

Moving from Static to Dynamic General Equilibrium Economic Models (Notes for a beginner in MPSGE)

Sergey Paltsev

Joint Program on the Science and Policy of Global Change
Massachusetts Institute of Technology

June 1999 (Revision April 2004)

Abstract

The document is intended to serve as a guide for beginners in MPSGE. It starts with a short introduction to the class of economic problems which can be solved with MPSGE, followed by a detailed description of step-by-step transformation of a simple static general equilibrium model into a dynamic Ramsey model. The model is based on a simplified dataset. Two cases are considered: the first dataset represents an economy on the steady-state growth path and the second dataset is off the steady-state growth path in a base year. The paper includes GAMS-MPSGE codes which can be copied and used as a starting point for further exploration of dynamic economic modelling.

Address for correspondence:

Sergey Paltsev
77, Massachusetts Ave., E40-429
Massachusetts Institute of Technology
Cambridge, MA 02139, USA

Phone: (617) 253-0514
Fax: (617) 253-9845
e-mail: paltsev@mit.edu
<http://web.mit.edu/paltsev/www>

1 Introduction

Originally, this document has been written as notes for myself during my struggle with understanding how to build a dynamic general equilibrium model based on Lau, Pahlke, and Rutherford (1997) paper¹. I subsequently discovered that the paper could be quite helpful to other novice modellers. The paper is organized around a simple two-sector, two-factor model. In the next section, I will try to share my understanding of what is MPSGE and what kind of problems can be solved with its help. Section 3 describes how a simple static model can be represented in the MPSGE format. If you are already familiar with MPSGE, you may skip that part and turn directly to section 4, which shows the transition of a static model into a dynamic model². This is done for the case where an economy is initially on a steady-state growth path. Section 5 presents adjustments in the model, which are necessary for the off steady-state case. Program listings are provided in the Appendices. I have also included Appendices on MPSGE-MCP conversion based on the example of a simple Shoven-Whalley (1984) model, and representation of different functional forms in MPSGE format. They are not directly relevant to dynamic models. However, they are important for better understanding the process of economic modelling with MPSGE.

2 MPSGE and General Economic Equilibrium Models

MPSGE (Mathematical Programming System for General Equilibrium Analysis) is a programming language designed by Thomas Rutherford (1999) in the early 80s for solving Arrow-Debreu economic equilibrium models. MPSGE uses the programming language GAMS (General Algebraic Modeling System) as an interface³. As a result, if one wants to use MPSGE, he needs to learn basic GAMS syntax (Brook et al, 2003) as well. As it follows from the name, MPSGE solves computable general equilibrium (CGE) economic models. These models consist of economic agents who interact among each other through prices which emerge from markets for goods and factors of production. The word “general” means that *all* economic flows are accounted for, i.e there is a “sink” for every “source”. Finding an equilibrium for a CGE model involves finding equilibrium prices, quantities, and incomes.

As an illustration of a simple economic model, consider the task of finding an equilibrium in the economy (for the future reference I will call it *Wonderland*), which consists of two economic agents: *consumers* and *producers*. Consumers have an initial endowment of labor L and capital K . For simplicity, there is a single representative consumer⁴ *CONS* in Wonderland. The consumer derives his income I from the sales of his endowments. Then he purchases his preferred choice of goods. There are two goods, X and Y , in the economy. The consumer obtains utility from consumption of the goods. Producers are the firms that take initial endowments of the consumer as inputs of production and convert them into outputs. Both production sectors, X and Y , are characterized by the available technology F and G , respectively.

We want to determine the prices and quantities which maximize producers’ profits and consumer’s utility⁵. The solution to this problem (which is a simple Arrow-Debreu (1954) problem) can be found using a non-linear formulation. It can be represented as an optimization problem of the consumer subject to income, technology, and feasibility constraints.

¹The previous version of the document and its Spanish translation kindly made by Juan Carlos Segura can be found (as of April 2004) at: <http://web.mit.edu/paltsev/www/current.html>

²The discussion in this paper is limited to a Ramsey model. Another approach to introducing dynamics is overlapping generations (OLG) models. For OLG formulation in MPSGE, see Rutherford (1998) and Rasmussen and Rutherford (2001).

³To learn more about GAMS, visit their homepage at <http://www.gams.com>

⁴Or, equivalently, a population of identical households.

⁵It may sound strange for non-economists that there is a single consumer who owns the firms and purchases the goods produced by these firms. But for economists with their “economic way of thinking” it seems natural. The significance of this assumption is that distributional considerations between different types of consumers are ignored.

$$\begin{aligned} \max W(X, Y) \quad \text{s.t.} \quad & p_x X + p_y Y = wL + rK \\ & X = F(K_x, L_x) \\ & Y = G(K_y, L_y) \\ & L = L_x + L_y \\ & K = K_x + K_y \end{aligned}$$

where W is a utility function; p_x and p_y are the prices of goods X and Y ; r is a price of capital K ; w is a price of labor L ; K_x and L_x are capital and labor used in the production sector X ; K_y and L_y are capital and labor used in the sector Y . This is a standard microeconomic textbook optimization problem, and a usual technique for finding the solution is the method of Lagrange multipliers. This problem can be solved in GAMS as a non-linear programming (NLP).

There are some cases (such as a presence of several consumers, taxes, or other distortions) where it is not possible to solve the problem of finding a market equilibrium as an optimization problem. Then the problem could be approached in a different way. It can be turned into a Mixed Complementarity Problem (MCP) and solved as a system of non-linear equations. NLP problems are a subset of MCP and MPSGE finds an equilibrium as a solution to MCP.

For the purposes of the models presented in this paper, complementarity really plays no role. Therefore, I just briefly describe my understanding of MCP. An interested reader is referred to Rutherford (1995) paper. I consider a nonlinear complementarity problem (NCP), which is also a special case of MCP. It can be written as:

$$\begin{aligned} \text{Given: } & f : R^n \rightarrow R^n \\ \text{Find: } & z \in R^n \\ \text{s.t. } & f(z) \geq 0, z \geq 0, z^T f(z) = 0. \end{aligned}$$

I have tried to convert the mathematical symbols into simple English, but it has not helped me very much. It sounds like: given the function f between two n -dimensional sets of real numbers (function f assigns to each member of the first set exactly one member of the second set), find z which belongs to an n -dimensional set of real numbers, such that function $f(z)$ is greater or equal to zero, z is greater or equal to zero, and associated complimentary slackness condition⁶ is satisfied. The complimentary slackness condition requires that either z equals zero (i.e. the dual multiplier vanishes), or the inequality constraint is satisfied with strict equality, or both.

However, I found out that these words is just a usual scientific way of hiding something rather simple. Do you recall from your 6th grade math the solution to an equation like: $x(5 - x) = 0$? Yes, it is $x = 0$ and $x = 5$. To make it more clear for a comparison, the equation above can be rewritten as $xf(x) = 0$, where $f(x) = 5 - x$. It leads to the condition that either x or $f(x)$ has to be equal zero. This is the main idea behind MCP.

In the case when we have not just one x , but a vector of $\bar{x} = (x_1, x_2, \dots, x_n)$, then there is a system of n equations like $\bar{x}f(\bar{x}) = 0$, which forms an MCP problem. The word ‘‘mixed’’ in MCP reflects the fact that the solution is a mix of equalities $f(x) = 0$ and inequalities $f(x) > 0$.

Mathiesen (1985) has shown that the Arrow-Debreu economic equilibrium model can be formulated as MCP, where three inequalities should be satisfied: zero profit condition, market clearance condition, and income balance condition. A set of three non-negative variables is involved in solving MCP problem: prices, quantities (they are called as *activity levels* in MPSGE), and income levels.

