	-2.0

## INDEX 1 = SHOVEN

	UNIF_K	UNIF_L	UNIF_VA
TAXRATE. _	47.1	134.2	25.3
WELFARE. OWNER	3.3	40.2	18.3
WELFARE. WORKER	-1.0	-29.2	-10.8
WELFARE. TOTAL	0.6	-3.6	-3.51710E-2
EMPLOY .X	-5.0	-19.7	-8.5
EMPLOY .Y	21.5	12.1	21.9
PRICE .X	-10.4	-9.0	-10.3
PRICE .Y	11.9	10.2	11.8
PRICE .K	6.2	49.8	24.2
PRICE .L	-5.0	-56.5	-23.6
OUTPUT .X	3.6	-7.9	0.3
OUTPUT .Y	-3.4	-2.0	-2.1

## INDEX 1 = SAMUELSON

	UNIF_K	UNIF_L	UNIF_VA
REVENUE . _	-1.4	-14.5	-6.7
TAXRATE . _	45.7	88.8	22.8
WELFARE . OWNER	4.7	43.9	21.1
WELFARE . WORKER	-2.0	-31.4	-12.9
WELFARE . TOTAL	0.5	-3.7	-0.4
EMPLOY .X	-4.9	-7.5	-5.9
EMPLOY .Y	24.5	37.5	29.7
PRICE .X	-10.7	-11.3	-10.9
PRICE .Y	12.2	13.0	12.5
PRICE .K	7.8	60.3	29.0
PRICE .L	-6.0	-51.8	-24.5
OUTPUT .X	3.0	-2.2	0.9
OUTPUT .Y	-2.3	3.3	-2.58148E-2
PROVISION. _	-0.8	-13.9	-6.1

Figure 9. Numerical Results from Alternative Models

## 6. Selected Publications

### 6.1. THE COMPLEMENTARITY FORMAT FOR ECONOMIC EQUILIBRIUM

Lars Mathiesen. Computational experience in solving equilibrium models by a sequence of linear complementarity problems. *Operations Research*, 33(6):1225–1250, 1985.

Lars Mathiesen. Computational of economic equilibria by a sequence of linear complementarity problems. *Mathematical Programming Study*, 23, 1985. North Holland.

Lars Mathiesen. An algorithm based on a sequence of linear complementarity problems applied to a walrasian equilibrium model: An example. *Mathematical Programming*, 37:1–18, 1987.

### 6.2. APPLICATIONS IN ENVIRONMENTAL ECONOMICS

Stefan Felder and Thomas Rutherford. Unilateral co2 reductions and carbon leakage: the effect of international trade in oil and basic materials. *Journal of Environmental Economics and Management*, 25:162–176, 1993.

Carlo Perroni and Thomas F. Rutherford. International trade in carbon emission arights and basic materials: general equilibrium calculations for 2020. *Scandinavian Journal of Economics*, 95(3):257–278, 1993.

Carlo Perroni and Randy M. Wigle. International trade and environmental quality: How important are the linkages? *Canadian Journal of Economics*, August 1994.

Deborah Vaughn Nestor and Carl Pasurka. A CGE model of pollution abatement processes for assessing the economic effects of environmental policy. *Economic Modelling*, 12(1):53–59, 1995.

Deborah Vaughn Nestor and Carl Pasurka. Alternative specifications for environmental control costs in a general equilibrium framework. *Economics Letters*, 48(3):273–280, June 1995.

Carlo Perroni and Randy M. Wigle. Environmental policy modeling. In Thomas Hertel, editor, *Global Trade Analysis: Modeling and Applications*. Cambridge University Press, 1996.

Christoph Böhringer and Thomas F. Rutherford. Carbon taxes with exemptions in an open economy: a general equilibrium analysis of the german tax initiative. *Journal of Environmental Economics and Management*, pages 189–203, 1996.

Glenn W. Harrison and Bengt Kriström. Carbon emissions and the economic costs of transport policy in sweden. In R. Roson and K.A. Small, editors, *Environment and Transport in Economic Modelling*. Kluwer Academic Press, Amsterdam, 1998. forthcoming.

### 6.3. APPLICATIONS IN ECONOMIC GROWTH

Alan Manne and Thomas Rutherford. A long-term model of oil markets, economic growth and balance of payments constraints. *Empirical Economics*, 16:51–69, 1991.

James R. Markusen and Robert Feenstra. Accounting for growth with new inputs. *Journal of International Economics*, 1993.

George R. Boyer, Timothy J. Hatton, and Kevin O'Rourke. Emigration and economic growth in Ireland, 1850–1914. *International Migration and World Development*, pages 221–239, 1994.

Kevin O'Rourke and Jeffrey G. Williamson. Around the european periphery 1870-1913: globalization, schooling and growth. *European Review of Economic History*, 1(2):153–190, 1995.

David Tarr and Thomas F. Rutherford. Blueprints, spillovers and the dynamic gains from trade liberalization in a small open economy. In Richard Baldwin, editor, *Dynamic Analysis of Commercial Policy*. forthcoming, 1998.

#### 6.4. APPLICATIONS IN ECONOMIC HISTORY

Kevin O'Rourke. Did the great irish famine matter? *Journal of Economic History*, 51:1–22, 1991.

Kevin O'Rourke. Burn everything British but their coal: the Anglo-Irish economic war of the 1930's. *Journal of Economic History*, 51:357–366, 1991.

Kevin O'Rourke. Rural depopulation in a small open economy: Ireland 1856–1876. *Explorations in Economic History*, 28:409–432, 1991.

Kevin O'Rourke and Jeffrey G. Williamson. Late 19th century Anglo-American factor price convergence: Were heckscher and ohlin right? *Journal of Economic History*, 54:892–916, 1994.

Timothy J. Hatton, Kevin O'Rourke, and Jeffrey G. Williamson. Mass migration, commodity market integration, and real wage convergence: the late nineteenth century atlantic economy. In Timothy J. Hatton and Jeffrey G. Williamson, editors, *International Migration and World Development*, pages 203–220. Routledge, 1994.

Kevin O'Rourke. The economic impact of the famine in the short and long run. *American Economic Review (Papers and Proceedings)*, 84:309–313, 1994.

Kevin O'Rourke. The repeal of the corn laws and irish emigration. *Explorations in Economic History*, 31:120–138, 1994.

Kevin O'Rourke and Jeffrey G. Williamson. Open economy forces and late 19th century swedish catch-up: a quantitative accounting. *Scandinavian Economic and Social History*, XLIII:171–203, 1995.

Kevin O'Rourke. The costs of international economic disintegration: Ireland in the 1930's. *Research in Economic History*, 15:215–259, 1995.

Kevin O'Rourke, Alan M. Taylor, and Jeffrey G. Williamson. Factor price convergence in the late nineteenth century. *International Economic Review*, 37(3):499–530, 1996.

Kevin O'Rourke. The european grain invasion, 1870-1913. *Journal of Economic History*, forthcoming, 1998.

#### 6.5. APPLICATIONS IN INTERNATIONAL TRADE

Jan Haaland, Victor Norman, Thomas F. Rutherford, and Tor Wergeland. Vemod: A model of world trade. *Scandinavian Journal of Economics*, 89(3):705–721, 1987.

Glenn W. Harrison, Thomas F. Rutherford, and Ian Wooton. The economic impact of the european community. *American Economic Review (Papers and Proceedings)*, 79:288–294, 1989.

Trien Nguyen, Carlo Perroni, and Randy M. Wigle. An evaluation of the draft final act of the uruguay round. *Economic Journal*, November 1993.

Glenn W. Harrison, Thomas F. Rutherford, and David Tarr. Trade reform in the partially liberalized economy of Turkey. *The World Bank Economic Review*, 7(2):191–217, 1993.

Florencio Lopez de Silanes, James R. Markusen, and Thomas F. Rutherford. Complementarity and increasing returns in intermediate inputs. *Journal of Development Economics*, 45:133–151, 1994.

Linda Hunter, James R. Markusen, and Thomas F. Rutherford. North American free trade and the production of finished automobiles. In Timothy Kehoe and Patrick Kehoe, editors, *Modeling North American Economic Integration*, pages 117–130. Kluwer, 1995.

Joseph F. Francois, B. McDonald, and H. Nordstrom. The uruguay round: A numerically based qualitative assessment. In W. Martin and L.A. Winters, editors, *The Uruguay Round and the Developing Countries*. Cambridge University Press, New York, 1996.

Florencio Lopez de Silanes, James R. Markusen, and Thomas F. Rutherford. Trade policy subtleties with multinational firms. *European Economic Review*, pages 1605–1627, 1996.

James R. Markusen. The boundaries of multinational enterprises and the theory of international trade. *Journal of Economic Perspectives*, 9:169–189, 1995.

Glenn W. Harrison, Thomas F. Rutherford, and David Tarr. Quantifying the Uruguay round. *Economic Journal*, 107:1405–1430, 1997.

Kevin O'Rourke. Measuring protection: a cautionary tale. *Journal of Development Economics*, 53(1):169–183, 1997.

Thomas F. Rutherford, David Tarr, and Elisabet Rutstrm. Morocco's free trade agreement with the EU: A quantitative assessment. *Economic Modelling*, 14(2):237–269, April 1997.

Glenn W. Harrison, Thomas F. Rutherford, and David Tarr. Opciones de politica comercial para Chile: una evaluacin cuantitativa. *Cuadernos de Economa* 34:101–137, 1997.

James R. Markusen and Anthony J. Venables. The role of multinational firms in the wage gap debate. *Review of International Economics*, 5:435–451, 1997.

James R. Markusen and Anthony J. Venables. Multinational firms and the new trade theory. *Journal of International Economics*, forthcoming, 1998. NBER working paper 5036.

#### 6.6. OTHER APPLICATIONS

Thomas F. Rutherford, John Whalley, and Randy Wigle. Capitalization, conditionality and dilution: Land prices and the U.S. wheat program. *Journal of Policy Modelling*, 12(3):605–622, 1990.

Kevin O'Rourke. Industrial policy, employment policy and the non-traded sector. *Journal of the Statistical and Social Inquiry Society of Ireland*, XXVII, Part II:61–80, 1994/1995.

Keith Maskus, Thomas F. Rutherford, and Steven Selby. Economic implications of changes in labor standards: A computational analysis for Mexico. *North American Journal of Economics and Finance*, 6(1):171–188, 1996.

## 7. Language Syntax

### 7.1. GENERAL SYNTAX RULES

- All input is free format (spaces and tabs are ignored) except keywords for which "\$" must appear in column 1.
- End-of-line is significant. Continuation lines are indicated by a "+" in column 1.
- In general, input is not case sensitive, except in the specification of sub-nests for production and demand functions.
- Numeric expression involving GAMS parameters or constants must be enclosed in parentheses.

### 7.2. KEYWORDS

Keywords typically appear in the following order:

**\$ONTEXT** Indicate the beginning of a GAMS comment block containing an MPSGE model.

**\$MODEL:** *model-name* *model-name* must be a legitimate file name. This name is subsequently used to form *model-name.GEN* (this file name *must* be upper case when running under UNIX).

**\$SECTORS:**, **\$COMMODITIES:**, **\$AUXILIARY:**, **\$CONSUMERS:** Four keywords define variables which are used in the model. Entries in these blocks share the same syntax. The **\$AUXILIARY** block is only used in models with side constraints and endogenous taxes or rationed endowments.

**\$PROD:sector** Production functions must be specified for each production sector in the model.

**\$DEMAND:consumer** Demand functions must be specified for every consumer in the model. General structure is the same as for production functions above.

**\$CONSTRAINT:auxiliary** Specifies a side constraint to be associated with a specified auxiliary variable.

**\$REPORT:** Identifies the set of additional variables to be calculated. These include outputs and inputs by sector and demands by individual consumers.

**\$OFFTEXT** Indicates the end of model specification.

### 7.3. VARIABLE DECLARATIONS

There are four classes of variables within an MPSGE model: activity levels for production sectors, prices for commodities, income levels for consumers and level values for auxiliary variables. These classes of variables are distinguished in order to permit additional semantic checking by the MPSGE preprocessor. For example, if P has been declared as a

price (within the \$COMMODITIES: block), then the preprocessor would report an error if it encountered "\$PROD:P".

The \$SECTORS:, \$COMMODITIES:, \$CONSUMERS: and \$AUXILIARY: blocks contain implicit GAMS variable declarations in which the index sets must be specified in the GAMS program above and the variable names must be distinct from all other symbols in the GAMS program. One or more variables may be declared per line separated by one or more spaces.

```
$SECTORS:
      Y(R,T)      ! Output in region R in period T
      K(T)        ! "Aggregate capital stock, period T"
```

In these declarations, the trailing comments (signified by "!") are interpreted as variable name descriptors which subsequently appear in the solution listing.

The equivalent GAMS declaration for these variables would be:

```
VARIABLES  Y(R,T)  Output in region R in period T
            K(T)    "Aggregate capital stock, period T";
```

As with the usual GAMS syntax, when a variable descriptor contains a punctuation symbol such as ",", it is required to be enclosed in quotes.

```
$SECTORS:
      X(R,T)$XO(R) GT 0)
```

Here, the GAMS conditional operator "\$" is used to restrict the domain of the variable X. The expression following the dollar sign is passed through to the GAMS compiler and must conform to GAMS syntax rules.

```
$SECTORS:
      X Y(R)$YO(R) Z      ! Descriptor for Z
      W(G,R,T)           ! Descriptor for W
```

More than one symbol may appear on a single line. The descriptor only applies to the last one.

All MPSGE variables must be declared. When multidimensional variables are specified, they must be declared explicitly - declarations like X(\*) are not permitted. Two further restrictions are that the sets used in the declaration must be static rather than dynamic, and any variable which is declared must be used in the model. There is a simple way to work around these restrictions. Let me illustrate with an example. Suppose that in a model the set of production sectors AL is employed for all elements of a static set S which satisfy a particular condition, for example BMX(S) not equal to 0. This would require that AL be declared as follows:

```
$SECTORS:
    AL(S)$BMX(S)
```

In this context, the symbol “\$” is used as an “exception operator” which should be read as “such that”. In this case, we have generated one AL sector for each element of the set  $S$  for which  $BMX(S)$  is nonzero.

#### 7.4. FUNCTION DECLARATIONS

Functional declarations characterize nested CES functions which characterize both preferences and technology. The former are written within a \$DEMAND: block and the latter within a \$PROD: block. Tax entries may appear within a \$PROD: block but not within a \$DEMAND: block, otherwise the syntax is nearly identical. The syntax for these blocks will be described through a sequence of examples:

```
$PROD:Y(R)  s:1
O:P(R)      Q:YO(R)
I:W(F,R)    Q:FDO(F,R)
```

This block characterizes a Cobb-Douglas production function in which the elasticity of substitution between inputs is one - "s : 1" in the first line which sets a top level substitution elasticity equal to unity. Variable  $Y(R)$  is an activity level declared within the \$SECTORS: block. Variables  $P(R)$  and  $W(F,R)$  are prices declared within the \$COMMODITIES: block. The O: label indicates an output, and the I: prefix indicates an input. The Q: fields in both records represent “reference quantities”.  $YO(R)$  and  $FDO(F,R)$  must be GAMS parameters defined previously in the program.

```
$PROD:X(R)  s:ESUB(R)  a:0  va:(ESUB(R)*0.2)
O:PX        Q:XO(R)
I:PY(G)     Q:YXO(G,R)  a:
I:PL        Q:LXO(R)   va:
I:PK        Q:KXO(R)   va:
```

The keyword line specifies three separate elasticities related to this function.  $ESUB(R)$  is the top level elasticity of substitution. There are two sub-nests in the function. Nest  $a:$  is a Leontief nest (in which the compensated elasticity is zero). The elasticity of substitution in nest  $va:$  is one-fifth of the top level elasticity.

In the function specification, commodities  $PY(G)$  (one input for each element of set  $G$ ) enter in fixed proportions. Commodities  $PL$  and  $PK$  enter in nest  $b$ .

If this function has been specified using a balance benchmark dataset with reference prices equal to unity, then the following identity should be satisfied:

$$XO(R) = \text{SUM}(G, YXO(G,R)) + LXO(R) + KXO(R)$$

```

$PROD:AL(S)  s:0  a:ELAS(S)
  O:P(G)      Q:A(G,S)
  I:P(G)      Q:B(G,S)
  I:W(F)      Q:FD(F,S)  P:PF(F,S)  A:GOVT  T:TF(F,S)  a:

```

In this function, we have two new ideas. The first is the use of a reference price denoted by "P:". This entry indicates that the function should be calibrated to a reference point where individual input prices (gross of tax) equal PF(F,S). If P: does not appear, prices of one are assumed.

The second new idea here is that taxes may be levied on production inputs. The A: label identifies the name of the tax agent (a \$CONSUMER:). The T: label identifies the ad-valorem tax rate.

```

$DEMAND:RA(R)$RAO(R)  s:1
  E:PL                Q:LE(R)
  D:P(G)$DG(G)       Q:DO(G,R)$DD(G,R)  P:PO(G,R)

```

This function specification demonstrates the use of conditionals. This function is only generated when RAO(R) is nonzero. The demands D: for a particular element of set G are suppressed entirely when DG(G) equals 0. The Q: field also has an exception operator, so that the default value for Q: (unity) is applied when DD(G,R) equals zero.

This example is somewhat artificial, but it illustrates the distinction between how exception operators affect lead entries (I:, O:, D: and E:) and subsequent entries. When an exception is encountered on the lead entry, the entire record may be suppressed. Exceptions on subsequent entries only applied to a single field.

The valid labels in a function declaration (\$PROD: or \$DEMAND:) line include:

- s: Top level elasticity of substitution between inputs or demands.
- t: Elasticity of transformation between outputs in production. (valid only in \$PROD blocks)
- a:,va:,etc. Elasticities of substitution in individual input nests.

The recognized labels in an I: or O: line include:

- Q: Reference quantity. Default value is 1. When specified, it must be the second entry.
- P: Reference price. Default value is 1.
- A: Tax revenue agent. Must be followed by a consumer name.
- T: Tax rate field identifier. (More than one tax may apply to a single entry.)

- N:** Endogenous tax. This label must be followed by the name of an auxiliary variable.
- M:** Endogenous tax multiplier. The advalorem tax rate is the product of the value of the endogenous tax and this multiplier.
- a:, va:, etc.** Nesting assignments. Only one such label may appear per line, maximum four characters.

The valid labels in an **E:** line include:

- Q:** Reference quantity.
- R:** Rationing instrument indicating an auxiliary variable.

The valid labels in a **D:** line include:

- Q:** Referene quantity.
- P:** Reference price.
- a:, cl:, etc.** Nesting assignment Maximum four characters.

## 7.5. CONSTRAINTS

Auxiliary constraints in MPSGE models conform to standard GAMS equation syntax. They may refer to any of the four classes of variables, **\$SECTORS**, **\$COMMODITIES**, **\$CONSUMERS** and **\$AUXILIARY**, but they may not reference variables names declared within a **\$REPORT** block.

Complementarity conditions apply to upper and lower bounds on auxiliary variables and the associated constraints. For this reason, the orientation of the equation is important. When an auxiliary variable is designated **POSITIVE** (the default), the auxiliary constraint should be expressed as a “greater or equal” inequality (**=G=**). If an auxiliary variable is designated **FREE**, the associated constraint must be expressed as an equality (**=E=**).

```
$CONSTRAINT:TAU
```

```
    G =G= X * Y;
```

```
$CONSTRAINT:MU(I)$MUO(I)
```

```
    MU(I) * P(I) * Q(I) =G= SUM(J, THETA(I,J) * PX(J));
```

The exception applied in this example restricts the equation only to those elements of set **I** for which **MUO(I)** is not zero.

## 7.6. REPORT DECLARATION

The GAMS interface to MPSGE normally returns level values only for the central variables - those declared within \$SECTORS:, \$COMMODITIES:, \$CONSUMERS: and \$AUXILIARY: sections. An equilibrium determines not only these values, but also levels of demand and supply by individual sectors and consumers. Given benchmark information, elasticities and the equilibrium values, all such demands can be computed, but this can be tedious to do by hand. In order to have these values returned by MPSGE, it is necessary to indicate the name of the variable into which the value is to be returned.

The general form is as follows:

```
$REPORT:
  V:variable name  I:commodity      PROD:sector
  V:variable name  O:commodity      PROD:sector
  V:variable name  D:commodity      DEMAND:consumer
  V:variable name  W:consumer
```

The first row returns an input quantity, the second row returns an output quantity, the third returns a demand quantity, and the fourth row returns a consumer welfare index. (Note: the level value returned for a “consumer variable” is an income level, not a welfare index.)

```
$REPORT:
  V:DL(S)          I:PF("L")          PROD:Y(S)
  V:DK(S)          I:PF("K")          PROD:Y(S)
  V: SX(G,S)$SXO(G,S)  O:PX(G)          PROD:X(S)
  V:D(G,H)         D:P(G)           DEMAND:RA(H)
  V:W(H)           W:RA(H)
```

Note that the “\$” exception is only meaningful on the first entry. Also notice that the domain of the report variable must conform to the domain of the subsequent two entries.

## 7.7. EXCEPTION HANDLING

The GAMS exception operator can be used on virtually any entry in the MPSGE input file. For example, if you want to have sector X(S) have one production structure for elements S in a subset T(S), you can provide separate production function declarations as follows:

```
$PROD:X(S)$T(S)
...           ! sector X described for S in T

$PROD:X(S)$ (NOT T(S))
...           ! sector X described for S not in T.
```

The preprocessor does not require one and exactly one declaration for each sector. If multiple declarations appear, the later set of coefficients overwrites the earlier set.

## 7.8. SWITCHES AND DEBUG OUTPUT

Run-time tolerances and output switches may be specified within the vector-syntax model specification or using the PUT facility, they can be written directly to the MPS input file. Output switches control the level of debug output written by the MPSGE subsystem to the solver status file. Reports provided by \$ECHOP, \$FUNLOG and \$DATECH can be returned to the listing file by specifying "OPTION SYSOUT=ON;" within the GAMS program prior to the SOLVE statement. The recognized MPSGE parameters are:

`$ECHOP: logical           Default=.FALSE.`

is a switch for returning all or part of the scalar MPSGE file to the solver status file. In order to have this output printed in the listing file, enter the GAMS statement "OPTION SYSOUT=ON;" prior to solving the model.

`$PEPS: real               Default=1.0E-6`

is the smallest price for which price-responsive demand and supply functions are evaluated. If a price is below PEPS, it is perturbed (set equal to PEPS) prior to the evaluation.

`$EULCHK: logical        Default=.TRUE.`

is a switch for evaluating Euler's identity for homogeneous equations. The output is useful for monitoring the numerical precision of a Jacobian evaluation. When a price or income level is perturbed in a function, the Euler check may fail.

`$WALCHK: logical        Default=.TRUE.`

is a switch for checking Walras's law. Like EULCHK, this switch is provided primarily to monitor numerical precision. When an income level is perturbed, the Walras check may fail.

`$FUNLOG: logical        Default=.FALSE.`

is a switch to generate a detailed listing of function evaluations for all production sectors and consumers.

FUNLOG triggers a function evaluation report which provides detailed output describing the evaluation of supply and demand coefficients. The information provide is sufficient that an industrious graduate student should be able to reproduce the results (given a pencil, paper and slide rule).

The evaluation report has the following headings:

T Coefficient "type" with the following interpretation:

IA Input aggregate

OA Output aggregate

I Input

O Output

D Demand

E Endowment

N Name (either nest identifier or commodity name)

PBAR Benchmark price (the P: field value)

P Current price (gross of tax)

QBAR Benchmark quantity (the Q: field value)

Q Current quantity

KP Identifier for parent entry in nesting structure.

ELAS Associated elasticity (input or output aggregates only)

When \$FUNLOG: .TRUE is specified, a complete report of demand and supply coefficients for every production and demand function in every iteration. Be warned that with large models this can produce an enrmous amount of output!

The following two function evaluation reports are generated in the first iteration in solving case "L" for model HARBERGER:

```
$DATECH:    logical    Default=.FALSE.
```

is a switch to generate a annotated listing of the function and Jacobian evaluation including a complete listing of all the nonzero coefficients.

MPSGE generates an analytic full first-order Taylor series expansion of the nonlinear equations in every iteration. Nonzero elements of the Jacobian matrix are passed to the system solver (MILES or PATH) which uses this information in the direction-finding step of the Newton algorithm. Coefficients are produced with codes which help interpret where they came from. The following codes are used:

W0 indicates an element from the orthogonal part of F().

T	N	PBAR	P	QBAR	Q	KP	ELAS
IA	s	1.0000E+00	8.9198E-01	1.0000E+02	1.0000E+02		0.00
OA	t	1.0000E+00	1.0000E+00	1.0000E+02	1.0000E+02		0.00
IA	a	1.0000E+00	8.7998E-01	9.0000E+01	9.0000E+01	s	1.00
O	P.X	1.0000E+00	1.0000E+00	1.0000E+02	1.0000E+02	t	
I	P.Y	1.0000E+00	1.0000E+00	1.0000E+01	1.0000E+01	s	
I	W.K	2.0000E+00	1.5000E+00	2.0000E+01	2.3466E+01	a	
I	W.L	1.0000E+00	1.0000E+00	5.0000E+01	4.3999E+01	a	

Function evaluation for: RA.OWNER

T	N	PBAR	P	QBAR	Q	KP	ELAS
IA	s	1.0000E+00	1.0000E+00	7.0000E+01	7.0000E+01		1.00
OA	t	1.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00		0.00
IA	a	1.0000E+00	1.0000E+00	7.0000E+01	7.0000E+01	s	0.50
D	P.X	1.0000E+00	1.0000E+00	3.0000E+01	3.0000E+01	a	
D	P.Y	1.0000E+00	1.0000E+00	4.0000E+01	4.0000E+01	a	
E	W.K	1.0000E+00	1.0000E+00	6.0000E+01	0.0000E+00		
E	PT	1.0000E+00	1.0000E+00	1.0000E+01	0.0000E+00		

Figure 10. Function Evaluation for: AL.X

W1 indicates an element from the non-orthogonal part of  $F()$ .

B indicates a linear term from  $F$ .

E0 indicates a homogeneous Jacobian entry.

E1 indicates a non-homogeneous Jacobian entry.

The Euler checksum examines elements from the linearization which are type "E0". The Walras check sum examines elements from the function evaluation which are type "W0".

Needless to say, the \$DATECH:.TRUE. switch produces a very big status file for large models. It is not something which is very useful for the casual user.

Here is a partial listing of nonzeros generated during the first linearization for scenario "L" in model HARBERGER:

```

----- Coefficients for sector:AL.X
P.X          AL.X          1.0000E+02    B
AL.X        P.X          -1.0000E+02    B
P.Y          AL.X          -1.0000E+01    B
AL.X        P.Y          1.0000E+01    B
W.K          AL.X          -2.3466E+01    B
AL.X        W.K          3.5199E+01    B
W.L          AL.X          -4.3999E+01    B
AL.X        W.L          4.3999E+01    B
W.K          W.K          1.3037E+01    E0    1.3037E+01
W.K          W.L          -1.3037E+01    E0    -1.3037E+01
W.L          W.K          -1.9555E+01    E0    -1.9555E+01
W.L          W.L          1.9555E+01    E0    1.9555E+01
GOVT        AL.X          -1.1733E+01    B
GOVT        W.K          -1.1733E+01    E1
GOVT        W.K          6.5184E+00    E0    6.5184E+00
GOVT        W.L          -6.5184E+00    E0    -6.5184E+00

----- Income for consumer:RA.OWNER
W.K          6.0000E+01    W0
RA.OWNER    W.K          -6.0000E+01    B
PT          1.0000E+01    W0
RA.OWNER    PT          -1.0000E+01    B
RA.OWNER    RA.OWNER    1.0000E+00    B

----- Demands for consumer:RA.OWNER
P.X          -3.0000E+01    W0
P.Y          -4.0000E+01    W0
P.X          P.X          8.5714E+00    E0    8.5714E+00
P.X          P.X          1.2857E+01    E0    1.2857E+01
P.X          P.Y          -8.5714E+00    E0    -8.5714E+00
P.X          P.Y          1.7143E+01    E0    1.7143E+01
P.Y          P.X          -8.5714E+00    E0    -8.5714E+00
P.Y          P.X          1.7143E+01    E0    1.7143E+01
P.Y          P.Y          8.5714E+00    E0    8.5714E+00
P.Y          P.Y          2.2857E+01    E0    2.2857E+01
P.X          RA.OWNER    -4.2857E-01    E0    -3.0000E+01
P.Y          RA.OWNER    -5.7143E-01    E0    -4.0000E+01

```

Figure 11. Partial Listing for HARBERGER

## 8. File Structure

This appendix provides an overview of the structure of GAMS input files which include MPSGE models. The text of the paper presents many of these ideas by way of example, but it may also be helpful for some users to have a “template” for constructing MPSGE models. The discussion in this section focuses on a “generic” input file, the schematic form of which is presented in Figure 10. This section first presents a “top down” view of program organization, and then it discusses aspects of the new syntax for model specification.