Zero profit condition requires that any activity operated at a positive intensity must earn zero profit (i.e. value of inputs must be equal or greater than value of outputs). Activity levels y for constant returns to scale production sectors are the associated variables with this condition. It means that either $y > 0$ (a positive amount of y is produced) and *profit* is zero, or *profit* is negative and $y = 0$ (no production activity takes place). In terms of MCP, the following condition

⁶An expression written as $x^T y = 0$ means $x_i y_i = 0$, for all $i = 1, \dots, n$. The variables x_i and y_i are called a *complementary pair* and are said to be *complements* to each other.

should be satisfied for every sector in an economy:

$$-profit \geq 0, y \geq 0, output^T(-profit) = 0$$

Market clearance condition requires that any good with positive price must have a balance between supply and demand and any good in excess supply must have a zero price. Price vector p (which includes prices of final goods, intermediate goods and factors of production) is associated variable. In terms of MCP, the following condition should be satisfied for every good and every factor of production:

$$supply - demand \geq 0, p \geq 0, p^T(supply - demand) = 0$$

Income balance condition requires that for each agent (including any government entities) value of income must equal the value of factor endowments. Is it possible to formulate it in terms of MCP:

$$(endowment - income) \geq 0, income \geq 0, income^T(endowment - income) = 0$$

However, income usually does not equal to zero and the condition $endowment - income = 0$ is satisfied as an equality. As such, income balance condition is definitional rather than complimentary.

To illustrate the equilibrium conditions, consider the Wonderland example. For the reasons which will be clear later, additional production sector W is introduced to represent utility derived from the consumption of X and Y . Accordingly, P_w is a price of the output of the production sector W . First, consider zero profit conditions. In the Wonderland example, an activity level y is a vector with the following components $y = (X, Y, W)$. Profit in a particular sector is defined as a difference between total revenue and total cost. We assume that activity levels X , Y , and W are positive (i.e., the condition $y \geq 0$ is satisfied with the strict inequality as $y > 0$). Hence, zero profit conditions ($-profit = 0$) can be written as

$$-(P_x - C_x(w, r)) = 0, \tag{1}$$

$$-(P_y - C_y(w, r)) = 0, \tag{2}$$

$$-(P_w - e(P_x, P_y)) = 0, \tag{3}$$

where C_x, C_y are unit cost functions for X and Y (cost of production of one unit of a good under the factor prices w and r); e is a unit cost (expenditure) function for W (cost of buying one unit of utility under the prices P_x and P_y).

Market clearance condition requires that if prices are positive then supply should be equal to demand. It is represented by the following equations

$$X = \frac{\partial e}{\partial P_x} W, \tag{4}$$

$$Y = \frac{\partial e}{\partial P_y} W, \tag{5}$$

$$W = \frac{I}{P_w}, \tag{6}$$

$$L = \frac{\partial C_x}{\partial w} X + \frac{\partial C_y}{\partial w} Y, \tag{7}$$

$$K = \frac{\partial C_x}{\partial r} X + \frac{\partial C_y}{\partial r} Y. \tag{8}$$

On the demand side, partial derivatives $\partial e/\partial P_i$, $\partial C_i/\partial w$, and $\partial C_i/\partial r$ represent compensated demand functions (by Shephard's lemma)⁸.

Income balance condition states that the value of endowments equals consumer's income:

$$wL + rK = I \quad (9)$$

MPSGE would solve nine equations (1-9) for nine unknowns: activity levels, X, Y, W ; prices, P_x, P_y, P_w, w, r ; and income I . It should be noted that equilibrium conditions are hard to formulate in an explicit form for certain problems. The good news for a MPSGE user is that one does not need to spend any time deriving the equilibrium conditions. MPSGE builds them for you automatically. However, it is always helpful to understand what the algebraic representation of a particular MPSGE model is⁹. For a comparison, alternative formulations of a simple model of taxation and international trade (Shoven and Whalley, 1984) in MPSGE and its algebraic form (MCP formulation) are presented in Appendix 2. An MCP formulation of a dynamic Ramsey problem can be found in the Rutherford's (1997) paper. It should be noted that MPSGE is a tool for the formulation but not for the solution of complementarity problems. In order to run MPSGE program, MCP solvers should be used. GAMS has two MCP solvers: MILES and PATH. In Tom's words, MILES is his abandoned child, which leaves PATH as the first choice.

3 A Static Model

In the previous section we have briefly discussed how MPSGE finds an equilibrium. It is nice to know that MPSGE will solve it not as an optimization problem, but instead by finding an equilibrium in a system of inequalities. However, in order to create a simple MPSGE program a user does not need to know this. What is much more important for a beginner is the syntax of the MPSGE language¹⁰. A big advantage of MPSGE is that it is based on nested constant elasticity of substitution (CES) utility and production functions¹¹. Use of nested functions can provide a flexible representation of how inputs are related. To introduce the MPSGE syntax¹², we will consider again the Wonderland example described in Section 2. The GAMS-MPSGE program of the Wonderland static model (`move1.1.gms`) is listed in Appendix 3.

Economic models are based on data which in turn are usually organized into a Social Accounting Matrix (SAM)¹³, which is also sometimes called as *microconsistent matrix* (MCM). We assume the following data represented in Table 1 where a SAM has three production sectors X, Y, and W (they are declared in the \$SECTORS block of the MPSGE program), one consumer CONS (declared in the \$CONSUMERS block), and five markets X, Y, W, L, and K (declared in the \$COMMODITIES block). Because the variables which are associated with commodities are their prices, they usually begin with the letter P.

⁷An ordinary (uncompensated) demand function depicts the change in the quantity of a commodity (or a factor) demanded when price changes. It includes both income and substitution effects. A compensated demand curve shows the quantity demanded changes when price changes *and* the level of utility (or the level of output for factor demands) is fixed. It represents the substitution effect only. For a profit-maximizing level of output, the compensated and uncompensated factor demands are the same values.

⁸For more information on Shephard's lemma see, for example, Varian (1992).

⁹When I saw an MPSGE code for the first time, I said: "I need to see an underlying algebra". Later on I have realized that MPSGE representation is much more attractive and intuitive than its algebraic formulation. So now the knowledgeable person always smiles when I am saying: "I don't want those ugly MCP equations, show me an MPSGE code".

¹⁰MPSGE syntax as of April 2004 can be found at <http://debreu.colorado.edu/mainpage/mpsge.htm>

¹¹See Appendix 1 for the examples of conversion of different functions into the MPSGE format.

¹²In this treatment of simple static and dynamic models, many MPSGE features are not going to be discussed. For additional practice, see the Markusen examples (Markusen and Rutherford, 1995), available as of April 2004 at: <http://debreu.colorado.edu/markusen/markusen.htm>

¹³For more information on converting input-output data into an MPSGE model, see Rutherford and Paltsev (1999).

Markets	Production Sectors			Consumers
	X	Y	W	CONS
PX	100		-100	
PY		100	-100	
PW			200	-200
PL	-40	-60		100
PK	-60	-40		100

Table 1. A Static Model SAM (the Wonderland example)

A SAM describes a data set, which is usually referred to as a *benchmark* dataset. Numbers represent the values (price times quantity) of economic transactions in a given period of time. A positive entry means a value of output (or sales). A negative entry reflects a value of input (or purchases). Zero profit, market clearance and income balance conditions imply that row sums and column sums are equal to zero.