### 8.1. FLOW OF CONTROL

When a model is developed using GAMS as a front-end to MPSGE, the input file generally has five sections as identified in Figure 10. Section (i), the benchmarking section, contains standard GAMS statements. This includes GAMS **SET** declarations, input data (**SCALARS**, **PARAMETERS** and **TABLES**), and **PARAMETER** declarations for intermediate arrays used in benchmarking or model specification. In complex models, this section of the file will typically contain some algebraic derivations, the result of which is a calibrated benchmark equilibrium dataset.

Users who are unfamiliar with GAMS can consult the manual. Beginning GAMS programmers should remember that the MPSGE interface to GAMS is unlike other solution subsystems. “Level values” are passed between the GAMS program and MPSGE in the usual fashion, but MPSGE models do not require the explicit use of the **VARIABLE** or **EQUATION** statements.)

The second section of the file consists of a GAMS comment range, beginning with an **\$ONTEXT** record and ending with an **\$OFFTEXT** record, followed by an invocation of the preprocessor. The preprocessor writes operates on statements in the MPSGE model declaration which are “invisible” to the GAMS compiler. This program reads the MPSGE model statements and generates GAMS-readable code, including a `model-name.gen` file. Additional GAMS code produced by the preprocessor includes declarations for each of the central variables and report variables in the MPSGE model.

The third section of the generic input file performs a “benchmark replication” and may not be present in all applications. There are four statements required for benchmark validation. The first statement sets the iteration limit to be zero; the second statement causes the MPSGE model to be “generated”, and the third statement causes the MPSGE solver to read the model and return the pdeviations. In this call, the level values passed to the solver are unaltered because the iteration limit is zero. Market excess supplies and zero profit checks are returned in the “marginals” of the associated commodity prices and activity levels, respectively. The final statement in this section resets the iteration limit to 1000 (the default value) for subsequent counter-factual computations.

Section (iv) defines and then computes a counter-factual equilibrium. A counter-factual equilibrium is defined by parameter values such as tax rates or endowments which take on values different from those in the benchmark equilibrium. Within the GAMS interface to MPSGE, it is also possible to fix one or more central variables. When any variable is fixed, the associated equation is omitted from the equilibrium system during the solution process but the resulting imbalance is then reported in the solution returned through the marginal.

The final section of the file represents the GAMS algebra required for comparing counter-factual equilibria. It would be possible, for example, to construct welfare measures

or to report percentage changes in certain values. All of these calculations are quite easy because the equilibrium values are returned as level values in the associated variables.

For large models, the advantage of the vector format is that by using appropriately defined GAMS sets, the number of individual functions which need to be defined is reduced only to the number of “classes” of functions. This makes it possible to represent large dimensional models using only a few lines of code.

To summarize, here are the basic features of a program which uses GAMS as a front-end to MPSGE:

- (i) An MPSGE model is defined within a GAMS comment range followed by

```
$sysinclude mpsgeset model-name
```

- (ii) Every SOLVE statement for a particular model is preceded by \$INCLUDE MODEL.GEN. The GEN file is written by the preprocessor based on the model structure.

- (iii) Solution values for the central variables in the MPSGE model and any declared “report variables” are returned in GAMS variable level values. Level values for slacks are returned as “marginals” for the associated variables.

- (iv) The model description follows a format which is a direct extension of the scalar data format. Certain aspects of the new language, such as case folding, are incompatible with the original MPSGE syntax.

## 8.2. GAMS CODE GENERATED BY THE PREPROCESSOR: THE GEN FILE

Most novice users will find it easiest to treat the preprocessor output files as “black boxes”. These files contain GAMS source code required for declaring and generating the MPSGE input file. Figure 11 contains portions of the GEN file for the same model. Figure 12 shows the preprocessor-generated listing and symbol table which are always appended to the bottom of the GEN file. If a preprocessor error occurs, it can be helpful to consult the symbol table to track down the bug. Finally, Figure 13 shows the first page of scalar format MPSGE input file produced by HARBERGER.GEN. Normally, this file is written and then erased in the course of a GAMS run, although all or part of the file may be retained using the \$ECHOP: switch.

## 9. Algebraic Representations of Models in this Paper

### 9.1. THE HARBERGER MODEL

#### *Zero Net Profit*

The unit cost of production in sector  $A_{hs}$  is given by a nested Leontief-CES function defined over the cost of intermediate inputs and primary factors with ad-valorem taxes on factor demands. In equilibrium, the unit cost must be no less than the market price of output:

$$-\Pi_s = \sum_g p_g B_{gs} + \phi_s \left( \sum_f \alpha_{fs} (w_f (1 + t_{fs}))^{1-\sigma_s} \right)^{\frac{1}{1-\sigma_s}} - p_s \geq 0 \quad \forall s | \sigma_s \neq 1$$

and

$$-\Pi_s = \sum_g p_g B_{gs} + \phi_s \left( \prod_f (w_f (1 + t_{fs}))^{\alpha_{fs}} \right) - p_s \geq 0 \quad \forall s | \sigma_s = 1$$

#### *Income Balance for Government*

Government tax income ( $PT$ ) is determined by the value of tax revenue, calculated using activity levels, compensated demands, market prices and ad-valorem tax rates:

$$PT = \sum_s \frac{\partial \Pi_s}{\partial (w_f (1 + t_{fs}))} AL_s w_f t_{fs}$$

#### *Income Balance for Households*

Household income is determined by the net of tax return to primary factors plus the household share of government revenue:

$$M_h = \sum_f w_f E_{fh} + s_h PT$$

#### *Market Clearance for Goods*

Producer output is equal to the sum of intermediate plus final demand:

$$AL_g \geq \sum_s AL_s B_{gs} + \sum_h \frac{\gamma_h M_h}{e_h(p)} \left( \frac{e_h(p)}{p_g} \right)^{\sigma_h}$$

where  $\gamma_h$  is the household budget share devoted to the consumption of goods, and  $e_h h$  is the “unit expenditure function” which may be written:

$$e_h(p) \equiv \left( \sum_i \beta_{ih} p_i^{1-\sigma_h} \right)^{\frac{1}{1-\sigma_h}}$$

*Market Clearance for Factors*

The aggregate supply of factors equals the sum of producer and consumer demand. Producers pay taxes on factor inputs, consumers do not because we consider these demands to be “leisure” or “household production”. Consumer demands for factors are specified as Cobb-Douglas (constant budget shares):

$$\sum_h E_{fh} = \sum_s AL_s \frac{\partial \Pi_s}{\partial (w_f(1+t_{fs}))} + \sum_h \frac{\mu_{fh} M_h}{w_f}$$

## 9.2. THE SHOVEN MODEL

*Zero Net Profit*

The unit cost of production in sector  $A_h s$  is given by a nested Leontief-CES function defined over the cost of intermediate inputs and primary factors with ad-valorem taxes on factor demands. Unlike the Harberger models, tax rates in this model are determined endogenously. In equilibrium, the unit cost must be no less than the market price of output:

$$-\Pi_s = \sum_g p_g B_{gs} + \phi_s \left( \sum_f \alpha_{fs} (w_f(1+\tau_{fs}))^{1-\sigma_s} \right)^{\frac{1}{1-\sigma_s}} - p_s \geq 0 \quad \forall s | \sigma_s \neq 1$$

and

$$-\Pi_s = \sum_g p_g B_{gs} + \phi_s \left( \prod_f (w_f(1+\tau_{fs}))^{\alpha_{fs}} \right) - p_s \geq 0 \quad \forall s | \sigma_s = 1$$

*Income Balance for Government*

Government tax income ( $PT$ ) is determined by the value of tax revenue, calculated using activity levels, compensated demands, market prices and ad-valorem tax rates:

$$PT = \sum_s \frac{\partial \Pi_s}{\partial (w_f(1+\tau_{fs}))} AL_s w_f \tau_{fs}$$

*Equal Yield*

In equilibrium, tax rates are multiplicatively adjusted to achieve a target level of government revenue:

$$PT = \sum_g \theta_g p_g$$

*Income Balance for Households*

Household income is determined by the net of tax return to primary factors plus the household share of government revenue:

$$M_h = \sum_f w_f E_{fh} + s_h PT$$

*Market Clearance for Goods*

Producer output is equal to the sum of intermediate plus final demand:

$$AL_g \geq \sum_s AL_s B_{gs} + \sum_h \frac{\gamma_h M_h}{e_h(p)} \left( \frac{e_h(p)}{p_g} \right)^{\sigma_h}$$

where  $\gamma_h$  is the household budget share devoted to the consumption of goods, and  $e_h$  is the “unit expenditure function” which may be written:

$$e_h(p) \equiv \left( \sum_i \beta_{ih} p_i^{1-\sigma_h} \right)^{\frac{1}{1-\sigma_h}}$$

*Market Clearance for Factors*

The aggregate supply of factors equals the sum of producer and consumer demand. Producers pay taxes on factor inputs, consumers do not because we consider these demands to be “leisure” or “household production”. Consumer demands for factors are specified as Cobb-Douglas (constant budget shares):

$$\sum_h E_{fh} = \sum_s AL_s \frac{\partial \Pi_s}{\partial (w_f (1 + \tau t_{fs}))} + \sum_h \frac{\mu_{fh} M_h}{w_f}$$

## 9.3. THE SAMUELSON MODEL

*Zero Net Profit for Private Production*

The unit cost of production in sector  $A_{hs}$  is given by a nested Leontief - Cobb-Douglas function defined over the cost of intermediate inputs and primary factors with ad-valorem taxes on factor demands. Unlike the Harberger models, tax rates in this model are determined endogenously. In equilibrium, the unit cost must be no less than the market price of output:

$$-\Pi_s = \sum_g p_g B_{bs} + \phi_s \left( \prod_f (w_f (1 + \tau t_{fs}))^{\alpha_{fs}} \right) - p_s \geq 0 \quad \forall s$$

*Zero Net Profit for Public Sector Contractors*

The unit cost of public provision is determined by the market price of commodity inputs to the Leontief activity. Input requirements are defined by a vector of public sector input coefficients,  $a_{hg}$ . In equilibrium, the price paid by the government equals the cost of market inputs:

$$\sum_g p_g a_{hg} - p_G \geq 0$$

*Income Balance for Government*

Government tax income ( $PT$ ) is determined by the value of tax revenue, calculated using activity levels, compensated demands, market prices and ad-valorem tax rates. In equilibrium, the value of tax revenue equals the market cost of public sector output:

$$PT = \sum_s \frac{\partial \Pi_s}{\partial (w_f (1 + \tau t_{fs}))} AL_s w_f \tau t_{fs} = p_G G$$

*Income Balance for Households*

Household income is determined by the net of tax return to primary factors plus the imputed value of public provision:

$$M_h = \sum_f w_f E_{fh} + v_h G$$

*Market Clearance for Private Goods*

Producer output is equal to the sum of intermediate plus final demand:

$$AL_g \geq \sum_s AL_s B_{gs} + \sum_h \frac{\gamma_h M_h}{e_h(p)} \left( \frac{e_h(p)}{p_g} \right)^{\sigma_h}$$

where  $\gamma_h$  is the household budget share devoted to the consumption of goods, and  $e_h(p)$  is the “unit expenditure function” which may be written:

$$e_h(p) \equiv \left( \sum_i \beta_{ih} p_i^{1-\sigma_h} \right)^{\frac{1}{1-\sigma_h}}$$

*Personalized Markets for Public Goods*

We assume a “pure” public good in this model, hence each household may attach a different marginal valuation to public provision in an equilibrium. In order to compute these marginal values, we include a separate public good “market” for each household which balances the level of provision with the household “demand”:

$$G = \frac{\mu_{Gh} M_h}{v_h}$$

in which  $\mu_{Gh}$  is the budget share of public goods in the top-level Cobb-Douglas preferences of household  $h$ .

*Market Clearance for Factors*

The aggregate supply of factors equals the sum of producer and consumer demand. Producers pay taxes on factor inputs, consumers do not because we consider these demands to be “leisure” or “household production”. Consumer demands for factors are specified as Cobb-Douglas (constant budget shares):

$$\sum_h E_{fh} = \sum_s AL_s \frac{\partial \Pi_s}{\partial (w_f (1 + \tau t_{fs}))} + \sum_h \frac{\mu_{fh} M_h}{w_f}$$

*Samuelson Rule for “Optimal” Provision of Public Goods*

The tax rate multiplier is adjusted to balance the marginal cost of public provision with the summation across households of marginal willingness to pay. Due to the existence of household factor demand, factor taxes are necessarily distortionary and there will be an excess social cost of public funds. For this reason, the Samuelson rule is neither necessary nor sufficient for optimal provision. We apply the rule here merely to illustrate the programming methodology, even though the resulting equilibrium may be “suboptimal”:

$$p_G = \sum_h v_h$$

# CES Preferences and Technology

## *A Practical Introduction*

**Abstract.** This chapter describes practical approaches to the representation of constant elasticity demand functions in computable equilibrium models. A number of examples are presented which relate the mathematical derivation of these functional forms to their representation in MPSGE models.



## 1. The Basics

In many economic textbooks the constant-elasticity-of-substitution (CES) utility function is defined as:

$$U(x, y) = (\alpha x^\rho + (1 - \alpha)y^\rho)^{1/\rho}$$

It is a tedious but straight-forward application of Lagrangian calculus to demonstrate that the associated demand functions are:

$$x(p_x, p_y, M) = \left(\frac{\beta}{p_x}\right)^\sigma \frac{M}{\beta^\sigma p_x^{1-\sigma} + (1 - \alpha)^\sigma p_y^{1-\sigma}}$$

and

$$y(p_x, p_y, M) = \left(\frac{1 - \alpha}{p_y}\right)^\sigma \frac{M}{\alpha^\sigma p_x^{1-\sigma} + (1 - \alpha)^\sigma p_y^{1-\sigma}}.$$

The corresponding indirect utility function has is:

$$V(p_x, p_y, M) = M (\alpha^\sigma p_x^{1-\sigma} + (1 - \alpha)^\sigma p_y^{1-\sigma})^{\frac{1}{\sigma-1}}$$

Note that  $U(x, y)$  is linearly homogeneous:

$$U(\lambda x, \lambda y) = \lambda U(x, y)$$

This is a convenient cardinalization of utility, because percentage changes in  $U$  are equivalent to percentage Hicksian equivalent variations in income. Because  $U$  is linearly homogeneous,  $V$  is homogeneous of degree one in  $M$ :

$$V(p_x, p_y, \lambda M) = \lambda V(p_x, p_y, M)$$

and  $V$  is homogeneous of degree -1 in  $p$ .

$$V(\lambda p_x, \lambda p_y, M) = \frac{V(p_x, p_y, M)}{\lambda}.$$

In the representation of technology, we have an analogous set of relationships, based on the cost and compensated demand functions. If we have a CES production function of the form:

$$y(K, L) = \gamma (\alpha K^\rho + (1 - \alpha)L^\rho)^{1/\rho}$$

the unit cost function then has the form:

$$c(p_K, p_L) = \frac{1}{\gamma} (\alpha^\sigma p_K^{1-\sigma} + (1 - \alpha)^\sigma p_L^{1-\sigma})^{\frac{1}{1-\sigma}}$$

and associated demand functions are:

$$K(p_K, p_L) = \left(\frac{y}{\gamma}\right) \left(\frac{\alpha \gamma c(p_K, p_L)}{p_K}\right)^\sigma$$

and

$$L(p_K, p_L) = \left(\frac{y}{\gamma}\right) \left(\frac{(1 - \alpha) \gamma c(p_K, p_L)}{p_L}\right)^\sigma.$$

In most large-scale applied general equilibrium models, we have many function parameters to specify with relatively few observations. The conventional approach is to

*calibrate* functional parameters to a single benchmark equilibrium. For example, if we have benchmark estimates for output, labor, capital inputs and factor prices, we calibrate function coefficients by inverting the factor demand functions:

$$\theta = \frac{\bar{p}_K \bar{K}}{\bar{p}_K \bar{K} + \bar{P}_L \bar{L}}, \quad \rho = \frac{\sigma - 1}{\sigma}, \quad \alpha = \theta \bar{K}^{-\rho}$$

and

$$\gamma = \bar{y} [\alpha \bar{K}^\rho + (1 - \alpha) \bar{L}^\rho]^{-1/\rho}$$

## EXERCISES

1. Mikki once lived in Boulder and spent 30% of her income for rent, 10% for food and 60% for skiing. She then moved to Georgetown where rent and food prices are identical to Boulder. In Georgetown, however, Mikki discovered that the quality-adjusted cost of skiing was ten-times the cost of skiing in Boulder. She adopted a lifestyle in which she spend only 30% of her income on skiing. Suppose that her preferences are characterized by a CES utility function. What values of  $\alpha$  and  $\sigma$  describe Mikki's utility function?
2. What fraction of Mikki's income does she spend on rent in Georgetown?
3. How much larger would Mikki's income need to be to compensate for the higher cost of skiing such that she would be indifferent between living in Boulder or Georgetown.

## 2. The Calibrated Share Form

Calibration formulae for CES functions are messy and difficult to remember. Consequently, the specification of function coefficients is complicated and error-prone. For applied work using calibrated functions, it is much easier to use the "calibrated share form" of the CES function. In the calibrated form, the cost and demand functions explicitly incorporate

- benchmark factor *demands*
- benchmark factor *prices*
- the elasticity of substitution
- benchmark *cost*
- benchmark *output*
- benchmark *value shares*

In this form, the production function is written:

$$y = \bar{y} \left[ \theta \left( \frac{K}{\bar{K}} \right)^\rho + (1 - \theta) \left( \frac{L}{\bar{L}} \right)^\rho \right]^{1/\rho}$$

The only *calibrated* parameter,  $\theta$ , represents the value share of capital at the benchmark point. The corresponding cost functions in the calibrated form is written:

$$c(p_K, p_L) = \bar{c} \left[ \theta \left( \frac{p_K}{\bar{p}_K} \right)^{1-\sigma} + (1-\theta) \left( \frac{p_L}{\bar{p}_L} \right)^{1-\sigma} \right]^{\frac{1}{1-\sigma}}$$

where

$$\bar{c} = \bar{p}_L \bar{L} + \bar{p}_K \bar{K}$$

and the compensated demand functions are:

$$K(p_K, p_L, y) = \bar{K} \frac{y}{\bar{y}} \left( \frac{\bar{p}_K c}{p_K \bar{c}} \right)^\sigma$$

and

$$L(p_K, p_L, y) = \bar{L} \frac{y}{\bar{y}} \left( \frac{c \bar{p}_L}{\bar{c} p_L} \right)^\sigma$$

Normalizing the benchmark utility index to unity, the utility function in calibrated share form is written:

$$U(x, y) = \left[ \theta \left( \frac{x}{\bar{x}} \right)^\rho + (1-\theta) \left( \frac{y}{\bar{y}} \right)^\rho \right]^{1/\rho}$$

Defining the unit expenditure function as:

$$e(p_x, p_y) = \left[ \theta \left( \frac{p_x}{\bar{p}_x} \right)^{1-\sigma} + (1-\theta) \left( \frac{p_y}{\bar{p}_y} \right)^{1-\sigma} \right]^{\frac{1}{1-\sigma}}$$

the indirect utility function is:

$$V(p_x, p_y, M) = \frac{M}{\bar{M} e(p_x, p_y)}$$

and the demand functions are:

$$x(p_x, p_y, M) = \bar{x} V(p_x, p_y, M) \left( \frac{e(p_x, p_y) \bar{p}_x}{p_x} \right)^\sigma$$

and

$$y(p_x, p_y, M) = \bar{y} V(p_x, p_y, M) \left( \frac{e(p_x, p_y) \bar{p}_y}{p_y} \right)^\sigma$$

The calibrated form extends directly to the  $n$ -factor case. An  $n$ -factor production function is written:

$$y = f(x) = \bar{y} \left[ \sum_i \theta_i \left( \frac{x_i}{\bar{x}_i} \right)^\rho \right]^{1/\rho}$$

and has unit cost function:

$$C(p) = \bar{C} \left[ \sum_i \theta_i \left( \frac{p_i}{\bar{p}_i} \right)^{1-\sigma} \right]^{\frac{1}{1-\sigma}}$$

and compensated factor demands:

$$x_i = \bar{x}_i \frac{y}{\bar{y}} \left( \frac{C \bar{p}_i}{\bar{C} p_i} \right)^\sigma$$

### EXERCISES

1. Show that given a generic CES utility function:

$$U(x, y) = (\alpha^\rho + (1 - \alpha)y^\rho)^{1/\rho}$$

can be represented in share form using:

$$\bar{x} = 1, \bar{y} = 1, \bar{p}_x = t\alpha, \bar{p}_y = t(1 - \alpha), \bar{M} = t.$$

for any value of  $t > 0$ .

2. Consider the utility function defined:

$$U(x, y) = (x - a)^\alpha (y - b)^{1-\alpha}$$

A benchmark demand point with both prices equal and demand for  $y$  equal to twice the demand for  $x$ . Find values for which are consistent with optimal choice at the benchmark. Select these parameters so that the income elasticity of demand for  $x$  at the benchmark point equals 1.1.

3. Consider the utility function:

$$U(x, L) = (\alpha L^\rho + (1 - \alpha)x^\rho)^{1/\rho}$$

which is maximized subject to the budget constraint:

$$p_x x = M + w(\bar{L} - L)$$

in which  $M$  is interpreted as non-wage income,  $w$  is the market wage rate. Assume a benchmark equilibrium in which prices for  $x$  and  $L$  are equal, demands for  $x$  and  $L$  are equal, and non-wage income equals one-half of expenditure on  $x$ . Find values of  $\alpha$  and  $\rho$  consistent with these choices and for which the price elasticity of labor supply equals 0.2.

4. Consider a consumer with CES preferences over two goods. A price change makes the benchmark consumption bundle unaffordable, yet the consumer is indifferent. Graph the choice. Find an equation which determines the elasticity of substitution as a function of the benchmark value shares. (You can write down the equation, but it cannot be solved in closed form.)
5. Consider a model with three commodities,  $x$ ,  $y$  and  $z$ . Preferences are CES. Benchmark demands and prices are equal for all goods. Find demands for  $x$ ,  $y$  and  $z$  for a doubling in the price of  $x$  as a function of the elasticity of substitution.

6. Consider the same model in the immediately preceding question, except assume that preferences are instead given by:

$$U(x, y, z) = (\beta \min(x, y)^\rho + (1 - \beta)z^\rho)^{1/\rho}$$

Determine  $\beta$  from the benchmark, and find demands for  $x$ ,  $y$  and  $z$  if the price of  $x$  doubles.

### 3. Flexibility and Non-Separable CES functions

We let  $\pi_i$  denote the user price of the  $i$ th input, and let  $x_i(\pi)$  be the cost-minimizing demand for the  $i$ th input. The reference price and quantities are  $\bar{\pi}_i$  and  $\bar{x}_i$ . One can think of set  $i$  as  $\{K, L, E, M\}$  but the methods we employ may be applied to any number of inputs. Define the reference cost, and reference value share for  $i$ th input by  $\bar{C}$  and  $\theta_i$ , where

$$\bar{C} \equiv \sum_i \bar{\pi}_i \bar{x}_i$$

and

$$\theta_i \equiv \frac{\pi_i \bar{x}_i}{\bar{C}}$$

The single-level constant elasticity of substitution cost function in "calibrated share form" is written:

$$C(\pi) = \bar{C} \left( \sum_i \theta_i \left( \frac{\pi_i}{\bar{\pi}_i} \right)^{1-\sigma} \right)^{\frac{1}{1-\sigma}}$$

Compensated demands may be obtained from Shephard's lemma:

$$x_i(\pi) = \frac{\partial C}{\partial \pi_i} \equiv C_i = \bar{x}_i \left( \frac{C(\pi)}{\bar{C}} \frac{\bar{\pi}_i}{\pi_i} \right)^\sigma$$

Cross-price Allen-Uzawa elasticities of substitution (AUES) are defined as:

$$\sigma_{ij} \equiv \frac{C_{ij} C}{C_i C_j}$$

where

$$C_{ij} \equiv \frac{\partial^2 C(\pi)}{\partial \pi_i \partial \pi_j} = \frac{\partial x_i}{\partial \pi_j} = \frac{\partial x_j}{\partial \pi_i}$$

For single-level CES functions:

$$\sigma_{ij} = \sigma \quad \forall i \neq j$$

The CES cost function exhibits homogeneity of degree one, hence Euler's condition applies to the second derivatives of the cost function (the Slutsky matrix):

$$\sum_j C_{ij}(\pi) \pi_j = 0$$

or, equivalently:

$$\sum_j \sigma_{ij} \theta_j = 0$$

The Euler condition provides a simple formula for the diagonal AUES values:

$$\sigma_{ii} = \frac{-\sum_{j \neq i} \sigma_{ij} \theta_j}{\theta_i}$$

As an aside, note that convexity of the cost function implies that all minors of order 1 are negative, i.e.  $\sigma_{ii} < 0 \quad \forall i$ . Hence, there must be *at least one* positive off-diagonal element in each row of the AUES or Slutsky matrices. When there are only two factors, then the off-diagonals must be negative. When there are three factors, then only one pair of negative goods may be complements.

Let:

- $k$  index a second-level nest
- $s_{ik}$  denote the fraction of good  $i$  inputs assigned to the  $k$ th nest
- $\omega_k$  denote the benchmark value share of total cost which enters through the  $k$ th nest
- $\gamma$  denote the top-level elasticity of substitution
- $\sigma^k$  denote the elasticity of substitution in the  $k$ th aggregate
- $p_k(\pi)$  denote the price index associated with aggregate  $k$ , normalized to equal unity in the benchmark, i.e.:

$$p_k(\pi) = \left[ \frac{\sum_i s_{ik} \theta_i}{\omega_k \left( \frac{\pi_i}{\bar{\pi}_i} \right)^{1-\sigma^k}} \right]^{\frac{1}{1-\sigma^k}}$$

The two-level nested, nonseparable constant-elasticity-of-substitution (NNCES) cost function is then defined as:

$$C(\pi) = \bar{C} \left( \sum_k \omega_k p_k(\pi)^{1-\gamma} \right)^{\frac{1}{1-\gamma}}$$

Demand indices for second-level aggregates are needed to express demand functions in a compact form. Let  $z_k(\pi)$  denote the demand index for aggregate  $k$ , normalized to unity in the benchmark; i.e.

$$z_k(\pi) = \left( \frac{C(\pi)}{\bar{C}} \frac{1}{p_k(\pi)} \right)^\gamma$$

Compensated demand functions are obtained by differentiating  $C(\pi)$ . In this derivative, one term arise for each nest in which the commodity enters, so:

$$x_i(\pi) = \bar{x}_i \sum_k z_k(\pi) \left( \frac{p_k(\pi) \bar{\pi}_i}{\pi_i} \right)^{\sigma^k} = \bar{x}_i \sum_k \left( \frac{C(\pi)}{\bar{C}} \frac{1}{p_k(\pi)} \right)^\gamma \left( \frac{p_k(\pi) \bar{\pi}_i}{\pi_i} \right)^{\sigma^k}$$

Simple differentiation shows that benchmark cross-elasticities of substitution have the form:

$$\sigma_{ij} = \gamma + \sum_k \frac{(\sigma^k - \gamma)s_{ik}s_{jk}}{\omega_k}$$

Given the benchmark value shares  $\theta_i$  and the benchmark cross-price elasticities of substitution,  $\sigma_{ij}$ , we can solve for values of  $s_{ik}$ ,  $\omega_k$ ,  $\sigma^k$  and  $\gamma$ . A closed-form solution of the calibration problem is not always practical, so it is convenient to compute these parameters using a constrained nonlinear programming algorithm, CONOPT, which is available through GAMS, the same programming environment in which the equilibrium model is specified. Perroni and Rutherford (EER, 1994) prove that calibration of the NNCES form is possible for arbitrary dimensions whenever the given Slutsky matrix is negative semi-definite. The two-level ( $N \times N$ ) function is flexible for three inputs; and although we have not proven that it is flexible for 4 inputs, the only difficulties we have encountered have resulted from indefinite calibration data points.

Two GAMS programs are listed below. The first illustrates two analytic calibrations of the three-factor cost function. The second illustrates the use of nonlinear programming to calibrate a four-factor cost function.

#### 4. Benchmarking Decreasing Returns to Scale Production Functions

This section describes how calibrate the fixed factor input for a constant returns to scale CES technology and obtain an arbitrary price elasticity of supply at a reference point. For concreteness, consider output as a function of labor and capital inputs. Consider the labor input to be variable and the capital input to be fixed. We then have a CES cost function which in equilibrium defines the price of output:

$$p = c(r, w)$$

in which  $w$  is the exogenous wage rate and  $r$  is the residual return to the sector's fixed factor. Because this factor is fixed, by Shepard's lemma we have the following relationship between output, the supply of the fixed factor and the return to the fixed factor:

$$y \frac{\partial c(r, w)}{\partial r} = \bar{R}$$

If we use the calibrated CES cost function of the form:

$$c(r, w) = (\theta r^{1-\sigma} + (1-\theta)w^{1-\sigma})^{\frac{1}{1-\sigma}}$$

then the calibration problem consists of finding a values for  $\theta$  and  $\sigma$  for which:

$$\frac{\partial y}{\partial(p/w)} \frac{(p/w)}{y} = \eta$$

at the benchmark point.

Note that we are free to choose units of the specific factor such that its benchmark price is unity. Hence, when we calibrate the share parameter, we are also determining the supply of the fixed factor:

$$\bar{R} = \theta \bar{y}$$

```

$TITLE  Two nonseparable CES calibrations for a 3-input cost function.

*      Model-specific data defined here:

SET    I  Production input aggregates / A,B,C /;  ALIAS (I,J);

PARAMETER

    THETA(I)  Benchmark value shares /A 0.2, B 0.5, C 0.3/

    AUES(I,J) Benchmark cross-elasticities (off-diagonals) /
                A.B      2
                A.C     -0.05
                B.C      0.5 /;

*      Use an analytic calibration of the three-factor CES cost
*      function:

ABORT$(CARD(I) NE 3) "Error: not a three-factor model!";

*      Fill in off-diagonals:

AUES(I,J)$AUES(J,I) = AUES(J,I);

*      Verify that the cross elasticities are symmetric:

ABORT$SUM((I,J), ABS(AUES(I,J)-AUES(J,I))) " AUES values non-symmetric?";

*      Check that all value shares are positive:

ABORT$(SMIN(I, THETA(I)) LE 0) " Zero value shares are not valid:",THETA;

*      Fill in the elasticity matrices:

AUES(I,I) = 0; AUES(I,I) = -SUM(J, AUES(I,J)*THETA(J))/THETA(I); DISPLAY AUES;

```

*Figure 1a.* Analytic Calibration: Define Benchmark Shares and Elasticities

in which we scale the benchmark price of output to unity.

If the relative price of output and the variable factor depart from their benchmark values, the supply constraint for sector-specific can be inverted to obtain an explicit expression for the return:

$$r = p \left( \frac{\theta y}{\bar{R}} \right)^{1/\sigma}$$

```

SET      N      Potential nesting /N1*N3/
        K(N)    Nesting aggregates used in the model
        I1(I)   Good fully assigned to first nest
        I2(I)   Good fully assigned to second nest
        I3(I)   Good split between nests;

SCALAR  ASSIGNED /0/;

PARAMETER
        ESUB(*,*)      Alternative calibrated elasticities
        SHR(*,I,N)     Alternative calibrated shares
        SIGMA(N)       Second level elasticities
        S(I,N)         Nesting assignments (in model)
        GAMMA          Top level elasticity (in model);

*      First the Leontief structure:

ESUB("LTF","GAMMA") = SMAX((I,J), AUES(I,J));
ESUB("LTF",N) = 0;
LOOP((I,J)$((AUES(I,J) EQ ESUB("LTF","GAMMA"))*(NOT ASSIGNED)),
      I1(I) = YES;
      I2(J) = YES;
      ASSIGNED = 1;
);

I3(I) = YES$((NOT I1(I))*(NOT I2(I)));
DISPLAY I1,I2,I3;
LOOP((I1,I2,I3),
      SHR("LTF",I1,"N1") = 1;
      SHR("LTF",I2,"N2") = 1;
      SHR("LTF",I3,"N1") = THETA(I1)*(1-AUES(I1,I3)/AUES(I1,I2)) /
                          ( 1 - THETA(I3) * (1-AUES(I1,I3)/AUES(I1,I2)) );
      SHR("LTF",I3,"N2") = THETA(I2)*(1-AUES(I2,I3)/AUES(I1,I2)) /
                          ( 1 - THETA(I3) * (1-AUES(I2,I3)/AUES(I1,I2)) );
      SHR("LTF",I3,"N3") = 1 - SHR("LTF",I3,"N1") - SHR("LTF",I3,"N2");
);
ABORT$(SMIN((I,N), SHR("LTF",I,N)) LT 0) "Benchmark AUES is indefinite.";

```

*Figure 1b.* Analytic Calibration: Nested Leontief Calibration

where we have substituted the equilibrium price for the cost function. Substituting back into the cost function, we have

$$p^{1-\sigma} = \theta p^{1-\sigma} \left( \frac{\theta y}{R} \right)^{\frac{1-\sigma}{\sigma}} + (1-\theta)w^{1-\sigma}$$

```

*      Now specify the two-level CES function:

ESUB("CES","GAMMA") = SMAX((I,J), AUES(I,J));
ESUB("CES","N1") = 0;
LOOP((I1,I2,I3),
    SHR("CES",I1,"N1") = 1;
    SHR("CES",I2,"N2") = 1;
    ESUB("CES","N2") = (AUES(I1,I2)*AUES(I1,I3)-AUES(I2,I3)*AUES(I1,I1)) /
        (AUES(I1,I3)-AUES(I1,I1));
    SHR("CES",I3,"N1") =
        (AUES(I1,I2)-AUES(I1,I3)) / (AUES(I1,I2)-AUES(I1,I1));
    SHR("CES",I3,"N2") = 1 - SHR("CES",I3,"N1");
);
ABORT$(SMIN(N, ESUB("CES",N)) LT 0) "Benchmark AUES is indefinite?";
ABORT$(SMIN((I,N), SHR("CES",I,N)) LT 0) "Benchmark AUES is indefinite?";

```

Figure 1c. Analytic Calibration: Nested CES Calibration

or

$$y = \bar{R}\theta^{\frac{1}{\sigma-1}} \left[ 1 - (1 - \theta) \left( \frac{w}{p} \right)^{1-\sigma} \right]^{\frac{\sigma}{1-\sigma}}$$

Differentiating this expression with respect the relative price of output, and setting all prices equal to unit, we have:

$$\eta = \frac{\sigma(1 - \theta)}{\theta}$$

This equation can be used in a variety of ways to calibrate the supply function. One approach would be to choose the value share of the fixed factor  $\theta$  to match the base year profits, and then assign the elasticity according to:

$$\sigma = \frac{\theta\eta}{(1 - \theta)}$$

Alternatively, one choose to use a Cobb-Douglas function and set the specific factor value share accordingly:

$$\theta = \frac{1}{1 + \eta}$$

## 5. Analytic Calibration of the GEMTAP Final Demand System

Following Ballard, Fullerton, Shoven and Whalley (BFSW), we consider a representative agent whose utility is based upon current consumption, future consumption and current leisure. Changes in *future consumption*; in this static framework are associated with changes in the level of savings. There are three prices which jointly determine the price index for future consumption. These are:

$P_I$  the composite price index for investment goods

```

PARAMETER          PRICE(I)          PRICE INDICES USING TO VERIFY CALIBRATION
                   AUESCHK(*,I,J)    CHECK OF BENCHMARK AUES VALUES;

PRICE(I) = 1;

$ontext

$MODEL:CHKCALIB

$SECTORS:
    Y          ! PRODUCTION FUNCTION
    D(I)

$COMMODITIES:
    PY         ! PRODUCTION FUNCTION OUTPUT
    P(I)       ! FACTORS OF PRODUCTION
    PFX        ! AGGREGATE PRICE LEVEL

$CONSUMERS:
    RA

$PROD:Y  s:GAMMA  K.TL:SIGMA(K)
          0:PY          Q:1
          I:P(I)#(K)    Q:(THETA(I)*S(I,K))    K.TL:

$PROD:D(I)
          0:P(I)  Q:THETA(I)
          I:PFX  Q:(THETA(I)*PRICE(I))

$DEMAND:RA
          D:PFX
          E:PFX  Q:2
          E:PY   Q:-1

$OFFTEXT
$SYSINCLUDE mpsgeset CHKCALIB

```

*Figure 1d.* Analytic Calibration: Simple MPSGE Model to Check Calibration

$P_K$  the composite rental price for capital services

$P_C$  the composite price of current consumption.

All of these prices equal unity in the benchmark equilibrium.

Capital income in each future year finances future consumption, which is expected to cost the same as in the current period,  $P_C$  (static expectations). The consumer demand

```

SCALAR DELTA /1.E-5/;

SET      FUNCTION /LTF, CES/;

ALIAS (I,II);

LOOP(FUNCTION,

      K(N) = YES$SUM(I, SHR(FUNCTION,I,N));
      GAMMA = ESUB(FUNCTION,"GAMMA");
      SIGMA(K) = ESUB(FUNCTION,K);
      S(I,K) = SHR(FUNCTION,I,K);

      LOOP(II,

            PRICE(J) = 1; PRICE(II) = 1 + DELTA;

$INCLUDE CHKCALIB.GEN
      SOLVE CHKCALIB USING MCP;

      AUESCHK(FUNCTION,J,II) = (D.L(J)-1) / (DELTA*THETA(II));

));

AUESCHK(FUNCTION,I,J) = AUESCHK(FUNCTION,I,J) - AUES(I,J);
DISPLAY AUESCHK;

```

*Figure 1e.* Analytic Calibration: Use the MPSGE to Verify Calibration

for savings therefore depends not only on  $P_I$ , but also on  $P_K$  and  $P_C$ , namely:

$$P_S = \frac{P_I P_C}{P_K}$$

The price index for savings is unity in the benchmark period. In a counter-factual equilibrium, however, we would expect generally that

$$P_S \neq P_I$$

. When these price indices are not equal, there is a *virtual tax payment*; associated with savings demand.

Following BFSW, we adopt a nested CES function to represent preferences. In this function, at the top level demand for savings (future consumption) trades off with a second CES aggregate of leisure and current consumption. These preferences can be summarized with the following expenditure function:

$$P_U = [\alpha P_H^{1-\sigma_S} + (1-\alpha)P_S^{1-\sigma_S}]^{\frac{1}{1-\sigma_S}}$$

Preferences are homothetic, so we have defined  $P_U$  as a linearly homogeneous cost index for a unit of utility. We conveniently scale this price index to equal unity in the

```

*      Evaluate the demand functions:

$LIBINCLUDE gnuplot
SET PR Alternative price levels /PRO*PR10/;

PARAMETER
    DEMAND(FUNCTION,I,PR)    Demand functions
    DPLOT(PR,FUNCTION)       Plotting output array;

LOOP(II,
    LOOP(FUNCTION,
        K(N) = YES$SUM(I, SHR(FUNCTION,I,N));
        GAMMA = ESUB(FUNCTION,"GAMMA");
        SIGMA(K) = ESUB(FUNCTION,K);
        S(I,K) = SHR(FUNCTION,I,K);
        LOOP(PR,
            PRICE(J) = 1;
            PRICE(II) = 0.2 * ORD(PR);
$INCLUDE CHKCALIB.GEN
            SOLVE CHKCALIB USING MCP;
            DEMAND(FUNCTION,II,PR) = D.L(II);
            DPLOT(PR,FUNCTION) = D.L(II);
        );
    );

$LIBINCLUDE gnuplot DPLOT

);

DISPLAY DEMAND;

```

Figure 1f. Analytic Calibration: Compare Demand Functions

benchmark. In this definition,  $\alpha$  is the benchmark value share for current consumption (goods and leisure).  $P_H$  is a composite price for current consumption defined as:

$$P_H = [\beta P_\ell^{1-\sigma_L} + (1-\beta)P_C^{1-\sigma_L}]^{\frac{1}{1-\sigma_L}}$$

in which  $\beta$  is the benchmark value share for leisure within current consumption.

Demand functions are:

$$S = S_0 \left( \frac{P_U}{P_F} \right)^{\sigma_S} \frac{I}{I_0 P_U},$$

$$C = C_0 \left( \frac{P_H}{P_C} \right)^{\sigma_L} \left( \frac{P_U}{P_H} \right)^{\sigma_S} \frac{I}{I_0 P_U},$$

and

$$\ell = \ell_0 \left( \frac{P_H}{P_L} \right)^{\sigma_L} \left( \frac{P_U}{P_H} \right)^{\sigma_S} \frac{I}{I_0 P_U}.$$

```

$TITLE Numerical calibration of Nested CES from KLEM elasticities

SET I Production input aggregates / K, L, E, M/; ALIAS (I,J);

* Model-specific data defined here:

PARAMETER
  THETA(I) Benchmark value shares /K 0.2, L 0.4, E 0.05, M 0.35/

  AUES(I,J) Benchmark cross-elasticities (off-diagonals) /
              K.L      1
              K.E     -0.1
              K.M      0
              L.E      0.3
              L.M      0
              E.M      0.1 /;

SCALAR EPSILON Minimum value share tolerance /0.001/;

* Fill in off-diagonals:

AUES(I,J)$AUES(J,I) = AUES(J,I);

* Verify that the cross elasticities are symmetric:

ABORT$SUM((I,J), ABS(AUES(I,J)-AUES(J,I))) " AUES values non-symmetric?";

* Check that all value shares are positive:

ABORT$(SMIN(I, THETA(I)) LE 0) " Zero value shares are not valid:",THETA;

* Fill in the elasticity matrices:

AUES(I,I) = 0; AUES(I,I) = -SUM(J, AUES(I,J)*THETA(J))/THETA(I); DISPLAY AUES;

```

Figure 2a. Numerical Calibration: Define Benchmark Shares and Elasticities

Demands are written here in terms of their benchmark values ( $S_0$ ,  $C_0$  and  $\ell_0$ ) and current and benchmark income ( $I$  and  $I_0$ ).

There are four components in income. The first is the value of labor endowment ( $E$ ), defined inclusive of leisure. The second is the value of capital endowment ( $K$ ). The third is all other income ( $M$ ). The fourth is the value of *virtual tax revenue*; associated with differences between the shadow price of savings and the cost of investment.

$$I = P_L E + P_K K + M + (P_S - P_I) S$$

The following parameter values are specified exogenously:

```

*      Define variables and equations for NNCES calibration:

SET      N          Nests within the two-level NNCES function /N1*N4/,
        K(N)       Nests which are in use;

VARIABLES
        S(I,N)     Fraction of good I which enters through nest N,
        SHARE(N)   Value share of nest N,
        SIGMA(N)   Elasticity of substitution within nest N,
        GAMMA      Elasticity of substitution at the top level,
        OBJ        Objective function;

POSITIVE VARIABLES S, SHARE, SIGMA, GAMMA;

EQUATIONS
        SDEF(I)    Nest shares must sum to one,
        TDEF(N)    Nest share in total cost,
        ELAST(I,J) Consistency with given AUES values,
        OBJDEF     Maximize concentration;

ELAST(I,J)$ (ORD(I) GT ORD(J))..
                AUES(I,J) =E=  GAMMA +
                SUM(K, (SIGMA(K)-GAMMA)*S(I,K)*S(J,K)/SHARE(K));

TDEF(K)..      SHARE(K) =E= SUM(I, THETA(I) * S(I,K));

SDEF(I)..      SUM(N, S(I,N)) =E= 1;

*      Maximize concentration at the same time keeping the elasticities
*      to be reasonable:

OBJDEF..      OBJ =E= SUM((I,K),S(I,K)*S(I,K))
                - SQR(GAMMA) - SUM(K, SQR(SIGMA(K)));

MODEL  CESCALIB /ELAST, TDEF, SDEF, OBJDEF/;

*      Apply some bounds to avoid divide by zero:

SHARE.LO(N) = EPSILON;

```

Figure 2b. Numerical Calibration: Define a Nonlinear Program for Calibration

```

SCALAR SOLVED Flag for having solved the calibration problem /0/
      MINSHR Minimum share in candidate calibration;

SET     TRIES Counter on the number of attempted calibrations /T1*T10/;

OPTION SEED=0;

LOOP(TRIES$(NOT SOLVED),

*       Initialize the set of active nests and the bounds:

      K(N) = YES;
      S.LO(I,N) = 0;           S.UP(I,N) = 1;
      SHARE.LO(N) = EPSILON;  SHARE.UP(N) = 1;
      SIGMA.LO(N) = 0;        SIGMA.UP(N) = +INF;

*       Install a starting point:

      SHARE.L(K) = MAX(UNIFORM(0,1), EPSILON);
      S.L(I,K)   = UNIFORM(0,1);
      GAMMA.L    = UNIFORM(0,1);
      SIGMA.L(K) = UNIFORM(0,1);
      SDEF.M(I)  = 0; TDEF.M(K)   = 0; ELAST.M(I,J) = 0;

      SOLVE CESCALIB USING NLP MAXIMIZING OBJ;

      SOLVED = 1$(CESCALIB.MODELSTAT LE 2);
      IF (SOLVED,
        MINSHR = SMIN(K, SHARE.L(K)) - EPSILON;
        IF (MINSHR EQ 0,
          K(N)$(SHARE.L(N) EQ EPSILON) = NO;
          S.FX(I,N)$(NOT K(N)) = 0;
          SHARE.FX(N)$(NOT K(N)) = 0;
          SIGMA.FX(N)$(NOT K(N)) = 0;
          DISPLAY "Recalibrating with the following nests:",K;
          SOLVE CESCALIB USING NLP MAXIMIZING OBJ;
          IF (CESCALIB.MODELSTAT GT 2, SOLVED = 0;);
          MINSHR = SMIN(K, SHARE.L(K)) - EPSILON;
          IF (MINSHR EQ 0, SOLVED = 0;);
        );
      );
);
IF (SOLVED, DISPLAY "Function calibrated:",GAMMA.L,SIGMA.L,SHARE.L,S.L;
ELSE      DISPLAY "Function calibration fails!";
);

```

Figure 2c. Numerical Calibration: Perform the Calibration

$\zeta = 1.75$  is the ratio of labor endowment to labor supply,  $\zeta \equiv \frac{E}{L_0}$ , where  $L_0$  is the benchmark labor supply. Labor supply and  $\zeta$  also define benchmark leisure demand,  $\ell_0 = L_0(\zeta - 1)$ .

$\xi = 0.15$  is the uncompensated elasticity of labor supply with respect to the net of tax wage, i.e.

$$\xi = \frac{\partial L}{\partial P_L} \frac{P_L}{L} = \frac{\partial(E - \ell)}{\partial P_L} \frac{P_L}{L} = -\frac{\partial \ell}{\partial P_L} \frac{P_L}{L}$$

$\eta = 0.4$  is the elasticity of savings with respect to the return to capital:

$$\eta \equiv \frac{\partial S}{\partial P_K} \frac{P_K}{S}$$

Shephard's lemma applied at benchmark prices provides the following identities which are helpful in deriving expressions for  $\eta$  and  $\xi$ :

$$\frac{\partial P_U}{\partial P_H} = \alpha, \quad \frac{\partial P_U}{\partial P_S} = 1 - \alpha, \quad \frac{\partial P_H}{\partial P_L} = \beta, \quad \frac{\partial P_H}{\partial P_C} = 1 - \beta$$

It is then a relatively routine application of the chain rule to show that:

$$\xi = (\zeta - 1) \left[ \sigma_L + \beta(\sigma_S - \sigma_L) - \alpha\beta(\sigma_S - 1) - \frac{E}{I_0} \right]$$

and

$$\eta = \sigma_S \alpha + \frac{K}{I_0}$$

The expression for  $\eta$  does not involve  $\sigma_L$ , so we may first solve for  $\sigma_S$  and use this value in determining  $\sigma_L$ :

$$\sigma_S = \frac{\eta - \frac{K}{I_0}}{\alpha}$$

and

$$\alpha_L = \frac{\frac{\xi}{\zeta - 1} - \sigma_S \beta(1 - \alpha) - \alpha\beta + \frac{E}{I_0}}{1 - \beta}$$

\$TITLE A Maquette Illustrating Labor Supply / Savings Demand Calibration

\* Exogenous elasticity:

SCALAR XI UNCOMPENSATED ELASTICITY OF LABOR SUPPLY /0.15/,  
ETA ELASTICITY OF SAVINGS WRT RATE OF RETURN /0.40/,  
ZETA RATIO OF LABOR ENDOWMENT TO LABOR SUPPLY /1.75/;

\* Benchmark data:

SCALAR CO CONSUMPTION /2.998845E+2/,  
SO SAVINGS /70.02698974/,  
LSO LABOR SUPPLY / 2.317271E+2/,  
KO CAPITAL INCOME /93.46960577/,  
PLO MARGINAL WAGE /0.60000000/;

*Figure 3a.* Labor Supply and Savings Demand: Input Data

```

*      Calibrated parameters:

SCALAR  ELO      LABOR ENDOWMENT
        LO      LEISURE DEMAND
        MO      NON-WAGE INCOME
        I       EXTENDED GROSS INCOME
        ETAMIN  SMALLEST PERMISSIBLE VALUE FOR ETA,
        XIMIN  SMALLEST PERMISSIBLE VALUE FOR XI,
        ALPHA  CURRENT CONSUMPTION VALUE SHARE
        BETA   LEISURE VALUE SHARE IN CURRENT CONSUMPTION
        SIGMA_L ELASTICITY OF SUBSTITUTION WITHIN CURRENT CONSUMPTION
        SIGMA_S ELASTICITY OF SUBSTITUTION - SAVINGS VS CURRENT CONSUMPTION
        TS     SAVINGS PRICE ADJUSTMENT;

LSO = LSO * PLO;
ELO = ZETA * LSO;
LO = ELO - LSO;
MO = CO + SO - LSO - KO;
I = LO + CO + SO;
BETA = LO / (CO + LO);
ALPHA = (LO + CO) / I;
SIGMA_S = (ETA - KO / I) / ALPHA;
ETAMIN = KO / I;
ABORT$(SIGMA_S LT 0) " Error: cannot calibrate SIGMA_S", ETAMIN;

SIGMA_L = (XI*(LSO/LO)-SIGMA_S*BETA*(1-ALPHA)-ALPHA*BETA+ELO/I)/(1-BETA);
XIMIN = -(LO/LSO) * (- SIGMA_S * BETA * (1-ALPHA) - ALPHA*BETA + ELO/I);
ABORT$(SIGMA_L LT 0) " Error: cannot calibrate SIGMA_L", XIMIN;

DISPLAY "Calibrated elasticities:", SIGMA_S, SIGMA_L;

```

*Figure 3b.* Labor Supply and Savings Demand: Calibration Assignments

```

$ONTEXT

$MODEL:CHKCAL

$COMMODITIES:
    PL
    PK
    PC
    PS

$SECTORS:
    Y
    S

$CONSUMERS:
    RA

$PROD:Y
    O:PC    Q:(KO+LSO-SO)
    I:PL    Q:(LSO-SO)
    I:PK    Q:KO

$PROD:S
    O:PS    A:RA    T:TS
    I:PL

$DEMAND:RA  s:SIGMA_S  a:SIGMA_L
    E:PC    Q:MO
    E:PL    Q:ELO
    E:PK    Q:KO
    D:PS    Q:S0
    D:PC    Q:CO    a:
    D:PL    Q:L0    a:

$OFFTEXT
$SYSINCLUDE mpsgeset CHKCAL

S.L = S0;
TS = 0;

*          VERIFY THE BENCHMARK:

CHKCAL.ITERLIM = 0;
$INCLUDE CHKCAL.GEN
SOLVE CHKCAL USING MCP;

```

Figure 3c. Labor Supply and Savings Demand: Maquette Incorporating the Data

```
PL.L = 1.001;

CHKCAL.ITERLIM = 0;
$INCLUDE CHKCAL.GEN
SOLVE CHKCAL USING MCP;

*      Compute induced changes in labor supply using the labor market
*      "marginal", PL.M. This marginal returns the net excess supply of
*      labor at the given prices. We started from a balanced benchmark,
*      with no change in labor demand (the iteration limit was zero).
*      Hence, PL.M returns the magnitude of the change in labor supply.
*      We multiply by the benchmark wage (1) and divide by the benchmark
*      labor supply (LS0) to produce a finite difference approximation
*      of the elasticity:

DISPLAY "CALIBRATION CHECK -- THE FOLLOWING VALUES SHOULD BE IDENTICAL:", XI;
XI = (PL.M / 0.001) * (1 / LS0);
DISPLAY XI;
```

*Figure 3d.* Labor Supply and Savings Demand: Check the Labor Supply Elasticity

```

PL.L = 1.0;

*      CHECK THE ELASTICITY OF SAVINGS WRT RENTAL RATE OF CAPITAL:

PK.L = 1.001;
PS.L = 1 / 1.001;
TS   = 1 / 1.001 - 1;

CHKCAL.ITERLIM = 0;

*      Compute elasticity of savings with respect to the rental rate of
*      capital. This requires some recursion in order to account for the
*      effect of changes in savings on effective income. When PK increases,
*      PS declines -- there is an effective "subsidy" for saving, paid from
*      consumer income. In order to obtain a difference approximation for
*      the elasticity of savings response, we need to make sure the virtual
*      tax payments are properly handled. In the MPSGE model, this means
*      that the level value for S must be adjusted so that it exactly equals
*      the savings. We do this recursively:

SET ITER /IT1*IT5/;

PS.M = 1;
LOOP(ITER$(ABS(PS.M) GT 1.0E-8),

$INCLUDE CHKCAL.GEN
      SOLVE CHKCAL USING MCP;
      S.L = S.L - PS.M;
);

DISPLAY "CALIBRATION CHECK -- THE FOLLOWING VALUES SHOULD BE IDENTICAL:", ETA;
ETA = ((S.L - SO) / 0.001) * (1 / SO);
DISPLAY ETA;

```

*Figure 3e.* Labor Supply and Savings Demand: Check the Savings Elasticity

## 6. Alternative Elasticities and Functional Forms

A well known dimensionless index of second-order curvature is the compensated price elasticity (CPE), which is defined as:

$$\sigma_{ij}^C \equiv \frac{\partial \ln C_i}{\partial \ln p_j} = \frac{C_{ij} p_j}{C_i},$$

and a related measure of second-order curvature is the AUES, which has been already discussed. This can also be written as

$$\sigma_{ij}^A = \frac{\sigma_{ij}^C}{\theta_j}.$$

The AUES is a *one-input-one-price* elasticity of substitution (Mundlak, 1968), since, as the definition of  $\sigma_{ij}^A$  makes clear, it measures the responsiveness of the compensated demand for one input to a change in one input price. In contrast, the Morishima elasticity of substitution (MES; Morishima, 1967) constitutes a *two-input-one-price* elasticity measure, being defined as

$$\sigma_{ij}^M \equiv \frac{\partial \ln(C_i/C_j)}{\partial \ln(p_i/p_j)} = \sigma_{ij}^C - \sigma_{jj}^C.$$

Note that, in general, the MES is not symmetric, i.e.  $\sigma_{ij}^M \neq \sigma_{ji}^M$ .

A third type of curvature measure is represented by the class of *two-input-two-price* elasticities of substitution, which take the form  $\partial \ln(C_i/C_j)/\partial \ln(p_j/p_i)$ . One such index is the shadow elasticity of substitution (SES; Frenger, 1985), which is defined as

$$\sigma_{ij}^S \equiv \frac{\theta_i \sigma_{ij}^M + \theta_j \sigma_{ji}^M}{(\theta_i + \theta_j)}.$$

When technologies are of the CES type,  $\sigma_{ij}^A$ ,  $\sigma_{ij}^M$  and  $\sigma_{ij}^S$  are all identical, but they are generally different otherwise.