To illustrate how SAM is balanced, consider the following row and column of Table 1. A production sector X purchases 40 units¹⁴ of labor and 60 units of capital, and sales 100 units of good X. The sum over X column is equal to zero. It shows that there is no excess profit (zero profit) in the sector X. A row PX shows that 100 units of X which are produced and sold (positive entry of 100) are consumed by sector W (negative entry of 100). The sum over PX row is equal to zero. It shows that supply equals to demand (market clearance) for a good X.

For simplicity, sometimes a Harberger convention is adopted in a benchmark SAM. It consists in normalizing all prices to 1. Then quantities in a SAM represent expenditures, or how much of a good or factor one can buy for \$1. It should be noted that an Arrow-Debreu economy only depend upon relative prices. Doubling all prices double money profits and income, which is resulted in the same solution for quantities (or activity levels). The absolute price level has no impact on the equilibrium outcome.

As it has already been noted, MPSGE does not require a user to formulate an algebraic representation of production and utility functions. The user only needs to provide reference quantities, reference prices, and elasticities. Based on provided information, MPSGE constructs underlying production and utility functions. At the solution, MPSGE returns equilibrium values for the variables, described in `$SECTORS`, `$COMMODITIES`, and `$CONSUMERS`.

In the Wonderland example, there are two products (X and Y), two factors (L and K), and one consumer (CONS). An extra row and column (PW and W) are introduced in order to explicitly represent utility derived from an aggregate consumption. As we will discuss later, it could have been done directly in the `$DEMAND` block¹⁵.

The program starts with declaration of the title, which will be printed in a solution listing. Usually, GAMS-formatted declarations of scalars, parameters, and sets then follow. We do not need them in our simple static model. The next portion is in MPSGE format.

```
$ONTEXT
```

```
$MODEL:NAME
```

These two blocks shifts control from GAMS to MPSGE compiler. Model name is chosen by a user. For example, our model has the name MOVE1.1. The model name has to be a legitimate filename, because a file NAME.GEN (MOVE1.1.GEN in our case) is going to be generated.

```
$SECTORS:
```

In this block, production sectors are declared. They determine how inputs are converted into outputs. The variables here are activity levels and they are associated with zero profit conditions (i.e. they make sure that a production sector earn zero profit). In the Wonderland model, we have

¹⁴Make sure to remember that units here represent the value, i.e. price times quantity.

¹⁵However, a separate block for utility is helpful for an introduction of a consumption tax and for welfare analysis.

three sectors: X, Y, and W. Comments can be made in the same line after ! mark.

\$COMMODITIES:

Commodities are declared in this block. The variables here are prices. They are associated with market clearance conditions (i.e. they make sure that supply is equal to demand). We have five commodities: two output prices PX, PY, two factor prices PK, PL, and price index PW for welfare (i.e. utility derived from an aggregate consumption).

\$CONSUMERS:

Consumers who supply factors and receive utility are described. The variable here is income from all sources. It is associated with income balance condition (i.e. it makes sure that total income is equal to total consumption (or total demand)). In our case, we have one representative consumer CONS.

\$PROD:

Production block constructs the underlying production function (i.e., the rule how inputs are converted into output). Sectors and commodities used in a production block should be declared in respective blocks. Consider the first production block for X (Production blocks for Y and W have a similar structure):

```
$PROD:X s:1
  O:PX    Q:100
  I:PL    Q:40
  I:PK    Q:60
```

This block describes a Cobb-Douglas production function (we can see it from `s:1`, which means that the elasticity of substitution between inputs is equal to one). Inputs here are PL and PK (we can see it from `I:PL` and `I:PK`). Output is PX (`O:PX`). This production sector converts 40 units of PL and 60 units of PK into 100 units of PX. Mathematically, the production function for this technology can be written as $X = \phi L^{0.4} K^{0.6}$ (See Appendix 1 for conversion of production and utility functions into the MPSGE format). Figure 1 illustrates the calibration of a production function to benchmark prices and quantities.

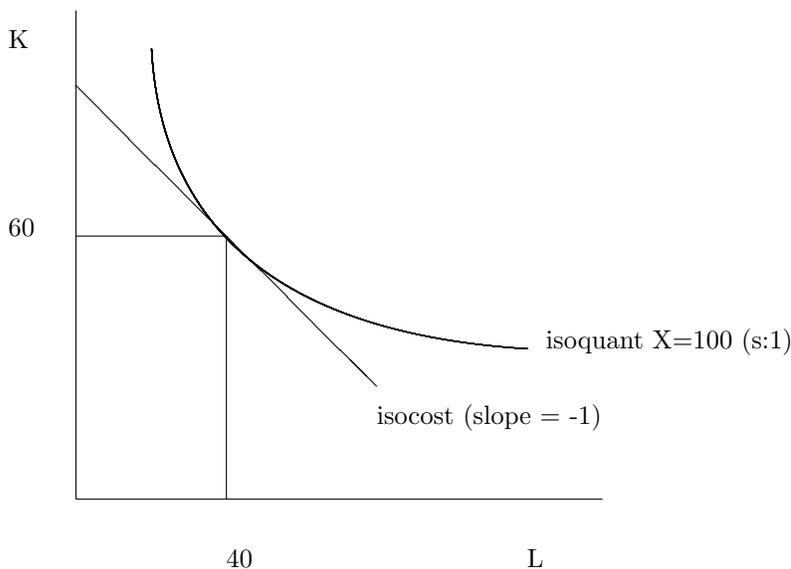


Fig.1. Production function calibration

MPSGE specifies the production function with a single reference point. In this example we have explicitly specified reference quantities only. Reference prices are set to 1 by default. Relative price of inputs determines the slope of isocost (which in turn equals to the slope of isoquant at the reference quantities). It means that our production block is identical to:

```
$PROD:X s:1
      O:PX      Q:100      P:1
      I:PL      Q:40       P:1
      I:PK      Q:60       P:1
```

The curvature of isoquant is determined by s , the elasticity of substitution between inputs ($s : 1$ corresponds to a Cobb-Douglas production function). Default value for elasticity is zero. The reference quantities and prices are used only for calibration. *They are not used by the solver as starting values of the variables.* Production block Y is similar to X.

Production block for W (Welfare) serves as a tool for conversion of goods X and Y consumption into utility derived from an aggregate consumption. In this simple model we have introduced this production block for clarity purposes only. We could have eliminated the production block W and changed the demand block into¹⁶

```
$DEMAND:CONS
      D:PX      Q:100
      D:PY      Q:100
      E:PL      Q:100
      E:PK      Q:100
```

However, in more complicated models the representation of welfare as a production block may be very useful (for example, in the analysis of a consumption tax).

```
$DEMAND:
```

Demand block represents an income balance constraint.

```
$DEMAND:CONS
      D:PW      Q:200
      E:PL      Q:100
      E:PK      Q:100
```

A consumer receives income from its L and K endowments (E:PK and E:PL) and demands good PW (D:PW). Reference quantities (Q fields) are used for calibration of the utility function in the same way that the production block calibrates production function.

```
$OFFTEXT
```

```
$$SYSINCLUDE mpsgeset NAME
```

These two blocks shift control from MPSGE back to GAMS.

```
NAME.ITERLIM = 0;
```

This command defines the maximum number of iterations which are going to be taken by the solver. When this is zero, we tell the solver not to solve the model but to return the values which are based on the initial data. Setting the iteration limit to zero is important to assure that the benchmark data (i.e., the data described in a SAM) represent the equilibrium solution.

```
$INCLUDE NAME.GEN
```

¹⁶See the code of the program `move1.2.gms` from Appendix 3. You can run it to compare the results with `move1.1.gms`.

SOLVE NAME USING MCP;

These two blocks are used for running the model.

The solution of the static model (reported in a listing (`move1_1.lst`) file) is:

	LOWER	LEVEL	UPPER	MARGINAL
VAR X	.	1.0000	+INF	.
VAR Y	.	1.0000	+INF	.
VAR W	.	1.0000	+INF	.
VAR PX	.	1.0000	+INF	.
VAR PY	.	1.0000	+INF	.
VAR PL	.	1.0000	+INF	.
VAR PK	.	1.0000	+INF	.
VAR PW	.	1.0000	+INF	.
VAR CONS	.	200.0000	+INF	.

In the solution, LOWER and UPPER columns show the bounds of respective variables. Zero is represented by “.”. INF means infinity. The LEVEL column reports the solution value. The MARGINAL column shows the value of the complimentary slackness (shadow) variable. We need to pay close attention to LEVEL and MARGINAL columns. Complimentary slackness implies that in equilibrium, either the level of a variable will be positive or the marginal value will be positive, but not both. If both of them are positive, we need to check our model.

The solution to the static model tells us that the benchmark data (represented in the SAM) are consistent with the equilibrium of the model. We can see it because all marginals are equal to zero after benchmark replication (i.e., after solving the model with ITERLIM=0). Equilibrium activity levels X, Y, and Z are equal to 1 (*and not to 100, 100, and 200 as in SAM !*). Equilibrium prices are equal to 1. Equilibrium income level of a representative consumer is equal to 200.

Somewhat confusing for beginners is that activity levels in the solution are all equal to one. This is just due to rescaling and it is done for the purpose of future analysis of counterfactual experiments, where a user can easily see the percentage change of a value deviated from 1. If we want to see unscaled activity levels, we need to multiply the levels from the solution by Q: fields in the production blocks. For example, for the activity level X it equals to $1 \cdot 100 = 100$.

There are other ways to get actual activity levels in the solution listing. One possible way is to rescale production blocks and assign initial values for activities as it is done in the program `move1_3.gms`. The solution for this case directly replicates the benchmark data:

	LOWER	LEVEL	UPPER	MARGINAL
VAR X	.	100.0000	+INF	.
VAR Y	.	100.0000	+INF	.
VAR W	.	200.0000	+INF	.
VAR PX	.	1.0000	+INF	.
VAR PY	.	1.0000	+INF	.
VAR PL	.	1.0000	+INF	.
VAR PK	.	1.0000	+INF	.
VAR PW	.	1.0000	+INF	.
VAR CONS	.	200.0000	+INF	.

However, for numerical reasons, it is advisable to scale equilibrium values close to unity. That is why it is better to use a REPORT block as it is shown in the program `move1_4.gms`, where the initial SAM is recreated using the reported values.

All initial values for activity levels and prices are equal to 1 by default. A variable CONS, which represents a consumer income level, is somewhat different. It is determined by endowments. *Remember from income balance condition that income is equal to total value of endowments, and in turn equal to total demand.* In the case of static model it is equal to 200 because endowment of capital (E:PK) is 100 and endowment of labor (E:PL) is 100, and prices PK and PL are equal to 1 by default.

As we have noted, the model determines only relative prices. So if we assign all initial prices as being equal to 2, then variable CONS will be calculated to be equal to 400. We can check it by modifying the static model. Initial price levels can be assigned after `$SYSINCLUDE mpsgeset NAME`, but before `$INCLUDE NAME.GEN` command in the following way:

```
PX.L = 2;
PY.L = 2;
PW.L = 2;
PK.L = 2;
PL.L = 2;
```

Then the solution listing shows an increase in all prices and the income level.

	LOWER	LEVEL	UPPER	MARGINAL
VAR X	.	1.0000	+INF	.
VAR Y	.	1.0000	+INF	.
VAR W	.	1.0000	+INF	.
VAR PX	.	2.0000	+INF	.
VAR PY	.	2.0000	+INF	.
VAR PL	.	2.0000	+INF	.
VAR PK	.	2.0000	+INF	.
VAR PW	.	2.0000	+INF	.
VAR CONS	.	400.0000	+INF	.

Due to determination of relative prices only, usually one price is fixed by a user. If no price is fixed initially (i.e. no good is chosen as a numeraire), then MPSGE will do a default price normalization. In the case of our static model, a consumer's income level CONS is fixed. It can be seen in the solution listing file (.lst) by looking at the statement **Default price normalization using income for CONS.**

The model can also be specified in a different format. Instead of using numbers in reference quantity and price field, we could have introduced parameters at the beginning of the program and used them for calibration in the fields of production and demand blocks. An example of the vector syntax is presented in the program `move1_5.gms`.

After benchmark calibration, counterfactual experiments (for example, an introduction of taxes or changes in labor endowment) are performed. It is important to remember to change the iteration limit before running counterfactuals, which is done with the GAMS statement like `NAME.ITERLIM=2000;`. The solver will stop as soon as this limit is reached if the solution is not found before a user defined number of iterations. In this paper, we will not describe any counterfactuals with our static model¹⁷. Our interest here is different and consists of the conversion of the static model into a dynamic one.

4 Dynamic models

When I had started to learn economic modeling, I immediately wanted to build a dynamic model. I wanted to predict the future, something which struck me as patently impossible with a static model.

¹⁷See Markusen examples for practice (Markusen and Rutherford, 1995).

Later on, I realized that my attempts of dynamic modeling were akin to building a spaceship with a screwdriver. Besides, I wanted too much out of dynamic models. Dynamic models are nowhere near the goal of predicting the future. A model can more or less tell you what is going to happen if there are no shocks and structural changes in an economy. In addition, one need to make assumptions about the rate of economic growth for several decades into the future, the rate of time preference, the growth rate of population, inflation, depreciation, etc. All these necessary assumptions bring us very far from the reality. But policy makers still need to make their decisions and economists need to provide answers about the future. Therefore, dynamic general equilibrium models are important tools for economic policy evaluation. But now I know that a good static model is much more useful than a bad dynamic one; and good dynamic models are quite rare.

With such an encouragement, we will attempt to develop a dynamic model based on the Wonderland example. There are different approaches to dynamic economic modeling (see Ginsburgh and Keyzer (1997) for an overview). We limit our discussion to a simple Ramsey model. The model behaves differently if it is on a so called *steady-state* than it is not. A steady-state is defined as a situation in which the various quantities (such as capital, output, consumption, etc.) grow at constant rates. We start our consideration with the situation when benchmark data describe an economy on a steady-state in a base period.

4.1 An economy on a steady-state in a base period

In a dynamic version of the Wonderland example, a representative consumer maximizes the present value of his lifetime utility

$$\max \sum_{t=0}^{\infty} \left(\frac{1}{1+\rho} \right)^t U(c_t), \quad (10)$$

where t - time periods, ρ - individual time-preference parameter, U - utility function, c_t - consumption in period t . The consumer faces the following constraints. First, total output produced in the economy is divided to consumption and investment, I_t . Second, capital depreciates at the rate δ . This constraints can be written as

$$c_t = F(K_t, L_t) - I_t, \quad (11)$$

$$K_{t+1} = K_t(1 - \delta) + I_t, \quad (12)$$

where K is capital and F represents production function.