### 6.1. THE TRANSLOG COST FUNCTION

The Translog unit cost function is defined as

$$\ln C(p) \equiv \ln b_0 + \sum_i b_i \ln p_i + \frac{1}{2} \sum_{ij} a_{ij} \ln p_i \ln p_j \equiv \ln b_0 + L(p).$$

*Compensated Demand Functions:*

$$x_i(p) = \frac{\partial C(p)}{\partial p_i} = \frac{\partial \ln C(p)}{\partial p_i} C = \frac{C(p)}{p_i} \left[ b_i + \sum_j a_{ij} \ln p_j \right]$$

*Restrictions:*

$$\sum_i b_i = 1;$$

$$a_{ij} = a_{ji}, \quad \forall i, \forall j;$$

$$\sum_j a_{ij} = 0, \quad \forall i.$$

*Calibration:*

$$a_{ij} = \theta_i \theta_j (\sigma_{ij}^A - 1), \quad i \neq j;$$

$$a_{ii} = - \sum_{j \neq i} a_{ij}, \quad \forall i;$$

$$b_i = \theta_i - \sum_j a_{ij} \ln p_j, \quad \forall i;$$

$$b_0 = \bar{C} e^{-L(p)}.$$

## 6.2. THE GENERALIZED LEONTIEF COST FUNCTION

The Generalized Leontief unit cost function is defined as

$$C(p) \equiv \frac{1}{2} \sum_{ij} a_{ij} \sqrt{p_i p_j}.$$

*Compensated Demand Functions:*

$$x_i(p) = \frac{\partial C(p)}{\partial p_i} = \sum_j \frac{a_{ij}}{2} \sqrt{\frac{p_j}{p_i}}.$$

*Restrictions:*

$$a_{ij} = a_{ji}, \quad \forall i, \forall j.$$

*Calibration:*

$$a_{ij} = 4\sigma_{ij}^A \theta_i \theta_j \frac{\bar{C}}{\sqrt{p_i p_j}}, \quad i \neq j;$$

$$a_{ii} = 2\theta_i \frac{\bar{C}}{p_i} - \sum_{j \neq i} a_{ij} \sqrt{\frac{p_j}{p_i}} \quad \forall i.$$

## 6.3. THE NORMALIZED QUADRATIC COST FUNCTION

The Normalized Quadratic unit cost function is defined as

$$C(p) \equiv \frac{1}{2} \frac{\sum_{ij} a_{ij} p_i p_j}{\sum_i b_i p_i}.$$

*Compensated Demand Functions:*

$$x_i(p) = \frac{\partial C(p)}{\partial p_i} = \frac{\sum_j a_{ij} p_j - C(p) b_i}{\sum_j b_j p_j}$$

*Restrictions:*

$$\begin{aligned} a_{ij} &= a_{ji}, & \forall i, \forall j; \\ b_i &\geq 0, & \forall i; \\ \sum_i b_i &= 1. \end{aligned}$$

*Calibration:*

$$a_{ij} = \frac{\bar{C}\theta_i\theta_j}{p_i p_j} \left( \sigma_{ij}^A \sum_k b_k p_k + \frac{b_i p_i}{\theta_i} + \frac{b_j p_j}{\theta_j} \right), \quad \forall i, \forall j;$$

We examined two alternative specifications, one in which  $b_i = \theta_i$ , and another in which  $b_i = 1/N$ . The first specification is reported by Perroni and Rutherford [1998] to produce a more stable function.

#### 6.4. THE NONSEPARABLE NESTED CES COST FUNCTION

We restrict our discussion to the case  $N = 3$  (for the general  $N$ -input case see Perroni and Rutherford, 1995), and focus on a particular nesting structure which we call ‘‘Lower Triangular Leontief’’ (LTL). Let us rearrange indices so that the maximum off-diagonal AUES element is  $\sigma_{12}^A$ . Then the three-input NNCES-LTL cost function can be defined as

$$C(p) \equiv \phi \left[ \alpha (a_1 p_1 + a_3 p_3)^{1-\gamma} + (1-\alpha) (b_2 p_2^{1-\mu} + b_3 p_3^{1-\mu})^{\frac{1-\gamma}{1-\mu}} \right]^{\frac{1}{1-\gamma}}.$$

*Compensated Demand Functions:* We simplify the algebra by defining price indices for the two nests:

$$p_{13} = a_1 p_1 + a_3 p_3$$

and

$$p_{23} = \left[ b_2 p_2^{1-\mu} + b_3 p_3^{1-\mu} \right]^{\frac{1}{1-\mu}}$$

we have:

$$\begin{aligned} x_1(p) &= \frac{\partial C(p)}{\partial p_1} = a_1 \alpha \phi \left( \frac{C(p)}{\phi p_{13}} \right)^\gamma \\ x_2(p) &= \frac{\partial C(p)}{\partial p_2} = b_2 (1-\alpha) \phi \left( \frac{C(p)}{\phi p_{23}} \right)^\gamma \left( \frac{\phi p_{23}}{p_2} \right)^\mu \\ x_3(p) &= \frac{\partial C(p)}{\partial p_3} = a_3 \alpha \phi \left( \frac{C(p)}{\phi p_{13}} \right)^\gamma + b_3 (1-\alpha) \phi \left( \frac{C(p)}{\phi p_{23}} \right)^\gamma \left( \frac{\phi p_{23}}{p_3} \right)^\mu \end{aligned}$$

*Restrictions:*

$$\begin{aligned} \gamma &\geq 0; \\ \mu &\geq 0; \\ \phi &\geq 0; \end{aligned}$$

$$\begin{aligned} a_i &\geq 0, & \forall i; \\ b_i &\geq 0, & \forall i. \end{aligned}$$

*Calibration:*

Let us denote with  $s_3$  the fraction of the total input of commodity 3 which enters the first subnest of the structure:

$$C(p) \equiv \phi \left[ \alpha (a_1 p_1 + a_3 p_3)^{1-\gamma} + (1-\alpha) (b_2 p_2^{1-\mu} + b_3 p_3^{1-\mu})^{\frac{1-\gamma}{1-\mu}} \right]^{\frac{1}{1-\gamma}}.$$

(with  $(1-s_3)$  representing the fraction entering the second subnest). If we select

$$\begin{aligned} \gamma &= \sigma_{12}^A; \\ \mu &= \frac{\sigma_{12}^A \sigma_{13}^A - \sigma_{23}^A \sigma_{11}^A}{\sigma_{13}^A - \sigma_{11}^A}; \end{aligned}$$

it can be shown that

$$s_3 = \frac{\sigma_{12}^A - \sigma_{13}^A}{\sigma_{12}^A - \sigma_{11}^A}.$$

The remaining parameters can then be recovered as follows:

$$\begin{aligned} \phi &= \bar{C}; \\ \alpha &= \theta_1 + s_3 \theta_3 \\ a_1 &= \frac{\theta_1}{\alpha p_1} \\ a_3 &= \frac{s_3 \theta_3}{\alpha p_3} \\ b_2 &= \frac{\theta_2}{(1-\alpha) p_2^{1-\mu}} \\ b_3 &= \frac{(1-s_3) \theta_3}{(1-\alpha) p_3^{1-\mu}} \end{aligned}$$



# Extensions of GAMS for Complementarity Problems in Economics

**Abstract.** This paper introduces new features of the GAMS modeling language which have been developed for solving nonlinear complementarity problems. The paper defines the "mixed complementarity problem" (MCP) and its various manifestations. Complementarity formulations for three economic models are developed, and computational benchmarks are presented for two large-scale MCP solvers. Finally, procedures for local sensitivity analysis are described.



## 1. Introduction

GAMS (General Algebraic Modelling System) is a computer language which was originally developed to assist economists at the World Bank in the quantitative analysis of economic policy questions (Meeraus (1983), Brooke, Kendrick and Meeraus (1987)). At the time of its conception and initial development, linear programming (LP) was the standard framework in which market equilibrium models were constructed. Subsequently, solution algorithms and codes for general non-linear programming representations became more common and alternative economic paradigms were developed.<sup>1</sup>

During this time, GAMS was extended from an LP matrix-generator to accommodate general nonlinear functions. GAMS was awarded the Computer Science Technical Section prize by the Operations Research Society of America in 1987. The release of a PC-based version of GAMS in 1988 led to its wide-spread recognition as an effective tool for the development of general purpose large-scale mathematical programming models.

This paper introduces new features of the GAMS modelling language which permit formulating economic equilibrium models as systems of nonlinear equations, complementarity problems or variational inequalities. These extensions accommodate market and game-theoretic equilibrium models which are not easily studied in an optimization context. These formulations are here referred to as “mixed complementarity problems” (MCPs), reflecting the fact that they may incorporate mixtures of equations and inequalities.<sup>2</sup>

The MCP is a natural format for expressing a variety of economic models for both markets and games. Computational evidence suggests that algorithms for solving MCPs are relatively reliable and efficient, particularly for models which are not natural optimization problems.

The development of the GAMS/MCP modelling format was motivated by recent theoretical and practical developments in algorithms for nonlinear complementarity problems and variational inequalities. Some of these algorithms are extensions of classical methods for nonlinear equations. Others exploit the variational structure which is characteristic of many economic equilibrium models (see Dafermos (1983)). The most recent techniques are based on ideas from interior-point algorithms for linear programming (Kojima, Megiddo, Noma and Yoshise (1991)). A survey of developments in the theory and application of these methods is provided by Harker and Pang (1990).

At this time, two large-scale solvers are available through GAMS/MCP. MILES (a Mixed Inequality and nonLinear Equation Solver) employs a modified Newton algorithm which was originally developed for applied general equilibrium modelling (see Rutherford (1987, 1993), and Anstreicher, Lee and Rutherford (1992)). PATH is a recently developed solver based on a promising new path-following procedure (Dirkse and Ferris (1993), Ralph (1994)). Both codes incorporate implicit bounds, mixtures of constrained and unconstrained variables, sparse matrix algebra and dynamic memory allocation. Both solvers employ the basis factorization package from MINOS 5.4, LUSOL (Gill et al.(1991)).

MILES executes a generalized Newton algorithm with a backtracking line search. This method is based on an algorithm investigated by Mathiesen (1985) who proposed a modeling format and sequential method for solving economic equilibrium models. Mathiesen’s method is closely related to algorithms proposed by Robinson (1972), Hogan (1975), Eaves (1978) and Josephy (1979). The algorithm executed by PATH is a “global Newton” method

<sup>1</sup> A chronology of these developments might include Samuelson (1952), Gale (1960), Takayama and Judge (1971), Goreux and Manne (1973) and Adelman and Robinson (1978).

<sup>2</sup> van der Laan and Tallman (1985) refer to these as “generalized complementarity problems”.

in which the backtracking line search is replaced by a “path search”. The algorithms share the same quadratic rate of convergence near a solution, but they may follow different trajectories away from the equilibrium. Because convergence cannot always be guaranteed with either algorithm, it is helpful to have both algorithms available when working with large or difficult problems.

There are two methods for expressing mixed complementarity problems in GAMS. The first is based on “standard” GAMS algebra with extended syntax to signify complementarity links between equations and variables. The second method is a GAMS interface for general equilibrium modeling with MPSGE (a Mathematical Programming System for General Equilibrium). MPSGE is a function and Jacobian evaluation program for a broad class of economic models. Nonlinear equations in an MPSGE model are automatically generated from a tabular description of cost and expenditure functions. The MPSGE interface is well suited for a specific class of functions, whereas GAMS/MCP can be applied to any complementarity problem which can be written with standard GAMS algebra. Any MPSGE model can be written in GAMS/MCP but not all GAMS/MCP models can be cast using MPSGE functions.<sup>3</sup>

The remainder of this paper is organized as follows. Section 2 defines the general class of mathematical programs which can be cast in an MCP format, Section 3 presents MCP formulations for three models which arise in market equilibrium analysis and game theory and computational benchmarks for MILES and PATH. Section 4 describes how local sensitivity analysis can be conducted within GAMS by using a nonlinear optimizer in conjunction with GAMS/MCP. Section 5 summarizes of GAMS/MCP syntax, and section 6 provides a brief conclusion. An appendix (available upon request) contains program listings for models described in the paper.

## 2. Manifestations of the Mixed Complementarity Problem

The “mixed-complementarity problem” (MCP) is defined here as:

$$\text{Given: } F : R^N \rightarrow R^N, \quad \ell, u \in R^N \quad (\text{MCP})$$

$$\text{Find: } z, w, v \in R^N$$

$$\text{s.t. } F(z) - w + v = 0$$

$$\ell \leq z \leq u, \quad w \geq 0, \quad v \geq 0$$

$$w^T(z - \ell) = 0, \quad v^T(u - z) = 0$$

in which  $-\infty \leq \ell \leq u \leq +\infty$ .  $F$  must be continuously differentiable in order to express a model using GAMS/MCP algebra. Sufficient conditions for convergence of a Newton-type algorithm place additional restrictions on  $F$ . Once such condition would be that  $F$  is a  $P$ -function (see Harker and Pang (1990)).<sup>4</sup>

<sup>3</sup> The present paper does not present details of the MPSGE syntax. The interested reader is referred to Rutherford (1992b). MPSGE accommodates “auxiliary constraints” written in GAMS/MCP algebra for economic features which fall outside the standard Arrow-Debreu framework.

<sup>4</sup> In practice, it is quite difficult to determine whether the properties of a particular mapping guarantee convergence. For applied work, it is normal practice to first confirm convergence for a small-scale prototype and then proceed to develop a large scale model.

This MCP format encompasses a number of special cases, including the following:

(i) *a linear system of equations:*

$$\text{Given: } A \in R^{n \times n}, \quad b \in R^n \quad (\text{LSYS})$$

$$\text{Find: } x \in R^n$$

$$\text{s.t. } Ax = b$$

which is represented as an MCP by letting  $\ell = -\infty$ ,  $u = +\infty$ ,  $z = x$ , and  $F(z) = Az - b$ ;

(ii) *a nonlinear system of equations:*

$$\text{Given: } f : R^n \rightarrow R^n, \quad b \in R^n \quad (\text{NLSYS})$$

$$\text{Find: } x \in R^n$$

$$\text{s.t. } f(x) = 0$$

which is represented as an MCP by letting  $\ell = -\infty$ ,  $u = +\infty$ ,  $z = x$ , and  $F(z) = f(z)$ ;

(iii) *a linear complementarity problem:*

$$\text{Given: } M \in R^{n \times n}, \quad q \in R^n \quad (\text{LCP})$$

$$\text{Find: } z \in R^n$$

$$\text{s.t. } q + Mz \geq 0, \quad z \geq 0, \quad z^T(q + Mz) = 0$$

which is represented as an MCP by letting  $\ell = 0$ ,  $u = +\infty$ , and  $F(z) = q + Mz$ ;

(iv) *a nonlinear complementarity problem:*

$$\text{Given: } f : R^n \rightarrow R^n \quad (\text{NCP})$$

$$\text{Find: } z \in R^n$$

$$\text{s.t. } f(z) \geq 0, \quad z \geq 0, \quad z^T f(z) = 0$$

which is represented as an MCP by setting  $\ell = 0$ ,  $u = +\infty$ , and  $F(z) = f(z)$ ;

(v) *a nonlinear program:*

$$\text{Given: } f : R^n \rightarrow R, \quad g : R^n \rightarrow R^m, \quad \hat{\ell}, \hat{u} \in R^n \quad (\text{NLP})$$

$$\text{Find: } x \in R^n \text{ to}$$

$$\max f(x)$$

$$\text{s.t. } g(x) = 0$$

$$\hat{\ell} \leq x \leq \hat{u}$$

which (when  $f()$  is concave and  $g()$  is convex) may be represented as an MCP by setting  $N = n + m$ , and partitioning<sup>5</sup>:

$$z = \begin{pmatrix} x \\ y \end{pmatrix}, \quad \ell = \begin{pmatrix} \hat{\ell} \\ -\infty \end{pmatrix}, \quad u = \begin{pmatrix} \hat{u} \\ +\infty \end{pmatrix}, \quad F(z) = \begin{cases} \nabla f(x) - \nabla g(x)^T y \\ g(x) \end{cases}$$

;

(v) a finite-dimensional system of variational inequalities:

$$\text{Given:} \quad f : R^n \rightarrow R, \quad g : R^n \rightarrow R^m \quad (\text{VIP})$$

$$\text{Find:} \quad x^* \in X \equiv \{\xi \in R^n | g(\xi) \geq 0\}$$

$$\max f(x^*)^T (x - x^*) \geq 0 \quad \forall x \in X$$

which (when  $f()$  is convex and  $g()$  is concave) is represented as an MCP by setting  $N = n + m$ , and partitioning<sup>6</sup>

$$z = \begin{pmatrix} x \\ y \end{pmatrix}, \quad \ell = \begin{pmatrix} -\infty \\ 0 \end{pmatrix}, \quad u = (+\infty), \quad F(z) = \begin{cases} f(x) - \nabla g(x)^T y \\ g(x) \end{cases}$$

.

When a model can be directly expressed as a linear or nonlinear program, it is typically more efficient and reliable to apply a linear or nonlinear programming algorithm, several of which are available through GAMS. MCP is particularly useful for mathematical programs which cannot be processed as optimization problems. Local sensitivity techniques for MCP models may be employed for any of these mathematical programs.

### 3. Economic Models

#### 3.1. SPATIAL EQUILIBRIUM MODELS

As an introduction to the MCP format, consider the Hitchcock-Koopmans transportation problem as described by Dantzig (1963). The data include a set of suppliers  $I$  and a set of markets  $J$ , with supplies  $a_i$ , demands  $b_j$ , transport costs  $c_{ij}$  from supplier  $i$  to market  $j$ . Figure 1 presents GAMS code defining the sets and data parameters used in the various models which are to be subsequently presented. Cast as a planning problem, this linear program seeks a transport schedule which minimizes the cost of supplying all markets.

<sup>5</sup>  $y$  corresponds to the Lagrange multipliers on the constraints in the nonlinear program.

<sup>6</sup> Observe that  $x^*$  in (VIP) solves:

$$\min_{x \in X} f(x^*)^T x$$

Provided that the relevant convexity and regularity conditions are satisfied, the MCP system constitutes necessary and sufficient conditions for minimization, and therefore an MCP solution is a solution to the original system of variational inequalities.

That is:

$$\begin{aligned}
 & \min \sum_{ij} c_{ij} x_{ij} \\
 \text{s.t.} \quad & \sum_j x_{ij} \leq a_i \\
 & \sum_i x_{ij} \geq b_j \\
 & x \geq 0
 \end{aligned}$$

The GAMS code for the linear programming formulation is in Figure 2.<sup>7</sup>

It is well known that this problem can be interpreted as a market equilibrium problem in which the dual multiplier for an equation from the first set of constraints,  $w_i$ , represents the price in supply market  $i$ , and the dual multiplier for an equation in the second set of constraints,  $p_j$ , represents the price in demand market  $j$ .

Cast as a market equilibrium problem, a system of  $IJ$  inequalities and associated complementary slackness conditions replace the minimization operator. “Zero profit conditions” for transportation activities characterize a competitive, constant-returns-to-scale (CRTS) “transportation” industry in which free entry drives excess profits to zero for active trade links and no unprofitable activities are operated. The equations which characterize this equilibrium are:

$$\begin{aligned}
 \sum_j x_{ij} &\leq a_i, & w_i &\geq 0, & w_i (a_i - \sum_j x_{ij}) &= 0 & \forall i \\
 \sum_i x_{ij} &\geq b_j, & p_j &\geq 0, & p_j (b_j - \sum_i x_{ij}) &= 0 & \forall i \\
 w_i + c_{ij} &\geq p_{ij}, & x_{ij} &\geq 0, & x_{ij} (w_i + c_{ij} - p_{ij}) &= 0 & \forall i, j
 \end{aligned}$$

This system of inequalities is a *linear* complementarity problem which may be specified in GAMS/MCP syntax as shown in Figure 3.

There would be no particular reason to formulate a linear program as an MCP. Suppose, however, that demands and supplies were price responsive and all markets perfectly competitive. These assumptions alone would not rule out an optimization approach, but they would require use of a *nonlinear* programming algorithm. For concreteness, let demand and supply functions be isoelastic. The revised equilibrium conditions are:

$$\begin{aligned}
 \sum_j x_{ij} &\leq \alpha_i w_i^{\eta_i}, & w_i &\geq 0, & w_i (\alpha_i w_i^{\eta_i} - \sum_j x_{ij}) &= 0 & \forall i \\
 \sum_i x_{ij} &\geq \beta_j p_j^{-\sigma_j}, & p_j &\geq 0, & p_j (\beta_j p_j^{-\sigma_j} - \sum_i x_{ij}) &= 0 & \forall i \\
 w_i + c_{ij} &\geq p_{ij}, & x_{ij} &\geq 0, & x_{ij} (w_i + c_{ij} - p_{ij}) &= 0 & \forall i, j
 \end{aligned}$$

This is a nonlinear complementarity problem.

When the matrix of cross-price elasticities for a partial equilibrium model is symmetric, there is an associated optimization problem which can be used to compute the equilibrium prices and quantities. (The model is said to be *integrable* - see Takayama and Judge (1971).)

<sup>7</sup> This and subsequent Figures contain GAMS program fragments which illustrate ideas. Full program listings are provided in Appendix B. The MCP transportation model is in the GAMS model library. To copy it to your current directory execute “`gamslib transmcp`”. Readers unfamiliar with the rudiments of the GAMS language may refer to Chapter 2 of the GAMS Users Guide (1987).

## SETS

```

I   canning plants   / SEATTLE, SAN-DIEGO /
J   markets          / NEW-YORK, CHICAGO, TOPEKA / ;

```

## PARAMETERS

```

A(I) capacity of plant i in cases (when prices are unity)
/   SEATTLE      325
   SAN-DIEGO     575 /,
B(J) demand at market j in cases (when prices equal unity)
/   NEW-YORK     325
   CHICAGO       300
   TOPEKA        275 /,

ETA(I) Price elasticity of supply
/   SEATTLE      1.0
   SAN-DIEGO     1.0 /,

SIGMA(J) Price elasticity of demand
/   NEW-YORK     1.5
   CHICAGO       1.2
   TOPEKA        2.0 /;

```

TABLE DIST(I,J) distance in thousands of miles

	NEW-YORK	CHICAGO	TOPEKA
SEATTLE	2.5	1.7	1.8
SAN-DIEGO	2.5	1.8	1.4 ;

SCALAR F freight in dollars per case per thousand miles /90/ ;

PARAMETER C(I,J) transport cost in thousands of dollars per case ;

$C(I,J) = F * DIST(I,J) / 1000 ;$

PARAMETER PBAR(J) Reference price at demand node J (supply price = 1)

```

/   NEW-YORK      1.225
   CHICAGO        1.153
   TOPEKA         1.126 /;

```

Figure 1a. GAMS Set and Data Statements for the Transportation Model

In the present example, cross-price elasticities are zero, and the associated optimization

```

PARAMETER      ALPHA(I)  Supply function share coefficient (MCP),
                BETA(J)   Demand function share coefficient (MCP),
                GAMMA(I)  Supply function share coefficient (NLP),
                KAPPA(J)  Demand function share coefficient (NLP),
                EPSD(J)   Demand function elasticity coefficient,
                EPSY(I)   Supply function elasticity coefficient;

```

```

ALPHA(I) = A(I);
BETA(J)  = B(J) * PBAR(J)**SIGMA(J);
GAMMA(I) = (ETA(I) / (1 + ETA(I))) * (1/ALPHA(I))**(1/ETA(I));
KAPPA(J) = (SIGMA(J) / (SIGMA(J)-1)) * BETA(J)**(1/SIGMA(J));
EPSD(J)  = (SIGMA(J) - 1) / SIGMA(J);
EPSY(I)  = (ETA(I) + 1) / ETA(I);

```

Figure 1b. GAMS Parameter Declarations and Assignments for the Transportation Model

#### VARIABLES

```

X(I,J)          SHIPMENT QUANTITY FROM I TO J
COST             MINIMAND - TOTAL COST OF SHIPMENT

```

POSITIVE VARIABLE X;

#### EQUATIONS

```

SUPPLY(I)       SUPPLY LIMIT
DEMAND(J)       DEMAND CONSTRAINT (FIXED)
OBJDEF          DEFINES COST;

```

```

SUPPLY(I)..     A(I) =G= SUM(J, X(I,J));
DEMAND(J)..     SUM(I, X(I,J)) =G= B(J);
OBJDEF..        COST =E= SUM((I,J), C(I,J) * X(I,J));

```

MODEL MINCOST / SUPPLY, DEMAND, OBJDEF/;

SOLVE MINCOST USING LP MINIMIZING COST;

Figure 2. The Transportation Model Formulated as a Linear Program

problem is:

$$\begin{aligned}
 \max \quad & \sum_i \gamma_i y_i^{e_i^y} + \sum_j \kappa_j d_j^{e_j^d} - \sum_{ij} c_{ij} x_{ij} \\
 \text{s.t.} \quad & \sum_j x_{ij} \leq y_i \\
 & \sum_i x_{ij} \geq d_j \\
 & x_{ij} \geq 0, \quad d_j \geq 0, \quad y_i \geq 0
 \end{aligned}$$

## POSITIVE VARIABLES

W(I) SHADOW PRICE AT SUPPLY NODE I,  
 P(J) SHADOW PRICE AT DEMAND NODE J,  
 X(I,J) SHIPMENT QUANTITIES IN CASES;

## EQUATIONS

SUPPLY(I) SUPPLY LIMIT AT PLANT I,  
 DEMAND(J) FIXED DEMAND AT MARKET J,  
 PROFIT(I,J) ZERO PROFIT CONDITIONS;

PROFIT(I,J).. W(I) + C(I,J) =G= P(J);

SUPPLY(I).. A(I) =G= SUM(J, X(I,J));

DEMAND(J).. SUM(I, X(I,J)) =G= B(J);

MODEL TRNSP / PROFIT.X, SUPPLY.W, DEMAND.P / ;

SOLVE TRNSP USING MCP;

*Figure 3.* The Transportation LP Formulated as an LCP

where:

$$\gamma_i = \frac{\eta_i}{1 + \eta_i} \left( \frac{1}{\alpha_i} \right)^{1/\eta_i}, \quad \kappa_j = \frac{\sigma_j}{1 + \sigma_j} \beta_j^{1/\sigma_j}, \quad \epsilon_i^y = \frac{1 + \eta_i}{\eta_i}, \quad \text{and} \quad \epsilon_j^d = \frac{\sigma_j - 1}{\sigma_j}$$

In economic terms, the market allocation “maximizes the sum of producer and consumer surplus”.

Economic equilibrium models are typically used to assess the consequences of market distortions, typically in the form of taxes, tariffs or other types of government policies. In public finance, equilibrium models are often used to measure the “excess burden” of a given tax structure. When *ad-valorem* taxes are present, the NLP formulation of a market equilibrium problem is not straight-forward. No single optimization problem characterizes the equilibrium because the resulting allocation is inefficient. Such an equilibrium could be computed by solving a *sequence* of nonlinear programs, but in these cases the MCP formulation is certainly more transparent.

As illustration of a model which is difficult to treat in an optimization framework but easy to pose as an MCP, consider applying *ad-valorem* taxes on trade flows in the spatial equilibrium model from above. This causes the supply price to be a non-unitary multiple of the marginal cost of supply, destroying integrability (notice that *specific* taxes do not cause this problem, as they can simply be added to the transport cost coefficients). The GAMS/MCP model is shown in Figure 4.

There are several directions in which the spatial price equilibrium model can be extended. Harker (1986) describes several spatial equilibrium models with imperfect competition. The variational and complementarity versions of these models are implemented in GAMS library models HARKER and HARKERMCP, respectively.

```

PARAMETER          T(I,J)          AD-VALOREM TAX RATE;

POSITIVE VARIABLES
                    W(I)            SHADOW PRICE AT SUPPLY NODE I,
                    P(J)            SHADOW PRICE AT DEMAND NODE J,
                    X(I,J)          SHIPMENT QUANTITIES IN CASES;

EQUATIONS
                    SUPPLY(I)       SUPPLY LIMIT
                    DEMAND(J)       DEMAND CONSTRAINT (FIXED)
                    PROFIT(I,J)     ZERO PROFIT CONDITIONS;

SUPPLY(I)..        ALPHA(I) * W(I)**ETA(I) =G= SUM(J, X(I,J));

DEMAND(J)..        SUM(I, X(I,J)) =G= BETA(J) * P(J)**(-SIGMA(J));

PROFIT(I,J)..      (1 + T(I,J)) * (W(I) + C(I,J)) =G= P(J);

MODEL TRNSP / PROFIT.X, SUPPLY.W, DEMAND.P/ ;

* Avoid function evaluation errors by installing positive bounds:

P.LO(J) = 0.001;  W.LO(I) = 0.001;

* 10% tax on all trade:

T(I,J) = 0.10;

SOLVE TRNSP USING MCP;

```

*Figure 4.* Spatial Price Equilibrium with Ad-Valorem Taxes

### 3.2. AN INTERTEMPORAL GENERAL EQUILIBRIUM MODEL

Many economic problems involves decision-making over time with “dated commodities”. The Ramsey model of forward-looking investment and savings decisions is one such model. In a closed economy setting, Ramsey’s model can be formulated as a nonlinear program. In a multi-regional framework, however, international capital flows are endogenous and an equilibrium cannot be determined by solving a single optimization problem. In the simplest model (see GAMS library models RAMSEY and MR5MCP), a single, homogeneous good is produced throughout the world and this good may be used for either consumption or investment. The market equilibrium can be supported by separate optimization problems for representative agents in each region, taking international prices as given. The region  $r$

planning problem is:

$$\begin{aligned}
 & \max \sum_{t=1}^T \Delta_{rt} \log C_{rt} \\
 \text{s.t.} \quad & C_{rt} = Y_{rt} + M_{rt} - X_{rt} - I_{rt} \\
 & Y_{rt} = \phi_r K_{rt}^{\alpha_r} \bar{L}_{rt}^{1-\alpha_r} \\
 K_{rt} = & \begin{cases} \bar{K}_{r1} & t = 1 \\ \lambda K_{rt-1} + \lambda \frac{n}{2} I_{rt-1} + \frac{n}{2} I_{rt} & t \neq 1 \end{cases} \\
 & I_{rT} = (g_r + \delta_r) K_{rT} \\
 & \sum_{t=1}^T \bar{p}_t (M_{rt} - X_{rt}) = 0
 \end{aligned}$$

The parameters for this planning problem are:

- $p_t$  the present-value international market price of period  $t$  output,
- $\bar{L}_{rt}$ ,  $\bar{K}_{r1}$  the exogenous primary factor endowments (labor and initial capital stock),
- $\Delta_{rt}$  the period  $t$  discount parameter reflecting differences across countries in the rate of time preference and thus in the rate of return to capital,

$$\Delta_{rt} \geq 0, \quad \sum_t \Delta_{rt} = 1 \quad \forall r$$

,

- $n$  the number of years per period,
- $\lambda_r$  the one-period survival share for physical capital subject to “evaporative decay”,
- $g_r + \delta_r$  the sum of growth and depreciation rates in the terminal period, and
- $\phi, \alpha_r$  parameters of region  $r$ 's Cobb-Douglas production function.

The planning problem takes international prices,  $p_t$ , as given and it produces a vector of net exports  $\xi_{rt}(p) = X_{rt}(p) - M_{rt}(p)$ .

An international equilibrium is a vector of present value prices for which  $\sum_r p_t \xi_{rt}(p) = 0 \quad \forall r$ . This interpretation suggests an algorithm in which  $p$  is iteratively adjusted in order to clear international markets. This and more sophisticated adjustment procedures can be applied to solve multi-consumer general equilibrium models (see Rutherford (1992a)), but in most cases the MCP formulation is both more easily implemented and more efficient.

The complementarity formulation for this model is as follows:

- 1.r The present value of expenditure equals factor income:

$$E_r = \sum_t w_{rt} \bar{L}_{rt} + p_{r1}^K \bar{K}_{r1}$$

- 2.t Aggregate world output equals consumption plus investment:

$$\sum_r Y_{rt} \geq \sum_r \frac{\text{Delta}_{rt} E_r}{p_t} + I_{rt}$$

- 3.rt The present value cost of output is a Cobb-Douglas cost function based on the regional wage and rental rate:

$$c_{rt} = \frac{1}{\phi_r} \left( \frac{r_{rt}^K}{\alpha_r} \right)^{\alpha_r} \left( \frac{w_{rt}}{1 - \alpha_r} \right)^{1 - \alpha_r}$$

- 4.rt Perfect competition assures no excess profits:

$$c_{rt} \geq p_t$$

- 5.rt The supply of capital equals demand for capital use:

$$K_{rt} \geq Y_{rt} \frac{(1 - \alpha_r) c_{rt}}{r_{rt}^K}$$

- 6.rt The exogenous labor supply equals demand:

$$\bar{L}_{rt} \geq Y_{rt} \frac{\alpha_r c_{rt}}{w_{rt}}$$

- 7.rt The capital stock equals depreciated stock from the previous period together with new investments from the previous and current period:

$$K_{rt} = \begin{cases} \bar{K}_{r1} & t = 1 \\ \lambda K_{rt-1} + \lambda \frac{n}{2} I_{rt-1} + \frac{n}{2} I_{rt} & t \neq 1 \end{cases}$$

- 8.rt Unrestricted international capital flows assure that the current consumption price is no less than the return on current investment:

$$p_t \geq \begin{cases} \frac{n}{2} p_{rt}^K + \frac{n}{2} \lambda p_{rt+1}^K & t < T \\ \frac{n}{2} p_{rt}^K + p_r^T & t \neq 1 \end{cases}$$

- 9.rt The cost of capital is no less than the current rental price plus the salvage value from sales in the subsequent period:

$$p_t^K \geq \begin{cases} r_{rt}^K + \lambda p_{rt+1}^K & t < T \\ r_{rt}^K - p_r^T (g_r + \delta_r) & t \neq 1 \end{cases}$$

- 10.r Terminal investment must be sufficient to assure steady-state growth in the post-terminal period:

$$I_{rT} = (g_{rT} + \delta_r) K_{rT}$$

```

$TITLE  MULTI-REGION RAMSEY MODEL

SET
      R          REGIONS /USA, OOECD, USSR, CHINA, ROW/,
      T          TIME PERIODS /T1*T11/,
      TINIT(T)   INITIAL PERIOD   /T1/
      TTERM(T)   TERMINAL PERIOD  /T11/;

SCALAR
      N          NUMBER OF YEARS PER PERIOD /10/

PARAMETERS
L(T,R)  LABOR STOCK IN EFFICIENCY UNITS
BETA(T,R)  INTERTEMPORAL UTILITY PARAMETER
KINIT(R)  INITIAL PERIOD CAPITAL STOCK
PHI(R)    PRODUCTION FUNCTION SCALE PARAMETER
ALPHA(R)  CAPITAL VALUE SHARE
DELTA(R)  ANNUAL CAPITAL DEPRECIATION RATE
GAMMA(R)  TERMINAL PERIOD GROWTH RATE
LAMBDA(R) ONE PERIOD CAPITAL SURVIVAL SHARE;

```

Figure 5. Set and Parameter Declarations for the Multi-Region Ramsey Model

Figure 5 displays set and parameter definitions for this model (input data tables are suppressed). The GAMS/MCP equations and solution statements are presented in Figure 6. These equations correspond more or less exactly to the formulation given above.

There are two successive solution statements shown in this Figure, illustrating an solution scheme which is advisable for any general equilibrium model in which a good estimate of equilibrium values is not easily deduced. The first MCP solves the “partial equilibrium relaxation” of the model by fixing relative income levels. (See Rutherford (1992a).) After having obtained this approximate equilibrium, the model with uncompensated demand functions is solved - using the partial equilibrium values as a starting point and holding the income for one agent fixed to normalize prices.

### 3.3. AN MCP FORMULATION FOR N-PLAYER NON-COOPERATIVE GAMES

The Lemke-Howson (1964) algorithm finds equilibria for two-player non-cooperative games with finite action spaces in pure or mixed strategies. An N-player extension of bimatrix games is conveniently expressed with the following notation:

- $j$  denotes players  $j \in \{1 \dots N\}$
- $i$  denotes pure strategies  $i \in \{1 \dots M\}$  (the same number of strategies are assumed available to all players to simplify notation)
- $s$  denotes a “pure strategy profile” in which one pure strategy is assigned to each player,  $s \in \{1 \dots M^N\}$
- $i(s, j)$  denotes the pure strategy assigned to player  $j$  in strategy profile  $s$ .
- $a_j s$  denotes the payoff to player  $j$  which arises from strategy profile  $s$ .

## POSITIVE VARIABLES

E(R) FACTOR INCOME  
 P(T) PRESENT VALUE PRICE OF OUTPUT  
 Y(R,T) NEW VINTAGE PRODUCTION  
 I(R,T) INVESTMENT  
 K(R,T) CAPITAL STOCK  
 C(R,T) UNIT COST OF PRODUCTION  
 W(R,T) WAGE RATE  
 RK(R,T) RENTAL PRICE ON CAPITAL  
 PK(R,T) PRESENT VALUE COST OF CAPITAL  
 PT(R) TERMINAL CAPITAL PRICE;

## EQUATIONS

INCOME(R) PRESENT VALUE OF FACTOR INCOME,  
 SUPPLY(T) GLOBAL SUPPLY-DEMAND,  
 COST(R,T) UNIT COST OF PRODUCTION  
 PROFIT(R,T) UNIT PROFIT,  
 K\_USE(R,T) CAPITAL RENTAL MARKET  
 LABOR(R,T) LABOR MARKET  
 K\_SUPPLY(R,T) CAPITAL STOCK,  
 PROFIT\_I(R,T) INVESTMENT PROFIT,  
 COST\_K(R,T) CAPITAL VALUE CONDITION,  
 TERM\_I(R) TERMINAL INVESTMENT;

INCOME(R)..  $E(R) = E = \text{SUM}(T, W(R,T) * L(T,R)) + \text{KINIT}(R) * \text{PK}(R, "T1");$

SUPPLY(T)..  $\text{SUM}(R, Y(R,T)) = G = \text{SUM}(R, \text{BETA}(T,R) * E(R) / P(T) + I(R,T));$

COST(R,T)..  $C(R,T) = E = (1 / \text{PHI}(R)) * (\text{RK}(R,T) / \text{ALPHA}(R)) ** \text{ALPHA}(R) * (\text{W}(R,T) / (1 - \text{ALPHA}(R))) ** (1 - \text{ALPHA}(R));$

PROFIT(R,T)..  $C(R,T) = G = P(T);$

K\_USE(R,T)..  $K(R,T) = G = Y(R,T) * \text{ALPHA}(R) * C(R,T) / \text{RK}(R,T);$

LABOR(R,T)..  $L(T,R) = G = Y(R,T) * (1 - \text{ALPHA}(R)) * C(R,T) / \text{W}(R,T);$

K\_SUPPLY(R,T)..  $(N/2) * I(R,T) + (N/2) * \text{LAMBDA}(R) * I(R, T-1) + \text{LAMBDA}(R) * K(R, T-1) + \text{KINIT}(R) * \text{TINIT}(T) = G = K(R,T);$

PROFIT\_I(R,T)..  $P(T) = G = (N/2) * \text{PK}(R,T) + (N/2) * \text{LAMBDA}(R) * \text{PK}(R, T+1) + \text{PT}(R) * \text{TTERM}(T);$

COST\_K(R,T)..  $\text{PK}(R,T) = G = \text{RK}(R,T) + \text{LAMBDA}(R) * \text{PK}(R, T+1) - (\text{PT}(R) * (\text{GAMMA}(R) + \text{DELTA}(R))) * \text{TTERM}(T);$

TERM\_I(R)..  $\text{SUM}(\text{TTERM}, I(R, \text{TTERM})) = G = \text{SUM}(\text{TTERM}, K(R, \text{TTERM}) * (\text{GAMMA}(R) + \text{DELTA}(R)));$

MODEL RAMSEY\_MCP / INCOME.E, SUPPLY.P, COST.C, PROFIT.Y, K\_USE.RK, PROFIT\_I.I, COST\_K.K, TERM\_I.PT/;

Figure 6a. MCP Formulation of the Multi-Regional Ramsey Model

Y.L(R,T)=1; E.L(R)=1; I.L(R,T)=1; K.L(R,T)=1; C.L(R,T)=1; P.L(T)=1; W.L(R,T)=1;  
 RK.L(R,T)=1; PK.L(R,T)=1; PT.L(R)=1; RK.LO(R,T)=0.0001; W.LO(R,T)=0.0001; P.LO(T)=0.0001;

\* FIRST SOLVE THE FIXED-INCOME RELAXATION:

E.FX(R)=SUM(T, L(T,R));

SOLVE RAMSEY\_MCP USING MCP;

\* THEN FIX ONLY ONE INCOME LEVEL TO NORMALIZE THE PRICE SYSTEM:

E.LO(RG) = 0; E.UP(RG) = +INF;  
 E.FX(RG)\$ (ORD(RG) EQ 1) = INCOME(RG);

SOLVE MRG USING MCP;

*Figure 6b.* Assign Initial Values and Solve the Multi-Region Ramsey Model

Each player  $j$  chooses a vector of probabilities,  $p_{ij}$ , representing the probability of playing pure strategy  $i$  in the equilibrium. By definition:

$$p_{ij} \geq 0 \quad \forall i, j, \quad \text{and} \quad \sum_i p_{ij} = 1 \quad \forall j$$

Let  $q_{js}$  denote the probability that player  $j$  is confronted by pure strategy profile  $s$  in the equilibrium. Formally:

$$q_{js} = \prod_{j' \neq j} p_{i(j',s),j'}$$

We assume that player  $j$  chooses a strategy profile to maximize the expected payoff, taking actions of the other players as given:

$$\begin{aligned} \max_{p_{ij}} \pi_j &= \sum_i p_{ij} \left( \sum_{s|i(s,j)=i} a_{js} q_{js} \right) \\ \text{s.t.} \quad \sum_i p_{ij} &= 1, \quad p_{ij} \geq 0 \end{aligned}$$

A workable MCP formulation for the equilibrium problem needs to incorporate individual simplex constraints in a way which does not render the mapping singular at the equilibrium. In the formulation presented here, an additional variable  $y_j$  is associated with the simplex constraint for player  $j$ . It is easy to show that in any solution to this MCP,  $y_j$  must have a value of unity. The MCP formulation is:

$$\begin{aligned} \pi_j &= \sum_s a_{js} \left[ \prod_{j'} \left( \frac{p_{i(s,j'),j'}}{y_{j'}} \right) \right] \quad \forall j \\ \pi_j &\geq \sum_{s|i(s,j)=i} a_{js} \left( \prod_{j' \neq j} \frac{p_{i(s,j'),j'}}{y_{j'}} \right), \quad p_{ij} \geq 0, \\ p_{ij} \left\{ \pi_j - \sum_{s|i(s,j)=i} a_{js} \left( \prod_{j' \neq j} \frac{p_{i(s,j'),j'}}{y_{j'}} \right) \right\} &= 0, \quad \forall i, j \\ \sum_i p_{ij} &\geq 1, \quad y_j \geq 0, \quad y_j (1 - \sum_i p_{ij}) = 1 \quad \forall j \end{aligned}$$

```

SET      J          PLAYERS /J1*J3/,
        I          ACTIONS /I1*I2/,
        SS         STRATEGY PROFILES /S1*S1000/,
        S(SS)      ACTIVE STRATEGY PROFILES,
        LINK(SS,I,J) CONNECTION BETWEEN ACTIONS AND PROFILES;

```

```

ABORT$(CARD(SS) LT CARD(I1)**3) " SET SS IS NOT LARGE ENOUGH.";

```

```

PARAMETER

```

```

        PAYOFF(SS,J)    PAYOFF DATA IN STRATEGY-PROFILE FORMAT;

```

```

*      READ THE DATA:

```

```

TABLE LOSS(J,I,I,I)  PAYOFF DATA IN DENSE FORM

```

	I1.I1	I1.I2	I2.I1	I2.I2
J1.I1	1	2	8	5
J1.I2	8	8	2	2
J2.I1	4	2	2	1
J2.I2	2	6	1	1
J3.I1	4	1	4	2
J3.I2	8	8	2	1;

```

*      LET I1, I2 AND I3 ALL STAND FOR SET I:

```

```

ALIAS (I,I1),(I,I2),(I,I3);

```

```

*      ASSIGN PLAYER ACTIONS TO STRATEGY PROFILES USING
*      DYNAMIC SET S().

```

```

*      INITIALIZE SET S WITH A SINGLE ENTRY, THE FIRST ELEMENT OF SS:

```

```

S("S1") = YES;

```

```

*      ONE ELEMENT OF SET S IS GENERATED FOR EACH COMBINATION OF PURE STRATEGIES:

```

```

LOOP((I1,I2,I3),

```

```

*      SET S CONTAINS A SINGLE ELEMENT FOR THE FOLLOWING STATEMENT:

```

```

        LINK(S,I1,"J1") = YES;
        LINK(S,I2,"J2") = YES;
        LINK(S,I3,"J3") = YES;
        PAYOFF(S,J) = -LOSS(J,I1,I2,I3);

```

```

*      THE NEXT STATEMENTS MOVE S THE SUBSEQUENT ELEMENT OF SS:

```

```

        S(SS)$S(SS-1) = YES;
        S(SS)$S(SS+1) = NO;

```

```

);

```

```

*      THE NEXT STATEMENTS DEFINE THE SET OF ACTION PROFILES AS
*      THOSE ELEMENTS OF SS FOR WHICH PAYOFFS HAVE BEEN ASSIGNED:

```

```

S(SS) = NO;    S(SS) = YES$SUM(J, ABS(PAYOFF(SS,J)));

```

```

POSITIVE
VARIABLE      P(I,J)          PROBABILITY OF STRATEGY I BY PLAYER J,
               Y(J)          SIMPLEX VARIABLE (=1 IN EQUILIBRIUM)
               PI(J)         EXPECTED RETURN TO PLAYER J;

EQUATION      PIDEF(J)       EXPECTED PAYOFF DEFINITION,
               SIMPLEX(J)    SIMPLEX CONSTRAINT
               EQUIL(I,J)    EXCESS RETURN FROM PURE STRATEGY;

*           EXPECTED PAYOFF:

PIDEF(J)..

    PI(J) =E= SUM(S, PAYOFF(S,J) * PROD(JJ, SUM(I$LINK(S,I,JJ), P(I,JJ)/Y(JJ))));

*           STRATEGY I BY PLAYER J YIELDS NO MORE THAN EXPECTED RETURN:

EQUIL(I,J)..

    PI(J) =G= SUM(S$LINK(S,I,J), PAYOFF(S,J) *

                PROD(JJ$(ORD(JJ) NE ORD(J)), SUM(II$LINK(S,II,JJ), P(II,JJ)/Y(JJ))));

*           PROBABILITIES SUM TO 1:

SIMPLEX(J)..    SUM(I, P(I,J)) =G= 1;

```

Figure 8. GAMS/MCP Code for the General N-Player M-Strategy Nash Game

The GAMS implementation of this model is displayed in Figures 7 and 8. Figure 7 sets up the data structures using dynamic sets to produce the mapping  $i(s,j)$ .

This is the "trickiest" part of the programming because it relies on the use of dynamic ordered sets. Once the data structure  $LINK(S,I,J)$  has been installed, it is straightforward to formulate the equilibrium conditions shown in Figure 8.

### 3.4. COMPUTATIONAL BENCHMARKS

The two solvers available for solving mixed complementarity problems in GAMS are both Newton-type algorithms, hence they share the same local convergence properties. The solvers differ in the adjustment process far from the equilibria.

Some statistics comparing their performance on the the multi-regional Ramsey model and the N-player Nash equilibrium models are presented in Figure 9.<sup>8</sup>

<sup>8</sup> In this Figure "convergence rate%" is 100 times the number of convergent runs divided by the total number of starting points evaluated, "equilibria" is the number of distinct equilibria identified by either

For these tests, both solvers are using default tolerances, iteration and resource limits. The Ramsey model results confirm that both algorithms are capable of processing very large-dimensional models when they are sparse. PATH is generally more efficient for these test problems.<sup>9</sup>

Both solvers find the non-cooperative games difficult. The first three problems are from van der Laan et al. (1987).<sup>10</sup> The fourth problem is based on randomly generated payoffs. MILES is more robust for these problems, but this may be due to lower effective iteration limits for PATH. The usefulness of having two solvers is illustrated with the non-cooperative games, where use of both solvers considerably increases convergence rates (see values for PATH & MILES).

The comparison between alternative solvers available within GAMS/MCP is not as revealing as a comparison between these solvers and other programming tools. The use of analytical gradients and sparse data structures provide a significant advantage for GAMS/MCP relative to other approaches. Sparsity is not unusual. For many economic equilibrium problems, the most natural set of variables and equations is one in which many equations involve only a small number of variables. This causes the underlying Jacobian matrix to have a large number of zeros - i.e., it is sparse.

In order to illustrate the advantages of analytic gradients and sparse data structures, GAMS/MCP was compared with the widely advertised econometric software package GAUSS which includes a dense-matrix nonlinear system solver in which Jacobians are generated by numerical differencing. The GAUSS NLSYS solver otherwise implements a standard Newton algorithm with the same number of iterations. The test problem was a static, multi-sectoral general equilibrium model GEMMCP from the GAMS model library. This problem is a system of 262 nonlinear equations. In the test, four related equilibria were computed. The resulting clock times were 32 seconds for GAMS/MILES and 14 minutes for GAUSS.

#### 4. Local Sensitivity Analysis

Suppose that GAMS/MCP has been used to solve a nonlinear system of the form:

$$F(z; t) = 0$$

where  $z$  is an  $N$ -vector of decision variables,  $t$  is an  $M$ -vector of parameters, and  $F$  is a function mapping  $R^{N+M}$  to  $R^N$ .

Given a solution  $z^*$ , it is often helpful to know the local dependence of  $z^*$  on  $t$ .

Let  $J_z$  denote the Jacobian of  $F$  with respect to  $z$  (i.e.  $\nabla_z F$ ), and let  $J_t$  denote the Jacobian of  $F$  with respect to  $t$  (i.e.  $\nabla_t F$ ). It is easy to show that:

$$dz = -J_z^{-1} J_t dt \equiv S dt$$

algorithm during these tests, and "density" is 100 times the number of Jacobian non-zeros divided by the square of the number of variables.

<sup>9</sup> The subroutines for crashing a complementary basis described in Anstreicher et al. (1992) are not yet included in this version of MILES. Consequently, MILES solution time for the Ramsey test problems is spent largely on the first linear subproblem which is computed from a default Lemke starting point because the initial basis is typically singular.

<sup>10</sup> In their computational results, van der Laan et al. (1987) make no mention of the multiplicity of equilibria. Their computational results for a specialized fixed-point procedure are cited in terms of the cumulative number of function evaluations and are therefore difficult to compare.

## Statistics for Multi-region Ramsey Model (starting point: all values = 1)

## Size

Regions	5	5	5	12	12	12
Time periods	10	15	20	10	15	20
Variables	369	549	729	873	1298	1723
Non-zeros	1330	1985	2640	3178	4743	6308
Density	1.0%	0.7%	0.5%	0.4%	0.3%	0.2%

## Pivots

PATH	388	578	775	886	1364	1834
MILES	380	573	773	888	1431	***

## Solution time (seconds, 33MHz 80486)

PATH	31.85	55.91	83.43	97.33	181.59	282.65
MILES	35.93	72.39	133.69	175.65	585.62	***

## Statistics for N-Player Game (25 randomly selected starting points)

## Size

Players	3	3	4	4
Actions	2	3	2	4
Variables	12	15	16	24
Non-zeros	78	129	140	356
Density	54.2%	57.3%	54.7%	61.8%
Equilibria	1	6	3	11

## Convergence Rate (%)

PATH	80%	60%	60%	12%
MILES	64%	80%	64%	48%
PATH&MILES	80%	92%	76%	56%

## Average Solution Time (seconds, 33 MHz 80486)

PATH	2.8	6.6	6.7	27.5
MILES	2.8	5.0	6.8	102.6

Figure 9. Solver Performance for Ramsey and N-Player Problems

where  $J_z^{-1}$  is the inverse of  $J_z$ . This equation defines the “local sensitivity matrix”,  $S = -J_z^{-1}J_t$ . If the system of equations is specified so that  $J_z$  has full rank at the solution, then the gradients fall out as the solution of a linear system of equations. Setting up the linear system with the Jacobian matrices is easy in theory but tedious in practice. Fortunately, there is an easier approach by which GAMS generates the derivatives automatically.

Consider the following nonlinear program:

$$\begin{aligned} & \max_{z,t} z_i \\ \text{s.t.} \quad & F(z; t) = 0 \\ & t = \bar{t} \end{aligned}$$

The implicit function theorem (see Varian (1992)) implies that the  $i$ th row of  $S$  corresponds to the dual multipliers associated with the bounds on  $t$ . To compute the entire sensitivity matrix therefore requires the solution of  $N$  separate sets of nonlinear programs, all of which may be initiated at the equilibrium point.

A minor variation on this technique applies to complementarity problems involving mixtures of equations and inequalities. In the MCP case, the nonlinear system passed to the nonlinear program only include those equations which are *binding* at the solution (i.e., any equations which are slack in the MCP solution must be omitted from the NLP using GAMS exception (“\$”) operators). An illustration of this technique is provided in the Appendix.

## 5. GAMS/MCP Syntax

In a GAMS/MCP program, the central unknowns,  $z$ , are declared as VARIABLES, the vectors  $u$  and are specified as upper and lower bounds. The function  $F()$  is written using GAMS matrix algebra in equation statements. All aspects of the GAMS/MCP language are identical to GAMS/NLP except that in an MCP problem no objective function is specified, and bounded variables must be mapped to complementary inequalities.

In the canonical MCP structure, the equations  $w^T(z - \ell) = 0$  and  $v^T(u - z) = 0$  imply a particular association of variables with equations. Quite naturally,  $z_i$  is complementary with  $F_i(z)$ . In an empirical model, there are typically several classes of variables and equations, and the declaration sequence is arbitrary, so the rule  $z_i F_i = 0$  is impossible to interpret.<sup>11</sup>

To fix the complementarity structure, a GAMS/MCP program explicitly associates variables with equations in the MODEL statement. Unlike an NLP in which only equation names appear on the model statement, in an MCP program both equation and variable names may be listed. The syntax is:

```
MODEL NAME /EQU1.VAR1, EQU2.VAR2, .../;
```

In this list, EQU\_i and VAR\_i must be defined over the same domains.

Complementarity associations are only required for variables which are subject to upper or lower bounds. Otherwise, GAMS/MCP only requires that the number of unrestricted

---

<sup>11</sup> One way to make this association would have been to use the same name for both equations and variables. This syntax was rejected because it would not be “upward compatible” with NLP models. The present structure permits “sharing” a system of equations in both complementarity and optimization models within the same program. Furthermore, the present structure makes it quite simple to convert existing GAMS/NLP models into the complementarity structure.

variables equal the number of unassociated equations in the model. It is a simple matter to “mix” equations with inequalities in this format. The specification of linear and nonlinear systems of equations with no bounds and complementarity conditions requires no variable names on the MODEL list.

Joint consistency of inequalities and complements assignments is important when finite bounds are imposed on  $z$ . In the MCP structure, slack variables  $w$  and  $v$  are implicitly represented in a GAMS/MCP model through the equation statement. The fact that the lower bound slack variable ( $w$ ) appears with a coefficient of  $-1$  and the upper bound slack variable ( $v$ ) appears with a coefficient of  $+1$  means that the “orientation” of the inequality is significant. In GAMS/MCP, the equations “ $F(z) \geq 0$ ” and “ $-F(z) \leq 0$ ” are *not* equivalent. In the first case, the inequality is assumed to be complementary with the lower bound slack of the associated variable while in the second case the inequality is taken to be complementary with the upper bound slack.

Compiler error conditions can result from improperly specified bounds and associated equation types. If an equation is written using =G= then the associated lower bound must be finite. Likewise, if a constraint is written as =L= the associated upper bound must be finite. Table A.1 outlines the rules which apply when variables are associated with different types of equations.