To find the first-order conditions, consider two periods: t and $t + 1$. A representative agent makes his decision not only about his consumption today, but rather about consumption today and investment today (which will determine his capital and consumption tomorrow). So there are three decision variables: c_t , K_t , and I_t (It means that we need to consider two constraints on capital, one constraint is for K_t , and the other constraint is for K_{t+1} .) Therefore, the Lagrangian is:

$$\mathfrak{S} = \left(\frac{1}{1+\rho} \right)^t U(c_t) + \lambda_1 (F(K_t, L_t) - I_t - c_t) + \lambda_2 (K_{t-1}(1-\delta) + I_{t-1} - K_t) + \lambda_3 (K_t(1-\delta) + I_t - K_{t+1}) \quad (13)$$

We have three Lagrange multipliers, λ_1 is a multiplier on a consumption constraint, λ_2 is a multiplier on a “capital today” constraint, and λ_3 is a multiplier on a “capital tomorrow” constraint. Solving the utility maximization problem results in the following first-order conditions:

$$\frac{\partial \mathfrak{S}}{\partial c_t} = \left(\frac{1}{1+\rho} \right)^t \frac{\partial U(c_t)}{\partial c_t} - \lambda_1 = 0 \quad (14)$$

$$\frac{\partial \mathfrak{S}}{\partial K_t} = \lambda_1 \left(\frac{\partial F}{\partial K_t} \right) - \lambda_2 + \lambda_3(1 - \delta) = 0 \quad (15)$$

$$\frac{\partial \mathfrak{S}}{\partial I_t} = -\lambda_1 + \lambda_3 = 0 \quad (16)$$

Now recall that the Lagrange multiplier shows the sensitivity to changes in the constraint. In economic terms it is a measure of value of the scarce resources in the problem under consideration. In a usual utility maximization problem (maximization of utility subject to a budget constraint), the Lagrange multiplier can be interpreted as the marginal utility of (budget) money. In a usual cost minimization problem the Lagrange multiplier can be interpreted as a marginal cost of production (or the internal value, or imputed value, or more frequently, the shadow price). Mathematically, all these problems are identical. What is different is our interpretation. So in our Lagrange optimization the multiplier shows the shadow price (or marginal cost). Under constant returns to scale and perfect competition assumptions, price equals marginal cost. Therefore, we can rewrite the first-order conditions as:

$$P_t = \left(\frac{1}{1+\rho}\right)^t \frac{\partial U(c_t)}{\partial c_t}, \quad (17)$$

$$PK_t = (1-\delta)PK_{t+1} + P_t \frac{\partial F(K_t, L_t)}{\partial K_t}, \quad (18)$$

$$P_t = PK_{t+1}, \quad (19)$$

where P_t , PK_t , and PK_{t+1} are the values of the corresponding Lagrange multipliers. They can be interpreted as the price of output, the price of capital today, and the price of capital tomorrow, respectively. The utility maximization problem above is formulated as an NLP problem.

As has already been discussed, MPSGE solves it as an MCP problem. Let RK_t and W_t represent rental rate of capital and wage rate in period t . Denote a unit cost function as $C(RK_t, W_t)$ (Unit cost function is a solution of the problem $\min (W_t L_t + RK_t K_t)$ s.t. $F(K_t, L_t) = 1$), and a demand function as $D(P_t, M)$ (Demand function is a solution of the problem $\max \sum (1/(1+\rho))^t U(c_t)$ s.t. $\sum P_t c_t = M$), where M is consumer's income. Then MCP can be formulated as follows¹⁸.

Zero profit conditions:

$$P_t \geq PK_{t+1}, I_t \geq 0, I_t(P_t - PK_{t+1}) = 0 \quad (20)$$

$$PK_t \geq RK_t + (1-\delta)PK_{t+1}, K_t \geq 0, K_t(PK_t - RK_t + (1-\delta)PK_{t+1}) = 0 \quad (21)$$

$$C(RK_t, W_t) \geq P_t, Y_t \geq 0, Y_t(C(RK_t, W_t) - P_t) = 0 \quad (22)$$

Market clearance conditions:

$$Y_t \geq D(P_t, M) + I_t, P_t \geq 0, P_t(Y_t - D(P_t, M) + I_t) = 0 \quad (23)$$

$$L_t \geq Y_t \frac{\partial C(RK_t, W_t)}{\partial W_t}, W_t \geq 0, W_t(L_t - Y_t \frac{\partial C(RK_t, W_t)}{\partial W_t}) = 0 \quad (24)$$

$$K_t \geq Y_t \frac{\partial C(RK_t, W_t)}{\partial RK_t}, RK_t \geq 0, RK_t(K_t - Y_t \frac{\partial C(RK_t, W_t)}{\partial RK_t}) = 0 \quad (25)$$

Income balance condition:

$$M = PK_0 K_0 + \sum_{t=0}^{\infty} W_t L_t, M > 0 \quad (26)$$

¹⁸There are a number of equations in the model which can never be slack, simply because demand functions go to infinity as prices go to zero. We list all conditions here for consistency with our discussion of MCP formulation in Section 2.

As noted before, a modeller does not need to program these equilibrium conditions explicitly. MPSGE constructs them automatically. However, we need to describe a change in capital over time. This requires modification of the SAM, presented in Table 1. We add one more production sector for investment. This, in turn, will change the composition of the welfare block W, because now output can be consumed or invested, rather than just consumed as in the static case.

For the Wonderland example, a steady-state requires that investment is equal to 70. How did we come to this number? In order to quantify the value for investment on a steady-state growth path, we need to describe the evolution of capital and labor over time. It requires assumptions on the growth rate of labor, g , capital depreciation rate, δ , and interest rate, r . Denoting the initial labor force as L_0 , labor at time t is

$$L_t = L_0(1 + g)^t, \quad (27)$$

or, equivalently,

$$L_t = (1 + g)L_{t-1}. \quad (28)$$

Evolution of capital is given by equation (12). Note that if in a base period an economy is on a steady-state growth path, then all quantities (capital, labor, output, consumption) grow at the same constant rate g (off steady-state case is considered in the next section). As such, an equation for capital growth can be represented as

$$K_{t+1} = (1 + g)K_t. \quad (29)$$

In addition, we assume a constant interest rate r , so all future prices (including price of labor and capital) in terms of present value are:

$$P_{t+1} = \frac{P_t}{1 + r}. \quad (30)$$

Capital can be bought or rented. Therefore, implementation of dynamics involves two prices for capital: purchase price, PK , and rental price, RK . It should be stressed that in a static SAM (Table 1), the total value of capital endowment, VK (Check the intersection of PK and $CONS$ in Table 1 to see that $VK = 100$), equals to capital earnings (*and not to capital stock!*). The relationship between VK , rental price RK and a stock of capital K is as follows:

$$VK_t = K_t \cdot RK_t. \quad (31)$$

Now we need to consider the first-order conditions for capital and investment. Let's rewrite them as

$$PK_t = (1 - \delta)PK_{t+1} + RK_t \quad (32)$$

and

$$PK_{t+1} = P_t. \quad (33)$$

Equation (33) can be rearranged using equation (30) for PK:

$$PK_t = (1 + r)P_t. \quad (34)$$

Substitution of equation (34) for PK_t and equation (33) for PK_{t+1} into (32) leads to:

$$(1 + r)P_t = (1 - \delta)P_t + RK_t. \quad (35)$$

As such, an equation for the rental price is¹⁹:

$$RK_t = (\delta + r)P_t \quad (36)$$

¹⁹Setting $P(t) = 1$, an equation for RK makes perfect economic sense, because if capital and other loans are perfect substitutes, then households can also receive interest rate r on lending to other households. But capital depreciates at the rate δ . Then $r = RK - \delta$, or $RK = \delta + r$, which is the same as equation (36).