## 6. References

- ADELMAN, I. and S. ROBINSON (1978) *Income Distribution Policy in Developing Countries: A Case Study of Korea*, Oxford University Press, London.
- ANSTREICHER, K, J. LEE and T.F. RUTHERFORD (1992) “Crashing a Maximum Weight Complementary Basis”, *Mathematical Programming* 54(3), 281-290.
- BROOKE, T., D. KENDRICK and A. MEERAUS (1987) *GAMS: A User's Guide*, Scientific Press, South San Francisco.
- DAFERMOS, S. (1983) “An Iterative Scheme for Variational Inequalities”, *Mathematical Programming* 26, 40-47.
- DANTZIG, G. (1963) *Linear Programming and Extensions* Princeton University Press.
- DIRKSE, S. and M. FERRIS (1993) “The PATH Solver: A Non-Monotone Stabilization Scheme for Mixed Complementarity Problems”, Tech Report 1179, CS Dept, UW-Madison, September.
- DUFFIE, D. and W. SHAFER (1985) “Equilibrium in Incomplete Markets: I. A Basic Model of Generic Existence”, *Journal of Mathematical Economics* 14, 285-300.
- EAVES, B.C. (1978) “A locally quadratically convergent algorithm for computing stationary points,” Tech. Rep., Department of Operations Research, Stanford University, Stanford, CA.
- GALE, D. (1960) *The Theory of Linear Economic Models*, University of Chicago Press.
- GILL, P., W. MURRAY, M.A. SAUNDERS and M.H. WRIGHT (1991) “Maintaining LU factors of a general sparse matrix, *Linear Algebra and its Applications* 88/89, 239-270.
- GOREUX, L.M. and A.S. MANNE (1973) *Multi-level Planning: Case Studies in Mexico*. Amsterdam: North-Holland.
- HARKER, P. (1986) “Alternative Models of Spatial Competition”, *Operations Research*, 34(3), 410-425.

## Associated Equation Type

Variable	=G=	=L=	=E=
Lower	OK	Error	OK/warning
Upper	Error	OK	OK/warning
Both	OK/warning	OK/warning	OK/warning
Free	OK/message	OK/message	OK
Fixed	OK	OK	OK

Key:

\* ‘‘Error’’ means that the correspondence is inconsistent and the problem will not be processed.

\* ‘‘OK/warning’’ means that the association is acceptable but the constraint as written may not be satisfied in the solution. We accept the syntax and issue a warning on the list file if the constraint is not satisfied at the equilibrium. This warning is generated if the constraint is violated either by non-zero upper and lower bound slack variables or through the effects of the multipliers from the variational inequality side constraints.

\* ‘‘OK/message’’ means that an inequality constraint has been associated with an unrestricted variables. The variable will always be in the basis, so the constraint will never be slack and it is therefore treated as an equality. A message to this effect will be generated on the listing file.

Figure 10. Rules for Associating Complementary Variables and Equations

HARKER, P. and J.S. PANG (1990) ‘‘Finite-Dimensional Variational Inequality and Nonlinear Complementarity Problems: A Survey of Theory, Algorithms and Applications’’, *Mathematical Programming B*, 38(2), 161-190.

HOGAN, W.W. (1975) ‘‘Energy policy models for project independence,’’ *Computation and Operations Research* 2, 251-271.

JOSEPHY, N.H. (1979) ‘‘Newton’s method for generalized equations’’, Technical Summary Report #1965, Mathematical Research Center, University of Wisconsin - Madison.

KOJIMA, M., N. MEGIDDO, T. NOMA and A.YOSHISE (1991) ‘‘A unified approach to interior point algorithms for linear complementarity problems’’, *Lecture Notes in Computer Science* 538, Springer-Verlag.

LAAN, G. van der, and A.J.J. TALMAN (1985) ‘‘An algorithm for the linear complementarity problem with upper and lower bounds’’, *Journal of Optimization Theory and Applications* 62(1), 151-163.

LAAN, G. van der, and A.J.J. TALMAN and L. van der HEYDEN (1987) ‘‘Simplicial variable dimension algorithms for solving the nonlinear complementarity problem on a

product of unit simplices using a general labelling”, *Mathematics of Operations Research* 12(3), 337-397.

LEMKE, C.E. and J.T. HOWSON (1964) “Equilibrium points of bimatrix games,” *SIAM Review* 12, 413-23.

MATHIESEN, L. (1985) “Computation of Economic Equilibrium by a Sequence of Linear Complementarity Problems”, *Mathematical Programming Study* 23, North-Holland.

MEERAUS, A. (1983) “An algebraic approach to modeling,” *Journal of Economic Dynamics and Control*, 5(1), February.

RALPH, D. (1994) “Global Convergence of Damped Newton’s Method for Nonsmooth Equations, via the Path Search”, *Mathematics of Operations Research*, 19(2), 352-371.

ROBINSON, S. (1972) “A quadratically-convergent algorithm for general nonlinear programming problems,” *Mathematical Programming Study* 3, 145-156.

RUTHERFORD, T. (1987) “Applied general equilibrium modeling”, Ph.D. Dissertation, Department of Operations Research, Stanford University.

RUTHERFORD, T. (1992a) “Sequential joint maximization”, Working Paper 92-8, Department of Economics, University of Colorado.

RUTHERFORD, T. (1992b) “Applied general equilibrium modeling using MPS/GE as a GAMS subsystem”, Working Paper 92-15, Department of Economics, University of Colorado.

RUTHERFORD, T. (1993) “MILES: a Mixed Inequality and nonLinear Equation Solver”, Working Paper, Department of Economics, University of Colorado.

SAMUELSON, P. (1952) “Spatial price equilibrium and linear programming,” *The American Economic Review*, 42, 283-303.

TAKAYAMA, T. and G.G. JUDGE (1971) *Spatial and Temporal Price and Allocation Models*, North Holland, Amsterdam.

VARIAN, H. (1992) *Microeconomic Analysis (3rd edition)*, Norton, New York.



# Sequential Joint Maximization

## *Relating Optimization and General Equilibrium*

**Abstract.** This paper describes a new method by which a competitive market economy may be represented as the optimal solution to a planning problem. The resulting sequential joint maximization (SJM) algorithm solves a sequence of “partial equilibrium relaxations” of the underlying general equilibrium model. The partial equilibrium submodels can be solved as nonlinear complementarity problems or as constrained nonlinear optimization problems in either a primal or dual form. This paper introduces the three SJM algorithms, evaluates performance and examines convergence theory. Computational tests demonstrate that SJM is not always as efficient as complementarity methods. A counterexample is presented which demonstrates that local convergence of SJM cannot be guaranteed. Although SJM may have neither the theoretical pedigree of a fixed-point algorithm nor the local convergence rate of a Newton algorithm, the usefulness of the SJM algorithm is demonstrated by the range of real models for which the procedure has been successfully applied. SJM benefits from the availability of robust, large-scale nonlinear programming codes. Consequently, for large scale equilibrium models with inequalities SJM may not be the “best” algorithm in theory, but it works extremely well in practice.



## 1. Introduction

There is a close connection between the allocation of a competitive market economy and the optimal solution to a representative agent's planning problem.<sup>1</sup> This paper describes a new method by which this relationship can be exploited in computing equilibrium prices and quantities. This optimization-based algorithm can be applied to find equilibria for models in which consumers have heterogeneous preferences and where production sets may involve point-to-set mappings. Large-scale models may be solved through this technique. Extensions of the algorithm may be applied to models with tax distortions, price rigidities, monopolistic competition and arbitrary preferences. The present paper, however, only considers applications in which producers are perfectly competitive and consumer preferences are homothetic.

The algorithm is a Negishi or "joint maximization" procedure based on a sequence of convex nonlinear programming problems. Typically this sequence will converge to the equilibrium prices and quantities of a competitive market economy. This paper introduces and interprets the algorithm, provides some computational evidence and develops local convergence theory for a simple example.

The idea behind the SJM algorithm is found in the literature concerning conditions under which the demand function describing a set of heterogeneous consumers can be replaced by the demand function for a single agent. In many settings, this is known as the problem of exact aggregation. In neoclassical trade theory this relates to the "existence of community indifference curves". This literature investigates the conditions under which a country's offer curve may be represented as though it arose from a single optimizing agent. In trade theory, at least two such conditions have been identified. The first, due to Samuelson (1956), is that all consumers have identical, homothetic preferences. A second, less widely cited result is due to Eisenberg (1961) and Chipman (1974). This condition places restrictions on endowment vectors in addition to preferences. Eisenberg's result, as summarized by Chipman (1974), is:

"if each of  $m$  individuals has a fixed money income, and if they all have homogeneous utility functions (not necessarily identical, it must be emphasized), then their aggregate demand function is integrable, that is, it may be thought of as resulting from the maximization of some fictitious aggregate utility function, subject to total expenditure being equal to total income."

In the algorithm described in this paper, Eisenberg's aggregate demand function is embedded in an iterative procedure which accounts for changes in the relative incomes of agents in the equilibrium system.

The aggregation approach developed by Eisenberg and subsequently interpreted in an economic context by Chipman is presented in a "primal" form. This approach begins with an explicit representation of consumer preferences in which the utility of an individual is specified in terms of quantities consumed of different goods. An analogous result is obtained using preferences characterized in a "dual" form (using the indirect utility function which

---

<sup>1</sup> The use of an optimization problem to characterize equilibrium allocations in a general equilibrium framework is due to Negishi (1960). His original paper was primarily concerned with optimization as a means of proving existence. Dixon (1975) developed the theory and computational effectiveness of "joint maximization algorithms" for multi-country trade models. A survey of early applications of Negishi's theory is provided by Carey (1977). See also Dantzig, Eaves and Gale (1979). Extensions of joint maximization methods to account for tariffs and labor market imperfections are provided by Ginsburgh and Van der Heyden (1988) and Kehoe, Levine and Romer (1992).

maps prices and income into utility) which provides an interesting connection between the duality theory of linear programming and the duality of demand function theory. This formulation also renders the computational procedure applicable to models based on econometrically-estimated functions for which primal forms may not exist.

There is an interesting correspondence between the sequential joint maximization (SJM) algorithm and both Mathiesen's (1985) sequential linear complementarity (SLCP) algorithm and Goldsman and Harker's (1990) variational inequality (VI) algorithm for general equilibrium models. SJM can be regarded as a robust form of SLCP; one in which income and price adjustments are effectively decoupled. The SJM algorithm computes a sequence of "partial equilibrium relaxations" of the underlying general equilibrium model. Unlike SLCP (see Mathiesen, 1987), in SJM the choice of numeraire does not affect convergence.

In SJM, subproblems may, in some cases, be guaranteed solvable; however, the outer sequence generated by the algorithm may not converge in all cases. A small example from Manne, Chow and Wilson (1983) demonstrates that when income effects are exceptionally large, a backtracking line search is needed for convergence. A second small example from Scarf (1960) demonstrates that local convergence cannot be guaranteed, even in relatively simple models. In spite of these contrived examples, the experience with large-scale empirical models has been excellent.

Two large scale applications which rely on SJM can be cited.<sup>2</sup> Manne and Richels (1992,1995) use the primal form SJM algorithm to solve ATL, a 13 period, 5 region stochastic decision analysis model of energy-economy interactions, carbon emissions and climate change. The nonlinear programming subproblems for ATL involve 6,800 rows, 8,500 columns and 27,000 nonzeros. The SJM iterations typically converge in 5 major iterations (without a line search) to a satisfactory tolerance. Harrison, Tarr and Rutherford (1992) used the dual SJM algorithm for solving MRT, a 23 commodity, 12 region static international trade model with scale economies and imperfect competition. Monopoly markups and scale economies are handled through Jacobi iterations. Again, convergence involves at most 5 iterations.

Both ATL and MRT are of such size that the complementarity and variational approaches seem infeasible.<sup>3</sup> The fact of the matter is that nonlinear programming solvers like MINOS (Murtaugh and Saunders 1982), CONOPT (Drud 1985, 1994) and CNLP (Kallio and Rosa 1994) are efficient and robust, and SJM makes it possible to use them for equilibrium computations.

The remainder of this paper is organized as follows. Section 2 reviews the formulation of general equilibrium models in a complementarity format, indicating how the equilibrium model reduces to an optimization problem when there is a single household. Section 3 introduces Eisenberg's result in the present context, showing how the multi-agent model with heterogeneous, homothetic preferences can be reduced to a single agent model whenever consumers' endowment vectors are strictly proportional. This section also describes the iterative algorithm based on this formulation. Section 4 introduces the dual form of the SJM algorithm. Section 5 relates the SJM algorithm to the complementarity and variational inequality formulations. Section 6 develops necessary conditions for convergence

---

<sup>2</sup> These following claims amount to "proof by testimonial", a technique commonly employed in medical research.

<sup>3</sup> A recently developed complementarity solver, PATH (Ferris and Dirkse 1993) is promising, but it remains to be seen whether it can outperform the optimization-based SJM algorithm, particularly on sparse, large-scale scale models with activity analysis process modules.

of SJM with Scarf's globally unstable exchange model. Section 7 reports on computational experience with large scale models and concludes.

## 2. General Equilibrium in a Complementarity Format

Mathiesen (1985) provides a convenient framework in which to represent both partial and general economic equilibrium models. In Mathiesen's formulation, two types of variables characterize an equilibrium: prices,  $\pi \in \Re^n$ , for  $n$  commodities, and activity levels,  $y \in \Re^m$ , for  $m$  production activities. For simplicity, let an activity analysis matrix  $A = \{a_{ij}\}$  characterize production possibilities.<sup>4</sup> The element  $a_{ij}$  represents the output minus input of commodity  $i$  per unit operation of sector  $j$ . An equilibrium satisfies two classes of conditions:

- 1) Perfectly competitive markets assure that no sector earns an excess profit:

$$-A^T \pi \geq 0$$

- 2) Perfectly flexible prices assure non-positive excess demand for every commodity:

$$Ay - \xi(\pi) \geq 0$$

In the market clearance equation,  $\xi(\pi)$  represents the vector of excess consumer demand for market prices  $\pi$ . This vector is the sum of excess demands for each of the consumers:  $\xi(\pi) = \sum_h (d_h(\pi) - \omega_h)$ , where  $h$  is the vector of initial endowments for consumer  $h$ , and  $d_h(\pi)$  is the vector of final demands for consumer  $h$  which solves:

$$\begin{aligned} \max \quad & U_h(d) \\ \text{s.t.} \quad & \pi^T d \leq \pi^T \omega_h \end{aligned}$$

Note two features of this equilibrium. First, the equilibrium determines only relative prices because the excess profit equations ( $-A^T \pi \geq 0$ ) and the budget constraints ( $\pi^T d \leq \pi^T \omega_h$ ) are linearly homogeneous in  $\pi$ . For this reason, if  $(\pi^*, y^*)$  constitutes an equilibrium then so does  $(\lambda \pi^*, y^*)$  for any  $\lambda > 0$ . The second feature of the equilibrium is complementary slackness. If utility functions are weakly increasing in all inputs and strictly increasing in at least one commodity, then the excess demand vector will obey Walras' law:  $\xi(\pi)^T \pi = 0$ ; and, as a result, equilibrium market excess demands will exhibit complementary slackness with market prices, and equilibrium activity levels will be complementary with unit profits. (Complementary slackness is a feature of the equilibrium, and not an equilibrium condition, per se.)

When there is only one household, the general equilibrium system corresponds to the first-order conditions for the optimization problem:

$$\begin{aligned} \max \quad & U(d) \\ \text{s.t.} \quad & d \leq A^T y + \omega \end{aligned}$$

---

<sup>4</sup> The extension to models with smooth (nonlinear) constant returns to scale production technologies is presented in Appendix A. The activity analysis framework is adopted here to simplify notation and also to emphasize the applicability of this procedure for point-to-set mappings, e.g. staircase-shaped supply curves.

Equilibrium prices correspond to the optimal Lagrange multipliers on the system of constraints. The Karush-Kuhn-Tucker conditions impose non-negative profits and complementary slackness on the production activities.

### 3. Integrable Heterogenous Demand and a Joint Maximization Algorithm

Let us now consider a model in which all utility functions are homothetic and commodity endowments are strictly proportional. Homotheticity assures that the composition of a utility-maximizing commodity bundle is unaffected by the level of income. As income levels are varied while relative prices remain constant, demand quantities all change in fixed proportions (i.e., the Engel curves are straight lines from the origin). Without loss of generality, we may then assume that utility functions are linearly homogeneous.<sup>5</sup>

Define the expenditure function for household  $h$  as:

$$e_h(\pi) = \min \pi^T d$$

$$s.t. \quad U_h(d) = 1$$

Because  $U()$  is linearly homogeneous, the expenditure function provides an index of the “price of a unit of utility”. The maximum attainable utility at market prices is then given by the indirect utility function  $V_h(\pi) = \pi^T \omega_h / e_h(\pi)$ .

Let  $\Omega = \sum_h \omega_h$  stand for the aggregate (world) endowment vector, and assume that there exists a vector  $\theta$  for which  $\omega_h = \theta_h \Omega$ . It can now be shown that the excess demand function representing these preferences and endowments is integrable. Consider the joint maximization problem:

$$\max_{d_h} W(U_1(d_1), \dots, U_H(d_H); \theta) \equiv \prod U_h(d_h)^{\theta_h}$$

$$s.t. \quad \pi^T \sum_h d_h \leq \pi^T \Omega$$

where  $W()$  is a Cobb-Douglas aggregation function into which are nested the utility functions of the different households. Due to homogeneity, we can reduce this problem to:

$$\max \prod U_h(d_h)^{\theta_h}$$

$$s.t. \quad \sum_h e_h(\pi) \leq \pi^T \Omega$$

where expenditure functions  $e_h(\pi)$  are evaluated at prices defined by Lagrange multipliers at the maximum. A global Cobb-Douglas utility function implies demand functions based on fixed expenditure shares for each household. Here, this implies the following

---

<sup>5</sup> An arbitrary function  $f(x)$  is homothetic if and only if there exists a non-decreasing function  $G$  such that  $g(x) = G(f(x))$  is linearly homogeneous in  $x$ , i.e.  $g(\lambda x) = \lambda g(x)$ . These models are based on ordinal rather than cardinal utility, hence  $g(x)$  and  $f(x)$  may be used interchangeably.

It should be possible to accommodate non-homothetic demand using ideas from Perroni (1992), but I leave this extension for future research.

relations:<sup>6</sup>

$$\pi^T d_h = e_h(\pi) U_h = \theta_h \pi^T \Omega = \pi^T (\theta_h \Omega) = \pi^T \omega_h$$

1                      2                      3                      4

In other words, the value of consumer  $h$  expenditures equals the value of consumer  $h$  endowments. This is true because the Cobb-Douglas value shares in  $W(U; \theta)$  correspond exactly to shares of aggregate endowments. It follows that for any price vector  $\pi$ , the demand vector arising from the joint-maximization problem is the same as would result from summing  $H$  independent demand vectors. This result corresponds to Theorem 4 in Eisenberg (1961). (See Chipman (1974) for an economic interpretation.)

In practical applications, endowment vectors are typically not proportional. At the same time, it is typical that changes in relative income are small relative to changes in relative prices or quantities. To account for changes in relative shares of aggregate income, the algorithm applies an iterative refinement of the share parameters  $\theta_h$ .

As an introduction to the algorithm, consider a two-commodity, two-consumer exchange economy in which utility functions have the following (homothetic) form:

$$U_h(x, y) = \alpha_h x^{(\sigma_h - 1)/\sigma_h} (1 - \alpha_h) y^{(\sigma_h - 1)/\sigma_h}.$$

Each agent has a fixed endowment of the goods  $x$  and  $y$ , and in equilibrium the sum of demands from agents  $h = (A, B)$  equals the sum of their endowments. Furthermore, at equilibrium prices, the value of each agent's consumption bundle equals the value of endowments.

It is convenient to use an Edgeworth-Bowley box diagram to illustrate both the equilibrium allocations and the joint maximization algorithm. Figure 1 provides a geometric interpretation of the equilibrium. In this diagram, the length of the horizontal axis measures allocations of commodity  $x$  and the vertical axis allocations of commodity  $y$ . Point  $e$  represents initial endowments - consumer  $A$  is endowed with  $(0.9, 0.1)$  units of  $(x, y)$  and consumer  $B$  is endowed with  $(0.1, 0.9)$  units; thus there is an economy-wide supply of 1 unit of each good. Point  $c$  in figure 1 represents the equilibrium allocation. It lies on the contract curve, the locus of tangency points for agent  $A$  and agent  $B$  iso-utility curves which runs above the diagonal of the factor box. The contract curve lies above the diagonal because agent  $A$  has a stronger preference for  $y$  over  $x$  as compared with agent  $B$ .

The equilibrium point on the contract curve is distinguished as that point from which the line tangent to both agent  $A$  and agent  $B$  utility surfaces runs directly through the endowment point. Thus, at this point both agents are on their respective budget lines. Point  $c$  is an equilibrium allocation for point  $e$  as well as for any point along the line from  $e$  through  $c$ , including point  $d$  which lies on the diagonal of the factor box.<sup>7</sup>

Figure 2 illustrates the sequential joint maximization iterative sequence. The equilibrium values of the Negishi weights are the shares of each agent in the value of global

<sup>6</sup> Justification:

1 : Implied by the definition of  $e_h()$

2 : Follows from the Cobb-Douglas structure of the aggregation function.

3 : Follows from linear algebra.

4 : From the definition of  $\omega_h$  (proportional endowments).

<sup>7</sup> The numerical parameters for this example are  $\alpha_A = 0.3$ ,  $\alpha_B = 0.8$ ,  $\sigma_A = 2$  and  $\sigma_B = 1.5$ . The equilibrium price ratio (good  $x$  in terms of good  $y$ ) is 1.127. A GAMS program which illustrates the various SJM formulations of the Arrow-Debreu model is presented in Appendix B.

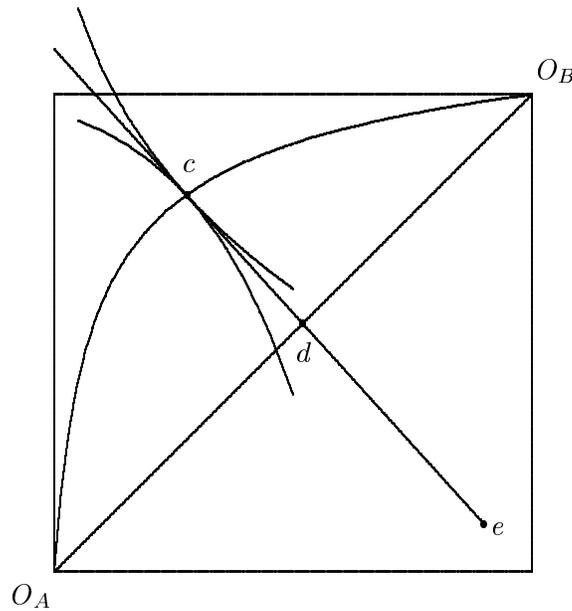


Figure 1. Equilibrium in a 2x2 Exchange Model

endowments. We begin with a price estimate higher than the equilibrium value,  $\pi^0 = 4$ . These prices define a line (with slope  $-4$ ) passing through the endowment point  $e$  and intersecting the diagonal of the factor box at the point labelled  $\theta_1$ . This point lies exactly 0.74 of the way from  $O_A$  to  $O_B$ . This ratio represents the value share of agent  $A$  endowment as a fraction of the total economy-wide endowment, at the initial price estimate ( $\pi^0 = 4$ ):

$$\theta^1 = \frac{0.9\pi^0 + 0.1}{\pi^0 + 1} = \frac{0.9(4) + 0.1}{(4) + 1} = 0.74$$

Given a value for  $\theta^1$ , we then solve the joint maximization problem:<sup>8</sup>

$$\begin{aligned} \max & 1.48 \log(0.3x_A^{1/2} + 0.7y_A^{1/2}) + 0.78 \log(0.8x_B^{1/3} + 0.7y_A^{2/3}) \\ \text{s.t.} & \quad x_A + x_B = 1, \quad y_A + y_B = 1 \end{aligned}$$

The solution to this optimization problem corresponds to point 1 in Figure 2 (the equilibrium allocation for an economy with endowments  $\theta^1$ ). The ratio of the Lagrange multipliers on the  $x$  constraint to the multiplier on the  $y$  constraint equals 0.721. This is taken as the price ratio for the start of iteration 2. Continuing in this fashion, we find successively more precise approximations, as indicated by points  $i_1$ ,  $i_2$  and  $i_3$  in the diagram. Table 1 shows the values of prices, Negishi weights, equilibrium allocations and deviations for 10 iterations of the procedure. Deviations are measured here as the maximum deviation between expenditure and endowment income, as a percentage of endowment income for either agent. In a two agent model, the imbalances for the two agents are always of equal magnitude and opposite sign.

<sup>8</sup> In forming the maximand, the homothetic utility functions have been scaled monotonically to be rendered linearly homogeneous. Hence, we use  $\bar{U}_A = U_A^2$ , so  $\theta \log \bar{U}_A = 2\theta \log U_A$ , and  $\theta \log \bar{U}_B = 3\theta \log U_B$

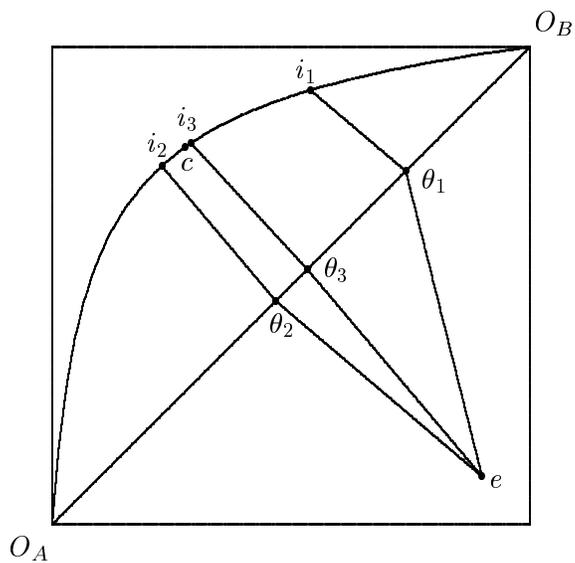


Figure 2. Iterations in the 2x2 model

Table I. Iterative Sequence for the 2x2 Exchange Model.

Iter	$P$	$\theta_A$	$X_A$	$Y_A$	Dev
1	4.000	0.740	0.540	0.910	58.300
2	0.850	0.468	0.229	0.751	12.500
3	1.186	0.534	0.290	0.799	3.400
4	1.086	0.517	0.273	0.787	0.900
5	1.111	0.521	0.277	0.790	0.200
6	1.105	0.520	0.276	0.789	0.100
7	1.106	0.520	0.277	0.790	0.000
8	1.106	0.520	0.277	0.790	0.000
9	1.106	0.520	0.277	0.790	0.000
10	1.106	0.520	0.277	0.790	0.000

Key:

<b>Iter</b>	SJM iteration
$P$	relative price of $X$ to $Y$
$\theta_A$	implied income share for consumer $A$
$X_A, Y_A$	consumption levels for consumer $A$
<b>DEV</b>	percentage deviation in income balance

```

If  $k = 1$  or  $\delta(\Theta^k) < \delta(\theta^{k-1})$ ,
  set  $\theta^{k+1} = \Theta$ ,
   $k \leftarrow k + 1$ 
return to step (1.k)
else
repeat:
   $\lambda \leftarrow \lambda/2$ 
   $\theta = \lambda\theta + (1 - \lambda)\theta^k$ 
  solve JM-P in order to compute  $\delta(\theta)$ 
  if  $\lambda < \underline{\lambda}$  or  $\delta(\theta) < \delta(\theta^k)$ 
    set  $\theta^{k+1} = \theta$ 
    return to step (2.k)
  endif
continue
endif

```

### 3.0.1. Algorithm SJM-P - Formal Statement

*Initialize:* Read an initial estimate of prices,  $\pi^0$ , and set an initial value for the value shares in  $W$ :

$$\theta_h^0 = \frac{\pi^0 \omega_h}{\pi^0 \Omega}$$

Set the iteration index,  $k \leftarrow 1$ , and specify the convergence tolerance ( $\varepsilon$ ) and the minimum step size ( $\underline{\lambda}$ ). Repeat until  $\delta(k) < \varepsilon$ . In iteration  $k = 1, 2, \dots$ :

(1.k) Solve:

$$\begin{aligned} \max \quad & \prod_h U_h(d_h)^{\theta_h} \\ \text{s.t.} \quad & \sum_h d_h \leq Ay \end{aligned} \tag{JM-P}$$

(2.k) Read prices from the Lagrange multipliers,  $\pi(\theta^k)$ .

(3.k) Evaluate the deviation, defined as the largest percentage deviation in any agent's budget constraint:

$$\delta(\theta^k) = \max \left| \frac{\pi(\theta^k)^T [d_h(\theta^k) - \omega_h]}{\pi(\theta^k)^T \omega_h} \right| \times 100$$

(4.k) Perform an Armijo line search (optional)

Initialize  $\lambda = 1$ , and define:

$$\Theta_h = \frac{\pi(\theta^k)^T \omega_h}{\pi(\theta^k)^T \Omega}$$

The foregoing describes the algorithm as it may be applied for a “generic” model. Some adjustments of the procedure are required to treat models with features such as ad-valorem taxes, nonlinear production, non-homothetic utility functions, etc. (See the Appendix B for the key ideas.)