From equations (12) and (29), we have the following rule for investment on a steady-state

$$I_t = (\delta + g)K_t, \quad (37)$$

or using (31) and (36), investment in a base period is equal to

$$I_0 = \frac{(\delta + g)VK_0}{\delta + r}. \quad (38)$$

Equation (38) describes the value of investment which we need to introduce into the SAM for a steady-state growth path. We assume²⁰ $\delta = 0.05$, $g = 0.02$, and $r = 0.05$. Then, investment is $I = (0.05 + 0.02) \cdot 100 / (0.05 + 0.05) = 70$. We assume that goods are invested in the same proportion (35:35). In addition, we need to modify the welfare block W by dividing it into consumption and savings. The total amount of produced goods is 200, 70 of which is invested. It leaves 130 units for consumption in a base period, 65 of which is consumption of good X and another 65 is consumption of good Y. The benchmark SAM after these changes is represented in Table 2.

Markets	Production Sectors				Consumers
	X	Y	W	inv	CONS
PX	100		-65	-35	
PY		100	-65	-35	
PW			200		-200
PL	-40	-60			100
PK	-60	-40			100
sav			-70	70	

Table 2. Dynamic Model SAM

An important characteristic of the dynamic problem is a treatment of capital in the last period of modeling. We cannot solve numerically for an infinite number of periods, hence, some adjustments are needed for approximation of finite horizon model to the infinite horizon choices. Special procedures should be introduced for the terminal capital, otherwise, all capital would be consumed in the last period and nothing would be invested. The infinite horizon consumer problem can be partitioned at year T with a single state variable K_{T+1} , linking the two problems. This means that if you set the terminal capital stock to the appropriate value, then the time path of consumer demands will be identical to the infinite horizon values²¹.

Here we have followed the Rutherford's (1997) paper and introduced the level of post-terminal capital as a variable and added a constraint on the growth rate of investment in the terminal period:

$$\frac{I_T}{I_{T-1}} = \frac{Y_T}{Y_{T-1}}, \quad (39)$$

where T is a terminal period. The advantage of the usage of this constraint is that it imposes balanced growth in the terminal period but does not require that the model achieve steady-state growth. The meaning of the constraint is that investment in a terminal period should grow at the same rate as output²².

²⁰It should be noted that the transversality condition requires that $r > g$, otherwise, there is a possibility of a chain-letter borrowing (for detailed discussion see, for example, Barro and Sala-i-Martin (1995)).

²¹A modeller also should be aware of the fact that approximation of infinite horizon welfare impacts requires an extra weight on the terminal consumption level. Otherwise, taxation, for example, may incorrectly lead to welfare-improving results from a dynamic model.

²²Because all quantities grow at the same rate, it is possible to use a growth of total output, output in a certain production sector, or consumption (or simply $I+g$ in a steady-state case) as a right-hand side variable of equation (39).

A program listing of the dynamic model is represented in Appendix 3 (file `move2.1.gms`). Below we list and then discuss the changes which are necessary to make in the code of the static model.

1. Introduce time sets;
2. Declare the assumed interest rate, growth rate and depreciation rate;
3. Declare four more scalars: rental rate of capital in a base period, RK_0 , initial capital stock, K_0 , initial investment, I_0 , and initial capital earnings, VK ;
4. Declare two parameters which represent the growth rate of quantities, $QREF(T)$ (i.e., an index of economic activity over the model horizon in the steady-state equilibrium), and the growth rate of prices, $PREF(T)$ (i.e., an index of present value prices over the model horizon in a steady-state equilibrium);
5. Introduce two more production blocks: capital accumulation, K , and investment, I ;
6. Change the reference quantities in the `$PROD:W` block: 130, 65, 65 instead of 200, 100, 100 to represent an adjustment for the level of consumption in a base period;
7. Change the reference quantities in the `$DEMAND` block: base year consumption is 130 instead of 200 and baseyear capital stock is K_0 instead of 100 (*which was a representation of capital earnings, VK*); adjust labor endowment and consumption for the growth rates $QREF$ and $PREF$;
8. Introduce terminal capital in `$AUXILIARY`, `$CONSTRAINT` and `$DEMAND` blocks;
9. Make initial values assignment.

Introduction of time sets.

Introduction of time sets requires three sets T , $TFIRST$, and $TLAST$, GAMS code for which is

```
SET      T /1*10/, TFIRST(T), TLAST(T);
TFIRST(T) = YES$(ORD(T) EQ 1);
TLAST(T) = YES$(ORD(T) EQ CARD(T));
```

In the first line we declare the sets. We have 10 periods. Because of the importance of the base year period $TFIRST$ and the terminal period $TLAST$, they are declared as separate sets (and subsets of T). The next two lines assign the values for the first and last periods. It is done in a way which may seem a bit odd for a novice in GAMS. The purpose of this is convenience; because it allows you to change the time dimension of the model in a single line. For example, extension of the model to 100 periods requires only one change to `T /1*100/`.

Declaration of the assumed parameters.

We assume $g = 0.02$, $\delta = 0.05$, and $r = 0.05$, which is declared in the following GAMS statement.

```
SCALAR DELTA DISCOUNT FACTOR /0.05/
        R INTEREST RATE /0.05/
        G GROWTH RATE /0.02/;
```

Declaration of initial parameters for capital and investment.

We directly assign a value for initial capital earnings, VK , as

```
SCALAR VK INITIAL CAPITAL EARNINGS /100.0/;
```

Also, from equations (36), (31), and (37) the values for RK_0 , K_0 and I_0 are

```

RKO = DELTA+R;
KO = VK/RKO;
IO = (DELTA + G) * KO;

```

Declaration of growth rates for prices and quantities.

This step implements equations (29) and (30) for the growth rates in quantities and prices, which are

```

QREF(T) = (1 + G) ** (ORD(T) - 1);
PREF(T) = (1/(1 + R)) ** (ORD(T) - 1);

```

We have $\text{ORD}(T)-1$ as an exponent to represent the fact that in the base year QREF and PREF are equal to 1 and grow thereafter.

Production blocks for capital and investment.

New production block for $K(T)$ implements capital accumulation as it represented by equation (32). The MPSGE format of the production block is (we skip terminal capital for a moment):

```

$PROD:K(T)
      O:PK(T+1)      Q:(KO*(1-DELTA))
      O:RK(T)        Q:(KO*(R+DELTA))
      I:PK(T)        Q:KO

```

Those who learn the MPSGE representation of the Ramsey dynamic model are mostly confused by the reference quantity of $RK(T)$. The difference with equation (32) is that in the production block the reference quantity for capital rental price²³ is $KO*(R+DELTA)$, whereas in equation (32) $(r + \delta)$ does not appear as a coefficient of RK_t . The dynamic model would not calibrate without this reference quantity. The reason for this is the fact that RK is used in several production blocks: X , Y , and K and we need to reflect the relationship between RK_t and P_t given by equation (36)

Alternatively, we could have used KO as a reference price for $O:RK(T)$ and assign the rental price growth as $RK.L(T) = (DELTA+R)*PREF(T)$; , but in this case the input fields for production of X and Y blocks should also be changed as represented in the program `move2.2.gms` of Appendix 3.

Production block I introduces equation (33). We assume that investment is split equally between two production sectors.

```

$PROD:I(T) S:0
      O:PK(T+1)      Q:IO
      I:PY(T)        Q:(0.5*IO)
      I:PX(T)        Q:(0.5*IO)

```

Note that $S:0$ corresponds to Leontief production function, i.e., elasticity of substitution between inputs is equal to zero. If no elasticity is specified in a production block, MPSGE uses zero elasticity as a default value. Therefore, $S:0$ can be skipped from the code of this block.

Change in the Welfare block.

We need to make changes in the production of W , because in the dynamic version of the model not all output is consumed but some portion of it is invested. From Table 2 we know the new values for output (Q:130) and inputs (Q:65).

Change in the Demand block.

For the same reason, in the DEMAND block we need to modify a reference quantity for $PW(T)$ from 200 to 130. Also, growth in consumption and labor over time is represented by QREF(T).

²³Note that $KO*(R+DELTA)=VK$, so we could have used $Q:VK$ as a reference quantity for $O:RK(T)$.

We use the reference price $PREF(T)$ as the value of the discount on the future consumption given by equation (10). Introduction of individual time-preference as a separate parameter is not needed because in the Ramsey model, the steady-state leads to a constant pattern of per capita consumption and to the condition that $\rho = r$.

Another change in the DEMAND block involves the usage of initial capital stock $K0$ as a reference quantity for $PK(TFIRST)$, instead of the value of capital VK as in the static case. This is determined by income balance condition of the dynamic model.

One more modification in the DEMAND block is the introduction of intertemporal elasticity of substitution $s:1$. Note that we have assumed a Cobb-Douglas case in our simple dynamic model. In general, we could have introduced a scalar $SIGMA$ and use $s:SIGMA$ as an elasticity parameter (see Appendix 1 for conversion utility functions into the MPSGE format).

Terminal capital constraint.

In order to introduce a constraint in MPSGE, we need to use two types of blocks, which we have not discussed yet. A constraint is declared in \$AUXILIARY block and then formulated in \$CONSTRAINT block. For our model, we need to introduce a constraint on terminal capital given by equation (39). Therefore, a constraint TK is declared. Also, in order to induce investment in the last period of modeling, a price PKT is introduced. It represents the market for a post-terminal capital.

The post-terminal capital is output of the production blocks K and I . The condition $TLAST(T)$ shows that the post-terminal capital is going to be produced only in the last period. In CGE models, everything which is produced has to be consumed, therefore, an endowment of terminal capital in the \$DEMAND block is introduced as

$$E:PKT \quad Q:(-1) \quad R:TK$$

Negative value of the endowment represent the fact that terminal capital is going to be consumed. The field $R:TK$ implies that the following condition has to be satisfied:

$$SUM(T$TLAST(T), I(T)/I(T-1) - Y(T)/Y(T-1)) =G= 0;$$

Summation sign, SUM , is used due to the GAMS syntax which does not directly allow the assignment of a set member to a variable with a different domain.

Initial values.

The assignment is done after the \$SYSINCLUDE line but before the \$INCLUDE line of the program. The initial values in our model are assigned to reflect present values (prices are adjusted for $PREF(T)$ which is given by equation (30)) and growth in quantities (quantities are adjusted for $QREF(T)$ as represented by equation (29)). The only one exception is $PK.L(T)=(1+R)*PREF(T)$, which reflects equation (34).

The initial values for the level and price of a terminal capital are determined by the same logic as for other quantities and prices as²⁴:

$$\begin{aligned} TK.L &= K0 * (1 + G) ** CARD(T); \\ PKT.L &= SUM(TLAST, PK.L(TLAST)/(1+R)); \end{aligned}$$

The solution listing (run the program `move2.1.gms` to obtain the listing) shows that all marginals are equal to zero, activity levels are rising over time and prices are falling over time. Total consumption has the value of 1145. The price of capital differs from all other prices by $(1+r)$. Table 3 summarizes the results²⁵. After benchmark replication, one can run counterfactual experiments, such as introduction of taxes, change in growth rates, elasticities, etc.

²⁴An alternative formulation is $TK.L = SUM(TLAST, K.L(TLAST))$; and $PKT.L = SUM(TLAST, PREF(TLAST))$;

²⁵It could be a good practice for a novice in MPSGE to reproduce the results presented in Table 3

	Quantities	Prices	X	Y	W	K	I
1	1.0000	1.0000	100.00	100.00	130.00	1000.00	70.00
2	1.0200	0.9524	102.00	102.00	132.60	1020.00	71.40
3	1.0404	0.9070	104.04	104.04	135.25	1040.40	72.83
4	1.0612	0.8638	106.12	106.12	137.96	1061.21	74.29
5	1.0824	0.8227	108.24	108.24	140.72	1082.43	75.77
6	1.1041	0.7835	110.41	110.41	143.53	1104.08	77.29
7	1.1262	0.7462	112.62	112.62	146.40	1126.16	78.83
8	1.1487	0.7107	114.87	114.87	149.33	1148.69	80.41
9	1.1717	0.6768	117.17	117.17	152.32	1171.66	82.02
10	1.1951	0.6446	119.51	119.51	155.36	1195.09	83.66

Table 3. Steady-state prices and quantities

4.2 Calibrating dynamic models to a benchmark data which are not on a steady-state

Data for a base period may be inconsistent with a steady-state growth path. Table 4 gives an example of the situation where investment is “too big”, i.e., from equation (38) the value of investment is $80 > (0.05 + 0.02) \cdot 100 / (0.05 + 0.05) = 70$.

Markets	Production Sectors				Consumers
	X	Y	W	Inv	CONS
PX	100		-60	-40	
PY		100	-60	-40	
PW			200		-200
PL	-40	-60			100
PK	-60	-40			100
sav			-80	80	

Table 4. A Dynamic Model SAM off a Steady-State

One possible solution for model calibration in the case of the off-steady-state data is to assume that g is the labor growth rate, but the capital growth rate is different initially and converges to g over time²⁶. We use the calibration which is based on the Secant Method (for a description of the Secant Method see, for example Press et al (1992), or just search the Internet for “secant method”).

The method finds the solution to an equation $f(x) = 0$ as:

1. Initial guess on $x(n)$, where n is the number of iteration.
2. Calculate $x(n + 1)$ as²⁷:

$$x_{n+1} = x_n - \frac{f(x_n)}{f(x_n) - f(x_{n-1})} (x_n - x_{n-1}). \quad (40)$$

When the secant method converges it usually does so in 5 to 7 iterations if the initial guesses are good enough.

To reiterate our current dynamic problem, we know the initial values given in the SAM. The SAM is not consistent with a steady-state growth path. Also, we know (assumed) depreciation rate

²⁶Another way of calibrating to a static data set involves given data on a base year investment, interest rate, and return to capital. The long-run steady-state interest rate is determined based on assumed labor growth rate and calibrated utility discount rate.

²⁷Equation (40) could be derived from an equation of tangent: $\frac{y - y_n}{x - x_n} = f'(x_n)$. Put $y = 0$ and $y_n = f(x_n)$, then $x = x_n - \frac{f(x_n)}{f'(x_n)}$. Using $f'(x_n) = \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$ we receive equation (40).

and long-run interest rate but the rental price in a base period is unknown. Our goal is calibrating a dynamic model to the data represented in the SAM. In order to do that we use iterations based on the secant algorithm described above. The calibration involves the following iteration process: guessing initial rental rate of capital, RK_0 , calculating a model of the transition path, observing investment, I_0 , in the current equilibrium, and adjusting RK_0 .

The program listing for the off-steady-state dynamic model is provided in Appendix 3 (file `move3_1.gms`). At first, we guess about the rental price, RKG , in the first period. In the on-steady-state case the rental price (which is also equal to return to capital) is $\delta + r$. We make our guess using this knowledge. Note that if our guess is “too far”, the algorithm does not work. Also, if investment is “too far” from the steady-state value, the algorithm does not work as well.

The off-steady-state dynamic program has the same structure as before, except for the fact that we have used an alternative formulation similar to the program described in Appendix 3 (file `move2_2.gms`). We apply equation (40) for finding $f(x)$, which is in our case the difference between the value of investment calculated in the program and the one given in the SAM. Accordingly, $x(n)$ is the guess on rental price at the first period. During the first iteration we simply add 0.1 to our guess value for RK_0 in order to produce a basis for further iterations. During all other iterations we reproduce equation (40) as

```
RKG = RKG + ERROR(ITER)/(ERROR(ITER-1)-ERROR(ITER))
      * (RKOVAL(ITER)-RKOVAL(ITER-1));
```

where `ERROR` represents²⁸ $f(x)$ and `RKOVAL` represents $x(n)$.

The last portion of the program includes graphing the results for investment, consumption, growth rate and interest rate. We have used `GNUPLLOT` utility to produce graphs. It can be seen from the graphs (run the model `move3_1.gms` to obtain the graphs) that the values eventually converge to a steady-state growth path. The speed of convergence to a steady-state is determined by production and preference parameters (Barro and Sala-i-Martin, 1995).

5 Conclusion

Learn MPSGE!!!

References

- [1] Arrow, K.J., and G. Debreu (1954). “Existence of an Equilibrium for a Competitive Economy.” *Econometrica*, 22, 265-290.
- [2] Barro, R.J., and X. Sala-i-Martin (1995). *Economic Growth*, McGraw-Hill.
- [3] Brooke, A., D. Kendrick, A. Meeraus, and R. Raman (2003). *GAMS: A User’s Guide*, GAMS Deveopment Corporation. (available at <http://www.gams.com>)
- [4] Ginsburgh V., and M. Keyzer (1997). *The Structure of Applied General Equilibrium Models*, The MIT Press.
- [5] Lau M.I., A. Pahlke, and T.F. Rutherford (1997). “Modeling Economic Adjustment: A Primer in Dynamic General Equilibrium Analysis”, University of Colorado. Working Paper. (available at: <http://debreu.colorado.edu/primer/paper.htm>)
A version of the paper also appears as:

²⁸Note that for $f(x)$ we use an equation for consumption instead of investment in order to explicitly separate calculated and given values. Usage of `ERROR(ITER) = I.L("1")*I0 - I0`; instead of `ERROR(ITER) = C.L("1") - (200-I0)`; produces the same result.

- Lau M.I., A. Pahlke and T. F. Rutherford (2002). "Approximating infinite-horizon models in a complementarity format: a primer in dynamic general equilibrium analysis", *Journal of Economic Dynamics and Control*, 26, 577-609.
- [6] Markusen, J.R., and T.F. Rutherford (1995). The Markusen examples. Notes for GAMS workshop, 1995. (available at: <http://debreu.colorado.edu/markusen/markusen.htm>)
- [7] Mathiesen, L. (1985). "Computation of Economic Equilibrium by a Sequence of Linear Complementarity Problems", *Mathematical Programming Study*, 23, 144-162.
- [8] Press, W. H., B. Flannery, S. Teukolsky, and W. Vetterling (1992). "Secant Method, False Position Method, and Ridder's Method." 9.2 in *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, 2nd ed. Cambridge, England: Cambridge University Press, pp. 347-352.
- [9] Rasmussen, T., and T.F. Rutherford (2001). "Modeling Overlapping Generations in a Complementarity Format" University of Colorado, mimeo. (available at: <http://debreu.colorado.edu/olgmcp/default.htm>)
- [10] Rutherford, T.F. (1995). "Extension of GAMS for complementarity problems arising in applied economic analysis". *Journal of Economic Dynamics and Control*, 19(8), 1299-1324.
- [11] Rutherford, T.F. (1998). "Overlapping Generations with Pure Exchange: An MPSGE Formulation", University of Colorado, mimeo. (available at: <http://debreu.colorado.edu/olg/exchange/index.html>)
- [12] Rutherford, T.F. (1999). "Applied General Equilibrium Modeling with MPSGE as a GAMS Subsystem: An overview of the Modeling Framework and Syntax", *Computational Economics*, V.14, Nos. 1-2.
- [13] Rutherford, T.F., and S.V. Paltsev (1999). From an Input-Output Table to a General Equilibrium Model: Assessing the Excess Burden of Indirect Taxes in Russia. University of Colorado. Working Paper. (available at: <http://debreu.colorado.edu/papers/exburden.html>)
- [14] Rutherford, T.F., and S. Paltsev (2000). "GTAPinGAMS and GTAP-EG: Global Datasets for Economic Research and Illustrative Models", University of Colorado, Department of Economics Working Paper. (available at: <http://debreu.colorado.edu/papers/gtaptext.html>)
- [15] Rutherford, T.F. (2002). "Lecture Notes on Constant Elasticity Functions", University of Colorado, mimeo. (available at: <http://debreu.colorado.edu/ces.pdf>)
- [16] Shoven, J., and J. Whalley (1984). "Applied General-Equilibrium Models of Taxation and International Trade: An Introduction and Survey", *Journal of Economic Literature*, 22(3), 1007-1051.
- [17] Varian, H.R. (1992). *Microeconomic Analysis*, Third Edition, Norton & Company.

Appendix 1. Conversion of production and utility functions into the MPSGE format

MPSGE represents production and preferences as nested CES functions. This Appendix provides a very brief discussion of MPSGE production and consumption blocks. For more information about CES functions and their calibrated share form, see Rutherford (2002).

Production:

A production technology is identically represented by either the production function or corresponding cost function. A general form of a nested CES production function is $X(K, L) = (aK^\rho + (1-a)L^\rho)^{\frac{1}{1-\rho}}$ where ρ is the substitution parameter ($-\infty < \rho < 1$), which is related to the elasticity of substitution σ as $\sigma = \frac{1}{1-\rho}$. When $\rho = -\infty$ (or $\sigma = 0$), CES production function converts into Leontief (or fixed proportions) production function $X_1(K, L) = \min(aK_1, (1-a)L_1)$. When $\rho = 1$ (or $\sigma = \infty$), then inputs are perfect substitutes in production $X_2 = aK_2 + (1-a)L_2$. When $\rho = 0$ (or $\sigma = 1$), then CES production function converts into a Cobb-Douglas case $X_3(K, L) = K_3^a L_3^{1-a}$.

Let's focus on a Cobb-Douglas case $X(K, L) = Z K^a L^{1-a}$, where Z is a scale parameter. If we have data for output \bar{X} , capital \bar{K} , labor \bar{L} , capital price \bar{r} , and labor price \bar{w} , we can calibrate function coefficients in the following way:

$$a = \frac{\bar{r}\bar{K}}{\bar{r}\bar{K} + \bar{w}\bar{L}} \quad (41)$$

$$Z = \frac{\bar{X}}{\bar{K}^a \bar{L}^{1-a}} \quad (42)$$

MPSGE representation of a Cobb-Douglas production function is:

```
$PROD:X s:1
  O:PX   Q:Y
  I:PK   Q:K P:r
  I:PL   Q:L P:w
```

where $s : 1$ represents the elasticity of substitution for a Cobb-Douglas technology ($\sigma = 1$), whereas Y, K, L, r, w come from benchmark data used for calibration.

Consider MPSGE representation of a particular example of Cobb-Douglas production function: $X(K, L) = K^{0.6} L^{0.4}$. Based on equations (41) and (42), the following production block represents the technology:

```
$PROD:X s:1
  O:PX   Q:(0.6**0.6 * 0.4**0.4)
  I:PK   Q:0.6
  I:PL   Q:0.4
```

As such, the following blocks:

a)

```
$PROD:X s:1
  O:PX   Q:1
  I:PK   Q:0.6
  I:PL   Q:0.4
```

b)

```
$PROD:X s:1
  O:PX  Q:100
  I:PK  Q:60
  I:PL  Q:40
```

c)

```
$PROD:X s:1
  O:PX  Q:100
  I:PK  