#### 4. Joint Maximization with Dual Functions

In economic analysis, preferences may be described by a primal-form utility function ( $U_h(d)$ ) or, equivalently, using a dual-form indirect utility function ( $V_h(\pi, M_h)$  which expresses welfare as a function of prices and income).<sup>9</sup>

When preferences are homothetic, the unit expenditure function ( $e_h(\pi)$ ) conveys all of the information concerning the underlying preferences. The joint maximization problem (JM-P) can be represented using the expenditure function in place of the utility function, and in many models the computational complexity of the dual-form model is considerably less than for the primal form model.

Let  $\bar{M}_h$  be the current estimate of consumer  $h$  income. Consider then the following optimization problem:

$$\begin{aligned} \max \quad & \sum_h \bar{M}_h \log(e_h(\pi)) - \pi^T \Omega \\ \text{s.t.} \quad & A^T \pi \leq 0 \end{aligned}$$

The constraints for this problem are the zero profit conditions for the underlying equilibrium problem. The dual variables associated with these constraints are interpreted as the activity levels. The market clearance conditions for the equilibrium are incorporated in the first-order optimality conditions. To see this, differentiate the Lagrangian with respect to  $\pi_i$ . We have:

$$\sum_h \frac{\bar{M}_h}{e_h(\pi)} \frac{\partial e_h(\pi)}{\partial \pi_i} - \sum_j a_{ij} y_j - \sum_h \omega_h \leq 0$$

Notice that from the definition of the indirect utility function:

$$\frac{\bar{M}_h}{e_h(\pi)} = V_h(\pi, \bar{M}_h)$$

According to Sheppard's lemma (Varian 1991), the demand for good  $i$  by consumer  $h$  equals the utility level times the gradient of the expenditure function, hence:

$$V_h(\pi, \bar{M}_h) \frac{\partial e_h(\pi)}{\partial \pi_i} = d_{hi}(\pi, \bar{M}_h)$$

Substituting into the first order condition, we see that the optimality conditions for (JM-D) correspond to the market clearance conditions which appear explicitly in (JM-P).

The dual problem is low-dimensional. If we are working with a model in which all goods are demanded by all consumers, problem (JM-D) has  $n$  variables and  $m$  constraints while (JM-P) involves  $n \times p + m$  variables and  $n$  constraints.<sup>10</sup>

As illustration, consider the two agent exchange example from above. In this model, the primal and dual forms are both available in closed form. We have:

$$e_h(p_x, p_y) = (\alpha_h p_x^{1-\sigma_h} + (1 - \alpha_h) p_y^{1-\sigma_h})^{\frac{1}{1-\sigma_h}}$$

<sup>9</sup> See, for example, Varian (1991). Closed form expressions for  $U()$  and  $V()$  are not always available. Most "flexible functional forms" have a dual form but no explicit primal form. The constant elasticity functions used in the examples below admit both primal and dual forms.

<sup>10</sup> When production functions are nonlinear, there are as many additional choice variables in JM-P as there are nonlinear inputs and outputs.

so that when  $\pi^0 = 4$ , the (JM-D) problem is:

$$\begin{aligned} & \max 0.74 \log\left(\frac{3}{p_x} + \frac{7}{p_y}\right) + 0.52 \log\left(\frac{8}{p_x^{1/2}} + \frac{2}{p_y^{1/2}}\right) - p_x - p_y \\ & s.t. \quad p_x \geq 0, p_y \geq 0 \end{aligned}$$

### 5. The Connection between SJM, SLCP and VI

This section relates the sequential joint maximization algorithm to SLCP. To make this correspondence, we need to make a minor reformulation of Mathiesen’s equilibrium conditions, following Rutherford (1987). Consider formulating the model with three classes of variables: commodity prices ( $\pi$ ), activity levels ( $y$ ) and income levels ( $M$ ). An equilibrium satisfies three sets of conditions:

- 1) Zero excess profit:  $-A^T \pi \geq 0$
- 2) Market clearance:  $Ay - \xi(\pi, M) \geq 0$
- 3) Income balance:  $M = \omega \pi$

Apart from the intermediate income variables, these conditions are identical to Mathiesen’s formulation. The vector of excess consumer demand remains the sum of excess demands for each of the consumers:  $\xi(\pi, M) = \sum_h (d_h(\pi, M_h) - \omega_h)$ , but the demand vectors  $d_h(\pi, M_h)$  here solve:

$$\begin{aligned} & \max U_h(d) \\ & s.t. \quad \pi^T d \leq M_h \end{aligned}$$

SLCP is a Newton method which takes into account inequality constraints and complementary slackness conditions. In each iteration, the system of nonlinear inequalities is approximated by a first-order Taylor series expansion, and the resulting linear system is solved using Lemke’s algorithm (Lemke 1965, Anstreicher, Lee and Rutherford 1992).

SJM may be interpreted as a version of SLCP in which prices and activity levels are solved in a simultaneous nonlinear system of inequalities while income levels are adjusted using Gauss-Seidel iterations. The optimization step of the joint maximization algorithm solves the system of zero profit and market clearance conditions, the “fixed income relaxation.” During these computations, the income balance equations are (temporarily) ignored. After computing an equilibrium for this nonlinear inequality system, the income variables can be updated using the resulting market prices. Exact replication of the SJM iterative sequence requires that the nonlinear system of market clearance and zero profit conditions be solved completely between income revisions. In the numerical tests reported below, this algorithm is named SJM-C (sequential joint maximization with complementarity subproblem).

A mixture of the SJM and SLCP approaches is related to the variational inequality algorithm proposed by Goldsman and Harker (1990). In the VI algorithm, a fixed-income relaxation is employed for linearized subproblems while incomes levels are revised in every Newton iteration. Subproblems have the form:

$$\begin{aligned} & -A^T \pi \geq 0 \quad \perp y \geq 0 \\ & Ay - \nabla \xi(\pi, \bar{M}) \pi \geq \xi(\pi, \bar{M}) \quad \perp \pi \geq 0 \end{aligned}$$

This linear complementarity problem corresponds to the first-order conditions for the following quadratic program:

$$\begin{aligned} \max \quad & b^T \pi + \frac{1}{2} \pi^T Q \pi \\ \text{s.t.} \quad & A^T \pi \leq 0 \end{aligned}$$

where  $b = \xi(\bar{\pi}, \bar{M})$  and  $Q = \nabla \xi(\pi, \bar{M})$ .

The iterative sequence produced by this sequential quadratic programming algorithm will differ from the VI algorithm because income levels are treated as exogenous in each subproblem. By decoupling income and price adjustments, the required number of Newton iterations may increase relative to SLCP. At the same time, this approach may help to avoid defective subproblems which can plague SLCP. (See Mathiesen 1987.)

## 6. Convergence

Computational experiments demonstrate that a line search is sometimes necessary for convergence. This is the case for a modified version of the Jack-Sprat problem presented in Manne, Chao and Wilson (1980). When Jack-Sprat is initiated from a particular starting point, the SJM procedure with  $\underline{\lambda} = 1$  (i.e. without a line search) after four iterations begins to cycle between two sets of Negishi weights. When the Armijo line search is applied, the outer iterations converge in five steps. (See model Jack-Sprat in Appendix B.)

The second small model presents more serious theoretical problems for the algorithm. Scarf (1960) provides a model which demonstrates that the SJM *differential process* (for  $\lambda \rightarrow 0$ ) may be non-convergent. Furthermore, it can be shown that the conditions for convergence of the Negishi process locally reduce to exactly the same conditions for convergence of a simple Tatonnement adjustment process.

Scarf's model involves an equal number of  $n$  consumers and goods. Consumer  $h$  is endowed with 1 unit of good  $h$  and demands only goods  $h$  and  $h + 1$ . Let  $d_{ih}$  represent demand for good  $i$  by consumer  $h$ . Preferences are represented by constant elasticity of substitution utility functions with the following structure:

$$U_h(d) = \left( \theta^{1/\sigma} d_{hh}^{\frac{\sigma-1}{\sigma}} \right)^{\frac{\sigma}{\sigma-1}}$$

There are two utility function parameters.<sup>11</sup>  $\theta$  can be interpreted as the benchmark value share of good  $i$  in consumer  $i$  demand, and  $1 - \theta$  is then the benchmark value share of good  $i + 1$ . (Consumer  $n$  demands goods  $n$  and 1.)

The joint maximization algorithm works with consumer income levels, denoted  $M_h$  for household  $h$ . These may be normalized so that  $\sum_h M_h = n$ . Market clearing commodity prices are determined given the income levels. Let  $\pi_i(M)$  denote the price of good  $i$

consistent with income levels  $M = \begin{pmatrix} M_1 \\ M_2 \\ \vdots \\ M_H \end{pmatrix}$ . Provided that  $\sigma > 0$ , utility functions exhibit non-satiation, so  $\sum_i \pi_i = \sum_h M_h$ .

<sup>11</sup> The parameters of this function correspond to Scarf's parameters  $a$  and  $b$  (Scarf (1960, page 168)) as:  $\sigma = \frac{1}{1+a}$  and  $\theta = \frac{b}{1+b}$

Let  $\xi(p; M)$  denote the market excess demand function for fixed income. Given the special structure of preferences and endowments, this function has the form:

$$\xi_i(p; M) = d_{ii} + d_{i,i-1} - 1$$

Given our definition of  $p(M)$ , we have  $x_i(p(M), M) = 0 \forall i$ .

Let  $H_h(M)$  denote the difference between the value of agent  $h$  allocated income and the market value of the agent  $h$  endowment for prices  $p(M)$ . That is:

$$H_h(M) = p_h(M) - M_h$$

The structure of demand and endowments assure an equilibrium in which all income levels and prices equal unity. When  $M_i^* = 1$  then  $\pi_i(M^*) = 1 \forall i$ , and

$$H_h(M^*) = 0 \forall h$$

The SJM algorithm, as the step length goes to zero, represents an *income adjustment process*:

$$M_h = p_h(M) - M_h$$

Let the initial estimate  $M^0$  be selected on the  $n$ -simplex (i.e.,  $\sum_h M_h^0 = n$ ). The adjustment process then remains on the  $n$ -simplex:

$$\frac{d}{dt} \sum_h M_h = \sum_h M_h = \sum_h (p_h - M_h) = \sum_i p_i(M) - \sum_h M_h = 0$$

Local convergence concerns properties of the Jacobian matrix evaluated at the equilibrium point,  $\nabla H(M^*) = [H_{ij}]$ . This Jacobian has entries which are defined as follows:

$$H_{ij} \equiv \frac{\partial H_i}{\partial M_j} = \begin{cases} \frac{\partial p_i}{\partial M_i} - 1 & i = j \\ \frac{\partial p_i}{\partial M_j} & i \neq j \end{cases}$$

If all principal minors of  $\nabla H(M^*)$  are negative, the income adjustment process is locally convergent. If, however,  $\frac{\partial p_i}{\partial M_h} > 1$ , the process is "locally unstable" – a small increase from the equilibrium income level for consumer  $h$  causes consumer  $h$  endowment revenue to increase more than proportionally. When an equilibrium is unique and the process is uninterrupted, then local instability implies global instability.

For this model, the Tatonnement price adjustment process is unstable (in the case  $n=3$ ) when  $\frac{\theta}{1-\theta} > \frac{1}{1-2\sigma}$  (Scarf, 1960). It is shown in the following that the same condition implies instability for the income adjustment process, even though these algorithms produce different search directions away from a neighborhood of the equilibrium.

The function  $p(M)$  is defined implicitly by the equation:

$$\xi(p; M) = 0$$

In order to evaluate  $\nabla p$  at  $M^*$ , we make a first-order Taylor series expansion:

$$\nabla_p \xi(p; M) dp + \nabla_M \xi(p; M) dM = 0,$$

So

$$\nabla p = \left[ \frac{\partial p_i}{\partial M_h} \right] = -\nabla_p^{-1} \nabla_M \xi$$

Given the special structure of  $\xi_i(p; M)$ , we have:

$$\frac{\partial \xi_i}{\partial p_i} = \frac{\partial d_{ii}}{\partial p_i} + \frac{\partial d_{i,i-1}}{\partial p_i}, \quad \frac{\partial \xi_i}{\partial p_{i-1}} = \frac{\partial d_{i,i-1}}{\partial p_{i-1}}, \quad \text{and} \quad \frac{\partial \xi_i}{\partial p_{i+1}} = \frac{\partial d_{ii}}{\partial p_{i+1}}.$$

If we define the "unit-utility" expenditure function for consumer  $i$  as:

$$e_i(p) = (\theta p_i^{1-\sigma} + (1-\theta)p_{i+1}^{1-\sigma})^{\frac{1}{1-\sigma}}$$

demand functions are:

$$d_{ii} = \frac{\theta M_i}{e_i^{1-\sigma} p_i^\sigma}, \quad \text{and} \quad d_{i+1,i} = \frac{(1-\theta)M_i}{e_i^{1-\sigma} p_{i+1}^\sigma}$$

Evaluating gradients at  $p^* = 1$ , it is apparent that the "fixed income Slutsky matrix" has a tri-diagonal structure:

$$\nabla_p \xi = \begin{bmatrix} \delta & \alpha & 0 & \dots & 0 & \alpha \\ \alpha & \delta & \alpha & 0 & \vdots & 0 \\ 0 & \alpha & \delta & \alpha & & \vdots \\ \vdots & & & & & \\ 0 & 0 & \dots & \alpha & \delta & \alpha \\ \alpha & 0 & \dots & 0 & \alpha & \delta \end{bmatrix}$$

where  $\delta = -\sigma - (1-\sigma)[\theta^2 + (1-\theta)^2]$  and  $\alpha = -(1-\sigma)\theta(1-\theta)$ . The inverse matrix,  $\nabla_p \xi^{-1}$ , is:

$$\nabla_p^{-1} \xi = \begin{bmatrix} \gamma & \beta & \beta & \dots & \beta \\ \beta & \gamma & \beta & \dots & \beta \\ \beta & \beta & \gamma & \dots & \vdots \\ \vdots & & & \ddots & \\ \beta & \dots & \beta & \gamma \end{bmatrix}$$

where  $\gamma = \frac{-(\alpha+\delta)}{(2\alpha+\delta)(\alpha-\delta)}$  and  $\beta = \frac{\alpha}{(2\alpha+\delta)(\alpha-\delta)}$ .

The gradients of excess demands with respect to income at the equilibrium are:

$$\nabla_M \xi = \begin{bmatrix} \theta & 0 & 0 & \dots & 1-\theta \\ 1-\theta & \theta & 0 & \dots & 0 \\ 0 & 1-\theta & \theta & \dots & \vdots \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & 1-\theta & \theta \end{bmatrix}$$

Hence:

$$\frac{\partial p_i(M)}{\partial M_i} = \theta\delta + (1-\theta)\beta$$

Local instability for  $n \geq 3$  therefore implies:

$$\frac{\sigma\theta^2 - \theta^2 - \sigma\theta}{3(\sigma\theta^2 - \theta^2 - \sigma\theta) + 3\theta - 1} \geq 1$$

which reduces to the same condition as Scarf demonstrated for the Tatonnement process when  $n = 3$ .

Given the equivalent local behavior, one might be lead to believe that the income and price-adjustment processes are identical. This is not true. Only local to the equilibrium, where price effects dominate income effects, do these processes follow the same path. This is apparent in Figure 3 where the two vector fields are superimposed. In this diagram, the tatonnement field becomes more divergent than the SJM field as one moves further from the equilibrium point at the center of the simplex. At the equilibrium point the two fields coincide exactly, as demonstrated above.

## 7. Conclusion

This paper has described a new income adjustment procedure which makes it easy to apply the joint maximization algorithm proposed originally by Negishi and Dixon. The paper presented three alternative implementations of the solution procedure, based on primal optimization, dual optimization and complementarity subproblems.

Scarf's model demonstrates that convergence of SJM cannot be guaranteed. At the same time, the SJM approach seems to be the only reliable method for solving large scale models with activity analysis, such as Manne and Richels (1995). Even though SJM has neither the theoretical pedigree of a fixed-point algorithm nor the local convergence properties of a Newton algorithm, the usefulness of the SJM algorithm is demonstrated by the range of real models for which the procedure has been successfully applied.

## References

- Anstreicher, Kurt M., Jon Lee and Thomas F. Rutherford "Crashing a Maximum-Weight Complementary Basis," *Mathematical Programming*, 54(3), 1992, pp. 281-294.
- Carey, Malachy "Integrability and Mathematical Programming Models: A Survey and a Parametric Approach", *Econometrica*, 45(8), 1977, pp. 1957-1976.
- Chipman, John S. "Homothetic Preferences and Aggregation". *Journal of Economic Theory*, 8, 1974, pp. 26-38.
- Dantzig, G.B., B.C. Eaves, and D. Gale "An Algorithm for the Piecewise Linear Model of Trade and Production with Negative Prices and Bankruptcy", *Mathematical Programming*, 16(2), 1979, pp. 190-209.
- Dirkse, Steven and Michael C. Ferris, "The PATH Solver: A Non-Monotone Stabilization Scheme for Mixed Complementarity Problems", *Optimization Methods and Software*, 5, 1995, pp. 123-156.
- Dixon, Peter J. *The Theory of Joint Maximization*, North-Holland, 1975.
- Drud, Arne Stolbjerg "CONOPT: A GRG Code for Large Sparse Dynamic Nonlinear Optimization Problems", *Mathematical Programming* 31(2), 1985, pp. 153-191.
- Drud, Arne Stolbjerg "CONOPT: A Large Scale GRG Code", *ORSA Journal on Computing*, 6(2), 1994, pp. 207-216.
- Eisenberg, E. "Aggregation of Utility Functions", *Management Science* 7(4), 1961, pp. 337-350.
- Ginsburgh, Victor .A. and Ludo Van der Heyden, "On Extending the Negishi Approach to Computing Equilibria: The Case of Government Price Support Poicies", *Journal of Economic Theory*, 44, 1988, pp. 168-178.
- Goldsman, Lynn and Patrick Harker "A Note on Solving General Equilibrium Problems with Variational Inequality Techniques", *OR Letters* (9), 1990, pp. 335-339.
- Harrison, Glenn, Thomas F. Rutherford, and David Tarr "Increased Competition and Completion of the Market in the European Union: Static and Steady-State Effects", *Journal of Economic Integration* 11(3), forthcoming 1996.

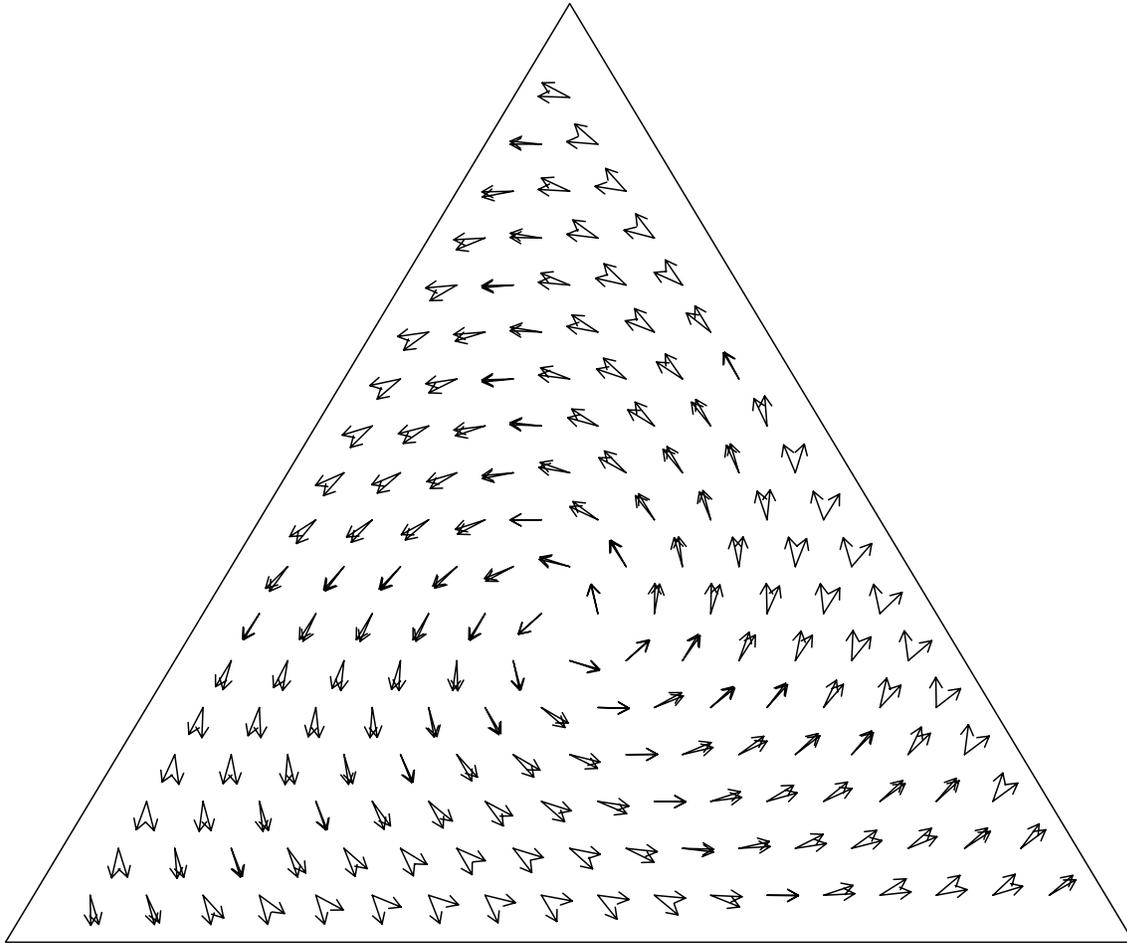


Figure 3. Tatonnement and SJM Vector Fields Compared

- Harrison, Glenn, Thomas F. Rutherford, and David Tarr "Quantifying the Uruguay Round", *The Economic Journal*, forthcoming 1997.
- Kallio, M. and C.H. Rosa "Large-Scale Convex Optimization with Saddle Point Computation", IIASA Working Paper WP-94-107, October, 1994.
- Kehoe, Timothy J., David K. Levine and Paul M. Romer "On Characterizing Equilibria of Models with Externalities and Taxes as Solutions to Optimization Problems", *Economic Theory* 2(1), 1992, pp. 43-68.
- Lemke, C.E. "Bimatrix Equilibrium Points and Mathematical Programming", *Management Science* 11(7), 1965, pp. 681-689.
- Manne, Alan S., Hung-Po Chao, and Robert Wilson, "Computation of Competitive Equilibria by a Sequence of Linear Programs", *Econometrica* 48(7), 1980, pp. 1595-1615.
- Manne, Alan S. and Richard B. Richels, *Buying Greenhouse Insurance*, MIT Press, 1992.
- Manne, Alan S. and Richard Richels "The Greenhouse Debate - Economic Efficiency, Burden Sharing and Hedging Strategies", *Energy Journal* 16(4), 1995, pp. 1-37.
- Manne, Alan and Thomas F. Rutherford (1992) "International Trade, Capital Flows and Sectoral Analysis: Formulation and Solution of Intertemporal Equilibrium Models", *New Directions in Computational Economics*, W.W. Cooper and A.B. Whinston (editors), 1994, pp. 191-205.
- Mathiesen, Lars "Computation of Economic Equilibrium by a Sequence of Linear Complementarity Problems", *Mathematical Programming Study* 23, North-Holland, 1985, pp. 144-162.
- Mathiesen, Lars "An Algorithm Based on a Sequence of Linear Complementarity Problems Applied to a Walrasian Equilibrium Model: An Example", *Mathematical Programming* 37(1), 1987, pp. 1-18.

- Murtaugh, Bruce A. and Michael A. Saunders "A Projected Lagrangian Algorithm and its Implementation for Sparse Nonlinear Constraints", *Mathematical Programming Study 16: Algorithms for Constrained Minimization of Smooth Nonlinear Functions*, North-Holland, 1982, pp. 84-117, .
- Negishi, T. "Welfare Economics and the Existence of an Equilibrium for a Competitive Economy", *Metroeconomica* 12, 1960, pp. 92-97.
- Perroni, Carlo "Homothetic Representation of Regular Non-Homothetic Preferences" *Economics Letters* 40(1), 1992, pp. 19-22.
- Rutherford, Thomas F. "A Modeling System for Applied General Equilibrium Analysis", Yale Cowles Foundation Discussion Paper 836, 1987.
- Samuelson, Paul A. "Social Indifference Curves", *Quarterly Journal of Economics* 70(1), 1956.
- Scarf, Herbert "Some Examples of Global Instability of the Competitive Equilibrium", *International Economic Review* 1 (3), 1960, pp. 157-172.
- Scarf, Herbert (with Terje Hansen) *The Computation of Economic Equilibria*, Yale University Press, 1973.
- Varian, Hal *Microeconomic Analysis*, Norton, 1991.

## Appendix

### A. Extensions of the SJM Algorithm

#### A.1. NONLINEAR PRODUCTION

As in the body of the paper, let  $e_h(\pi)$  represent the unit expenditure function for consumer  $h$ . Let  $\phi_j(\pi)$  be the analogous "unit revenue function" for sector  $j$ , defined as:

$$\phi_j(\pi) \equiv \max \pi^T x \quad \text{s.t. } x \in T^j$$

where  $T_j$  is the feasible set for the constant returns to scale technology for sector  $j$ , operated at unit intensity.

Shepard's lemma characterizes provides optimal producer netput (output minus input per unit activity):

$$x^j(\pi) = \nabla \phi_j(\pi)$$

A competitive equilibrium is supported by a solution to:

$$\max \sum_h \bar{M}_h \log(e_h(\pi)) - \pi^T \omega_h$$

$$\text{s.t.} \quad \phi_j(\pi) \leq 0 \quad \forall j$$

which has first-order conditions:

$$\sum_j \nabla \phi_j(\pi) y_j + \sum_h \omega_h \geq \sum_h \nabla e_h(\pi) \frac{\bar{M}_h}{e_h(\pi)}$$

#### A.2. AD-VALOREM TAXES

In the general equilibrium structure, price distortions can, without loss of generality, be applied only to producer inputs. Suppose that  $x_j$  is chosen to solve:

$$\max \hat{\pi}^T x$$

$$\text{s.t.} \quad x \in T^j$$

in which  $\hat{\pi}$  is a vector of tax-distorted prices (users costs), for example:  $\hat{\pi}_{ij} = \pi_i(1 + t_{ij})$ . When tax distortions are present, the tax revenue returned per unit operation of sector  $j$  is given by  $\pi - \hat{\pi})^T x$ .

In order to accommodate price distortions in the dual joint maximization procedure, the tax distortions are introduced into the constraints, and the tax revenue effects are treated symmetrically with factor endowments - using lagged values for production activities. The generic dual-form optimization problem is:

$$\max \sum_h \left[ \bar{M}_h \log(e_h(\pi)) - \pi^T \omega_h - \sum_j \tilde{y}_j (\pi - \hat{\pi})^T \tilde{x}^j \right]$$

$$\text{s.t.} \quad \phi_j(\hat{\pi}) \leq 0$$

Within the outer loop of the SJM algorithm, income levels are updated by:

$$M_h = \pi^T \omega_h + \sum_j \theta_{jh} \tilde{y}_j (\pi - \hat{\pi})^T \tilde{x}^j$$

in which  $\theta_{jh}$  is the share of sector  $j$  tax revenue which accrues to household  $h$ .

## Appendix

## A. GAMS Code

## A.1. TWO-BY-TWO EXCHANGE

\$TITLE: A pure exchange model solved with joint maximization

SET H HOUSEHOLDS / A, B/

ALIAS (H,HH);

PARAMETER

EX(H) ENDOWMENTS OF X /A 0.9, B 0.1/  
 EY(H) ENDOWMENTS OF Y /A 0.1, B 0.9/  
 ALPHA(H) PREFERENCE FOR X /A 0.3, B 0.8/  
 SIGMA(H) ELASTICITY OF SUBSTITUTION /A 2.0, B 1.5/  
 RHO(H) ELASTICITY PARAMETER (PRIMAL FORM)  
 THETA(H) NEGISHI WEIGHTS;

RHO(H) = (SIGMA(H)-1)/SIGMA(H) ;

VARIABLES U(H) UTILITY LEVELS  
 X(H) FINAL DEMAND FOR X  
 Y(H) FINAL DEMAND FOR Y  
 NEGISHI NEGISHI ITERATION MAXIMAND

EQUATIONS OBJDEF DEFINES THE SJM MAXIMAND  
 UDEF(H) DEFINES UTILITY INDICES  
 EQUILX MARKET CLEARANCE FOR GOOD X  
 EQUILY MARKET CLEARANCE FOR GOOD Y;

OBJDEF..

NEGISHI =E= SUM(H, THETA(H) \* LOG(U(H)));

UDEF(H)..

U(H) =E= ( ALPHA(H)\*\*(1/SIGMA(H)) \* X(H)\*\*RHO(H) +  
 (1-ALPHA(H))\*\* (1/SIGMA(H)) \* Y(H)\*\*RHO(H) )\*\*(1/RHO(H));

EQUILX..

1 =E= SUM(H, X(H));

EQUILY..

1 =E= SUM(H, Y(H));

```

MODEL SJM /OBJDEF, UDEF, EQUILX, EQUILY /;

THETA(H) = 1;

U.LO(H) = 0.01;
X.LO(H) = 0.01;
Y.LO(H) = 0.01;

PARAMETER      ITLOG   ITERATION LOG;

SCALAR         PXBAR   RELATIVE PRICE OF X IN TERMS OF Y /4/;

SET ITER       NEGISHI ITERATIONS /1*10/;

LOOP(ITER,

    ITLOG(ITER,"P") = PXBAR;

    THETA(H) = (EX(H) * PXBAR + EY(H))
               / SUM(HH, EX(HH) * PXBAR + EY(HH));

    ITLOG(ITER,"THETA") = THETA("A");

    SOLVE SJM USING NLP MAXIMIZING NEGISHI;

    ITLOG(ITER,"X") = X.L("A");
    ITLOG(ITER,"Y") = Y.L("A");

    PXBAR = EQUILX.M / EQUILY.M;

    ITLOG(ITER,"DEV") = ROUND(100 *
        ABS( PXBAR*X.L("A") + Y.L("A")
             - EX("A")*PXBAR - EY("A"))
        / ( EX("A") * PXBAR + EY("A") ), 1);
);
DISPLAY ITLOG;

```

## A.2. MANNE'S JACK-SPRAT EXAMPLE

\$TITLE: The Jack Sprat Problem

- \* Prices highly depend on income distribution
- \* Alan Manne 12/28/91
- \* The relaxation parameter is theta.

- \* The SJM procedure cycles when  $\theta = 0$ , converges
- \* when  $\theta = 0.5$ .
- \* From: A. S. Manne, H-P Chao, and R. Wilson:
- \* Computation of Competitive Equilibria by a Sequence of
- \* Linear Programs
- \* Econometrica, Vol. 48, No. 7, November 1980, p 1595-1615.
- \*
- \* with a nasty modification of original  $a(i,j)$  table.

```

set i commodities      / 1*5 /
    j activity levels / 1*5 /
    h households      / jack,wife /

```

```

table a(i,j)  input-output matrix

```

	1	2	3	4	5
1	-1	.5		-1	-1
2		-1		-1	
3			-1		-1
4				1	
5					1

- \* Note: in the version of this problem appearing in MCW,
- \* the (1,2) coefficient 0.5 is equal to zero.

```

table b(i,h)  endowments

```

	jack	wife
1	1.	1.
2		1.5
3	1.5	

```

table u(i,h)  requirements for good i per unit of utility

```

	jack	wife
4	1	
5		1

```

parameters nwt(h)  Negishi weights - initial guess

```

/jack	.1	
wife	.9	/

```

scalar theta  relaxation parameter / 0 /

```

\* Turn off solution output for the Negishi iterations:

```
option limrow = 0;
option limcol = 0;
option sysout = off;
option solprint = on;
```

```
variables y(j)      activity levels
          x(h)      utility levels
          gw        global welfare function
```

positive variables y,x

```
equations sd(i)    supply-demand balance
          gw        global welfare definition;
```

sd(i)..

```
sum(j, a(i,j)*y(j) ) + sum(h, b(i,h)) =e= sum(h, u(i,h)*x(h));
```

gw..

```
sum(h, nwt(h)*log(x(h))) =e= gw
```

model jacksp / all /;

```
x.lo(h) = .01;
```

\* declare Negishi iteration limit

```
set iter /it1*it10/;
```

parameters

```
      p(i)          prices
      n(h)          endowment values
      tn           total endowment value
      rwt(iter,h)   revised Negishi weights;
```

\* loop over revised Negishi weights

loop(iter,

```
      rwt(iter,h) = nwt(h);
      solve jacksp using nlp maximizing gwf;
```

```

p(i)      = - sd.m(i);
n(h)      =  sum(i, p(i)*b(i,h));
tn        =  sum(h,n(h));
nwt(h)    =  theta*nwt(h) + (1 - theta)*n(h)/tn
);

display p,n,tn,rwt;

```

### A.3. SCARF'S GLOBALLY UNSTABLE EXCHANGE MODEL

```

$title: Scarf's globally unstable exchange model

* This program computes SJM and Tatonnement search
* directions on a grid over the 3-simplex.

FILE KSJM /SJM.DAT/; KSJM.NR = 2; KSJM.NW = 15; KSJM.ND = 6;
FILE KTAT /TAT.DAT/; KTAT.NR = 2; KTAT.NW = 15; KTAT.ND = 6;

SET ITER TATONNEMENT ITERATION COUNT /IT1*IT2000/

SET H HOUSEHOLDS AND GOODS /G1*G3/;

ALIAS (I,H), (J,H);

PARAMETER
    WEIGHT(H)      INCOME BY HOUSEHOLD,
    PP(I)          PRICE BY COMMODITY (FOR TATONNEMENT)
    EE(I)          EXPENDITURE BY HOUSEHOLD
    X(I)           EXCESS DEMAND;

SCALAR
    THETA          OWN GOOD BUDGET SHARE,
    SIGMA          ELASTICITY PARAMETER /0.4/,
    LAMBDA         SPEED OF ADJUSTMENT /0.1/;

*      SET THETA SO THAT TATONNEMENT IS marginally UNSTABLE:

THETA = 0.5 / (1 - SIGMA) + 0.15;

ABORT$(THETA LT 0) " THETA IS LESS THAN ZERO?";
ABORT$(THETA GT 1) " THETA IS GREATER THAN 1?";

VARIABLE
    E(H)          EXPENDITURE FUNCTION
    P(J)          MARKET PRICE
    OBJ           OBJECTIVE FUNCTION;

```

```
EQUATIONS      EDEF(H)
                OBJDEF;
```

```
EDEF(H)..
```

```
E(H)**(1-SIGMA) =E= THETA * P(H)**(1-SIGMA) +
                  (1-THETA) * P(H++1)**(1-SIGMA);
```

```
OBJDEF..
```

```
OBJ =E= SUM(H, WEIGHT(H) * LOG(E(H))) - SUM(J, P(J));
```

```
MODEL   SJM /ALL/;
```

```
E.LO(H) = 0.01;
```

```
P.LO(H) = 0.01;
```

```
E.L(H) = 1;
```

```
P.L(H) = 1;
```

```
*      EVALUATE DIRECTIONS ON A GRID ON THE 3-SIMPLEX:
```

```
SCALAR  NGRID  NUMBER OF GRID POINTS FOR FLAGS /20/,
         M      GRID POINTS IN SUBSIMPLEX
         NPOINT FULL SIMPLEX POINT COUNT
         NP     SUBSIMPLEX POINT COUNT
         INDP   PRICE INDEX
         LENGTH FLAG LENGTH;
```

```
*      THIS LENGTH SEEMS ABOUT RIGHT:
```

```
LENGTH = 3 / (2 * NGRID);
```

```
*      TOTAL NUMBER OF GRID POINTS:
```

```
NPOINT = (NGRID+1) * (NGRID+2) / 2;
```

```
LOOP(ITER$(ORD(ITER) LE NPOINT),
```

```
    INDP = ORD(ITER);
```

```
*      THESE STATEMENTS GENERATE A GRID ON THE
```

```
* PRICE SIMPLEX:
```

```
M = ROUND(0.5 * SQRT(1.0+8.0*INDP) - 1.5);
```

```
IF (M GT (0.5 * SQRT(1.0+8.0*INDP) - 1.5), M = M - 1);
```

```

NP = (M+1) * (M+2) / 2;
IF (NP LT INDP, M = M + 1; NP=(M+1)*(M+2)/2; );

WEIGHT("G3") = NGRID - M;
WEIGHT("G1") = NP - INDP;
WEIGHT("G2") = NGRID - WEIGHT("G1") - WEIGHT("G3");

*   SCALE TO THE 3-SIMPLEX:

WEIGHT(I) = WEIGHT(I) / SUM(J, WEIGHT(J));

*   SKIP POINTS ON THE BOUNDARY:

IF (SMIN(I, WEIGHT(I)) GT 0,

*   Generate the SJM direction:

    PUT KSJM; LOOP(I, PUT WEIGHT(I));
    SOLVE SJM USING NLP MAXIMIZING OBJ;
    PUT KSJM; LOOP(I, PUT P.L(I)); PUT /;

*   Generate the Tatonnement direction:

    PP(I) = 3 * WEIGHT(I);

    EE(H) = ( THETA * PP(H)**(1-SIGMA) +
              (1-THETA) * PP(H++1)**(1-SIGMA))**(1/(1-SIGMA));

*   EVALUATE EXCESS DEMANDS:

    X(I) = THETA * (PP(I)/EE(I))
           * (EE(I)/PP(I)**SIGMA +
             (1-THETA) * (PP(I--1)/EE(I--1))
             * (EE(I--1)/PP(I)**SIGMA - 1;

*   SCALE ALL THE ARROWS TO BE THE SAME LENGTH:

    LAMBDA = LENGTH / SQRT(SUM(I, X(I) * X(I)));

    PUT KTAT; LOOP(I, PUT (PP(I)/3)););
    PP(I) = PP(I) + LAMBDA * X(I);
    PUT KTAT; LOOP(I, PUT (PP(I)/3)); PUT /;

);
);

```

## A.4. HANSEN'S ACTIVITY ANALYSIS EXAMPLE

\$TITLE: Joint Maximization: Three Alternative Formulations

\$ontext

Reference: Herbert Scarf with Terje Hansen (1973)  
The Computation of Economic Equilibria,  
Yale University Press.

This program generates the following output on a 90 MHz Pentium:

```

----      291 PARAMETER ITLOG           Iteration log

                DEV           CPU           ITER
PRIMAL.ITER1      9.8           1.0          243.0
PRIMAL.ITER2       0.9           0.4          22.0
PRIMAL.ITER3 7.748711E-2         0.3          19.0
PRIMAL.ITER4 6.915353E-3         0.3          15.0
PRIMAL.TOTAL                2.0          299.0

DUAL  .ITER1      9.8           0.4          40.0
DUAL  .ITER2       0.9           0.3           4.0
DUAL  .ITER3 7.746462E-2         0.3           4.0
DUAL  .ITER4 6.914645E-3         0.3           3.0
DUAL  .TOTAL                1.3          51.0

MCP   .ITER1      9.8           0.4          14.0
MCP   .ITER2       0.9           0.3           2.0
MCP   .ITER3 7.746431E-2         0.3           1.0
MCP   .ITER4 6.914911E-3         0.3           1.0
MCP   .TOTAL                1.3          18.0

```

For the most part, this is a standard GAMS program. There is one subtlety involved in representing the functions which may appear somewhat obscure to non-GAMS users. In evaluating the SJM maximand, we want to write:

```

WELFARE =E= SUM(H, THETA(H) * (1/RHO(H))

      * LOG( SUM(C, ALPHA(C,H) * (D(C,H)/DBAR(C,H))**RHO(H)) ) );

```

but we need to account for some special cases. For one thing, if household H has Cobb-Douglas preferences, RHO(H) goes to -INF and the function form (by L'Hopital's lemma) becomes:

```
SUM(C, ALPHA(C,H) * LOG(D(C,H)/DBAR(C,H)))
```

To incorporate both functional forms in a single statement, I have used the GAMS dollar operator, writing:

```
WELFARE =E= SUM(H, THETA(H) *
( ( (1/RHO(H)) *
  LOG( SUM(C, ALPHA(C,H) * (D(C,H)/DBAR(C,H))**RHO(H))
  )$(ESUB(H) NE 1) +
(
  SUM(C, ALPHA(C,H) * LOG(D(C,H)/DBAR(C,H)))
  )$(ESUB(H) EQ 1)
) );
```

This simply says, use the CES form if ESUB(H) is not unity, and use Cobb-Douglas if ESUB(H) is unity.

One additional adjustment of the equation is required because some of the share parameters, ALPHA(C,H), may be zero. To avoid evaluating 0/0, I have put another dollar exception operator on the loops over C.

It is an unfortunate fact of life that computers require a precise statement of things, and they get cranky if you ask them to evaluate functions involving 0 times infinity.

\$offtext

```
SETS  C          COMMODITIES
/     AGRIC, FOOD, TEXTILES, HSERV, ENTERT, HOUSEOP, CAPEOP,
      STEEL, COAL, LUMBER, HOUSBOP, CAPBOP, LABOR, EXCHANGE/
      N(C)      NUMERAIRE /LABOR/
      H          CONSUMERS
/     AGENT1, AGENT2, AGENT3, AGENT4 /

      S          SECTORS
/     DOM1, DOM2, DOM3, DOM4, DOM5, DOM6, DOM7,
      DOM8, DOM9, DOM10, DOM11, DOM12,
      IMP1, IMP2, IMP3, IMP4, IMP5, IMP6, IMP7,
      EXP1, EXP2, EXP3, EXP4, EXP5, EXP6, EXP7 /
```

ALIAS (C,CC);

TABLE E(C,H) Commodity endowments

	AGENT1	AGENT2	AGENT3	AGENT4
HOUSBOP	2	0.4		0.8
CAPBOP	3	2		7.5
LABOR	0.6	0.8	1	0.6

TABLE DBAR(C,H) Reference demands

	AGENT1	AGENT2	AGENT3	AGENT4
AGRIC	0.1	0.2	0.3	0.1
FOOD	0.2	0.2	0.2	0.2
TEXTILES	0.1	0.1	0.3	0.1
HSERV	0.1	0.1	0.1	0.1
ENTERT	0.1	0.1	0.1	0.1
HOUSEOP	0.3	0.1		0.1
CAPEOP	0.1	0.2		0.3;

PARAMETER ESUB(H) Elasticities in demand

/	AGENT1	1,	AGENT2	1,
	AGENT3	1,	AGENT4	1 /;

TABLE IODATA(\*,C,S) Activity analysis matrix

	DOM1	DOM2	DOM3	DOM4	DOM5	
OUTPUT.AGRIC	5.00					
OUTPUT.FOOD		5.00				
OUTPUT.TEXTILES			2.00			
OUTPUT.HSERV				2.00		
OUTPUT.ENTERT					4.00	
OUTPUT.HOUSEOP				0.32		
OUTPUT.CAPEOP	0.40	1.30	1.20			
INPUT .AGRIC		3.50	0.10		0.70	
INPUT .FOOD	0.90		0.10		0.80	
INPUT .TEXTILES	0.20	0.50		0.10	0.10	
INPUT .HSERV	1.00	2.00	2.00		2.00	
INPUT .STEEL	0.20	0.40	0.20	0.10		
INPUT .COAL	1.00	0.10	0.10	1.00		
INPUT .LUMBER	0.50	0.40	0.30	0.30		
INPUT .HOUSBOP				0.40		
INPUT .CAPBOP	0.50	1.50	1.50	0.10	0.10	
INPUT .LABOR	0.40	0.20	0.20	0.02	0.40	
	+	DOM6	DOM7	DOM8	DOM9	DOM10

OUTPUT.HOUSEOP	0.80					
OUTPUT.CAPEOP	1.10	6.00	1.80	1.20	0.40	
OUTPUT.STEEL			2.00			
OUTPUT.COAL				2.00		
OUTPUT.LUMBER					1.00	
INPUT .TEXTILES	0.80	0.40	0.10	0.10	0.10	
INPUT .HSERV	0.40	1.80	1.60	0.80	0.20	
INPUT .STEEL	1.00	2.00		0.50	0.20	
INPUT .COAL		0.20	1.00		0.20	
INPUT .LUMBER	3.00	0.20	0.20	0.50		
INPUT .CAPBOP	1.50	2.50	2.50	1.50	0.50	
INPUT .LABOR	0.30	0.10	0.10	0.40	0.40	
	+	DOM11	DOM12	IMP1	IMP2	IMP3
OUTPUT.AGRIC				1.00		
OUTPUT.FOOD					1.00	
OUTPUT.TEXTILES						1.00
OUTPUT.HOUSEOP		0.36				
OUTPUT.CAPEOP	0.90					
INPUT .HSERV			0.40	0.20	0.20	
INPUT .HOUSBOP		0.40				
INPUT .CAPBOP	1.00		0.20	0.10	0.10	
INPUT .LABOR			0.04	0.02	0.02	
INPUT .EXCHANGE			0.50	0.40	0.80	
	+	IMP4	IMP5	IMP6	IMP7	EXP1
OUTPUT.CAPEOP	1.00					
OUTPUT.STEEL		1.00				
OUTPUT.COAL			1.00			
OUTPUT.LUMBER				1.00		
OUTPUT.EXCHANGE					0.50	
INPUT .AGRIC					1.00	
INPUT .HSERV	0.40	0.40	0.40	0.40	0.20	
INPUT .CAPBOP	0.20	0.20	0.20	0.20	0.20	
INPUT .LABOR	0.04	0.04	0.04	0.04	0.04	
INPUT .EXCHANGE	1.20	0.60	0.70	0.40		
	+	EXP2	EXP3	EXP4	EXP5	EXP6
OUTPUT.EXCHANGE	0.40	0.80	1.20	0.60	0.70	
INPUT .FOOD	1.00					
INPUT .TEXTILES		1.00				
INPUT .HSERV	0.20	0.20	0.40	0.40	0.40	
INPUT .CAPEOP			1.00			
INPUT .STEEL				1.00		

```

INPUT .COAL                                1.00
INPUT .CAPBOP          0.10  0.10  0.20  0.20  0.20
INPUT .LABOR           0.02  0.02  0.04  0.04  0.04

          +      EXP7

OUTPUT.EXCHANGE      0.40
INPUT .HSERV         0.40
INPUT .LUMBER        1.00
INPUT .CAPBOP        0.20
INPUT .LABOR         0.04 ;

```

\* DECLARE SETS, PARAMETERS AND EQUATIONS FOR THE SJM ALGORITHM:

```

SETS      ITER      SJM ITERATION INDEX /ITER1*ITER10/
          STATS     STATISTICS /DEV, CPU, ITER/

```

PARAMETER

```

THETA(H)      Budget shares for Negishi model
P(C)          Market prices
PO(C)         Initial prices (random starting point)
INCOME(H)     Endowment income
EXPEND(H)     Value of allocated demand
ITLOG         Iteration log;

```

SCALAR

```

DEV           Current deviation
CONTOL        Convergence tolerance (\%) / 0.01/,
LAMDA         Damping factor /1/;

```

ALIAS (C,CC), (H,HH);

PARAMETER

```

ALPHA(C,H)    Demand function share parameter,
IBAR(H)       Reference income associated with DBAR,
RHO(H)        Primal form elasticity exponent,
A(C,S)        Activity analysis matrix;

```

```

ALPHA(C,H) = DBAR(C,H) / SUM(CC, DBAR(CC,H));
IBAR(H) = SUM(C, DBAR(C,H));
RHO(H) = (ESUB(H) - 1)/ESUB(H);
A(C,S) = IODATA("OUTPUT",C,S) - IODATA("INPUT",C,S);

```

```

*=====
* PRIMAL MODEL:

```

VARIABLES            Y(S)                    ACTIVITY LEVELS  
                       D(C,H)                   CONSUMPTION LEVELS  
                       WELFARE                SOCIAL WELFARE INDEX;

POSITIVE VARIABLE Y;

EQUATIONS            MARKET(C)              MARKET CLEARANCE FOR PRIMAL MODEL  
                       NEGISHI                 MAXIMAND FOR PRIMAL PROBLEM;

MARKET(C)..

$$\text{SUM}(H, D(C, H)) = L = \text{SUM}(S, A(C, S) * Y(S)) + \text{SUM}(H, E(C, H));$$

NEGISHI..

$$\text{WELFARE} = E = \text{SUM}(H, \text{THETA}(H) * ($$

$$\begin{aligned} & ( (1/\text{RHO}(H)) * \\ & \text{LOG}( \text{SUM}(C\$ALPHA(C, H), \text{ALPHA}(C, H) * (D(C, H)/\text{DBAR}(C, H))^{**\text{RHO}(H)}) \\ & ) \$ (\text{ESUB}(H) \text{ NE } 1) + \end{aligned}$$

$$\begin{aligned} & ( \text{SUM}(C\$ALPHA(C, H), \text{ALPHA}(C, H) * \text{LOG}(D(C, H)/\text{DBAR}(C, H))) \\ & ) \$ (\text{ESUB}(H) \text{ EQ } 1) \\ & ) \ ); \end{aligned}$$

MODEL PRIMAL / MARKET, NEGISHI/;

\*=====

\* DUAL MODEL:

VARIABLES            PI(C)                    MARKET PRICE  
                       WELFARE                SOCIAL WELFARE INDEX;

POSITIVE VARIABLE PI;

EQUATIONS            PROFIT(S)              PROFIT CONDITION  
                       DAFERMOS                MAXIMAND FOR DUAL PROBLEM;

PROFIT(S)..

$$O = G = \text{SUM}(C, A(C, S) * \text{PI}(C));$$

DAFERMOS..

$$\text{WELFARE} = E = \text{SUM}(H, \text{THETA}(H) * ($$

$$\begin{aligned} & ( (1/(1-\text{ESUB}(H))) * \\ & \text{LOG}( \text{SUM}(C\$ALPHA(C, H), \text{ALPHA}(C, H) * \text{PI}(C)^{*(1-\text{ESUB}(H))}) \end{aligned}$$

```

)$(ESUB(H) NE 1)
+ ( SUM(C$ALPHA(C,H), ALPHA(C,H) * LOG(PI(C)) )
)$(ESUB(H) EQ 1)
)
- SUM(C, PI(C) * E(C,H)) );

```

```
MODEL DUAL / PROFIT, DAFERMOS/;
```

```
=====
```

```
* COMPLEMENTARITY MODEL:
```

```
VARIABLES
```

```
PI(C)           MARKET PRICES
Y(S)           ACTIVITY LEVELS
```

```
EQUATIONS
```

```
MCPMKT(C)      MARKET CLEARANCE (FOR THE MCP MODEL);
```

```
MCPMKT(C)..
```

```
SUM(S, Y(S) * A(C,S)) + SUM(H, E(C,H)) =G=
SUM(H, ( ALPHA(C,H) * INCOME(H) / PI(C)
)$(ESUB(H) EQ 1)
```

```
+ ( DBAR(C,H)*INCOME(H)*
SUM(CC$ALPHA(CC,H),
ALPHA(CC,H)*PI(CC)**(1-ESUB(H)))*(1/PI(C))**ESUB(H)
)$(ESUB(H) NE 1) );
```

```
MODEL MCP / PROFIT.Y, MCPMKT.PI/;
```

```
=====
```

```
* Randomly chosen starting point:
```

```
OPTION SEED=1001;
PO(C) = UNIFORM(0,2);
```

```
=====
```

```
* Run the primal SJM algorithm:
```

```
P(N) = 1;
P(C) = PO(C);
```

```
THETA(H) = SUM(C, P(C) * E(C,H));
INCOME(H) = THETA(H);
```

```

D.L(C,H) =
  ( ALPHA(C,H) * INCOME(H) / P(C) )$(ESUB(H) EQ 1) +
  ( DBAR(C,H)*INCOME(H) *
    SUM(CC$ALPHA(CC,H), ALPHA(CC,H)*P(CC)**(1-ESUB(H)))
    * (1/P(C))**ESUB(H) )$(ESUB(H) NE 1);

D.LO(C,H)$ALPHA(C,H) = 1.E-5;
D.FX(C,H)$ALPHA(C,H) EQ 0) = 0;

DEV = +INF;

LOOP(ITER$(DEV GT CONTROL),

      SOLVE PRIMAL USING NLP MAXIMIZING WELFARE;

      P(C) = MARKET.M(C);
      LOOP(N, P(C)$P(N) = P(C) / MARKET.M(N) );

      INCOME(H) = SUM(C, P(C) * E(C,H));
      EXPEND(H) = SUM(C, P(C) * D.L(C,H));
      DEV = 100 * SMAX(H$INCOME(H), ABS(INCOME(H)-EXPEND(H))
                    / INCOME(H));

      ITLOG("PRIMAL",ITER,"DEV") = DEV;
      ITLOG("PRIMAL",ITER,"ITER") = PRIMAL.ITERUSD;
      ITLOG("PRIMAL",ITER,"CPU") = PRIMAL.RESUSD;

      THETA(H) = LAMDA * INCOME(H) + (1 - LAMDA) * THETA(H);
);
ITLOG("PRIMAL","TOTAL","CPU")
= SUM(ITER, ITLOG("PRIMAL",ITER,"CPU"));
ITLOG("PRIMAL","TOTAL","ITER")
= SUM(ITER, ITLOG("PRIMAL",ITER,"ITER"));

*=====

* Run the dual SJM algorithm from the same starting point --
* this should generate an identical sequence:

P(N) = 1;
P(C) = P0(C);

THETA(H) = SUM(C, P(C) * E(C,H));
PI.L(C) = P(C);
PI.LO(C)$SMAX(H, ALPHA(C,H)) = 1.E-5;

DEV = +INF;

```

```

LOOP(ITER$(DEV GT CONTOL),

    SOLVE DUAL USING NLP MAXIMIZING WELFARE;

    P(C) = PI.L(C);
    LOOP(N, P(C)$P(N) = P(C) / PI.L(N); );

    INCOME(H) = SUM(C, P(C) * E(C,H));
    EXPEND(H) = THETA(H);
    LOOP(N, EXPEND(H)$P(N) = THETA(H) / PI.L(N); );
    DEV = 100 * SMAX(H$INCOME(H), ABS(INCOME(H)-EXPEND(H))
        / INCOME(H));

    ITLOG("DUAL",ITER,"DEV") = DEV;
    ITLOG("DUAL",ITER,"ITER") = DUAL.ITERUSD;
    ITLOG("DUAL",ITER,"CPU") = DUAL.RESUSD;

    THETA(H) = LAMDA * INCOME(H) + (1 - LAMDA) * THETA(H);
);

ITLOG("DUAL","TOTAL","CPU")
= SUM(ITER, ITLOG("DUAL",ITER,"CPU"));
ITLOG("DUAL","TOTAL","ITER")
= SUM(ITER, ITLOG("DUAL",ITER,"ITER"));

*====
* Run the complementarity-based SJM algorithm from the same
* starting point -- this should generate an identical sequence:

P(N) = 1;
P(C) = PO(C);
THETA(H) = SUM(C, P(C) * E(C,H));

INCOME(H) = THETA(H);
PI.L(C) = P(C);
PI.LO(C)$SMAX(H, ALPHA(C,H)) = 1.E-5;

DEV = +INF;
LOOP(ITER$(DEV GT CONTOL),

    SOLVE MCP USING MCP;

    P(C) = PI.L(C);
    LOOP(N, P(C)$P(N) = P(C) / PI.L(N); );

    INCOME(H) = SUM(C, P(C) * E(C,H));
    EXPEND(H) = THETA(H);

```

```
LOOP(N, EXPEND(H) = THETA(H) / PI.L(N); );
DEV = 100 * SMAX(H$INCOME(H), ABS(INCOME(H)-EXPEND(H))
              / INCOME(H));

ITLOG("MCP",ITER,"DEV") = DEV;
ITLOG("MCP",ITER,"ITER") = MCP.ITERUSD;
ITLOG("MCP",ITER,"CPU") = MCP.RESUSD;

THETA(H) = LAMDA * INCOME(H) + (1 - LAMDA) * THETA(H);
);

ITLOG("MCP","TOTAL","CPU")
= SUM(ITER, ITLOG("MCP",ITER,"CPU"));
ITLOG("MCP","TOTAL","ITER")
= SUM(ITER, ITLOG("MCP",ITER,"ITER"));

* Produce a summary report:

OPTION ITLOG:1:2:1;
DISPLAY ITLOG;
```

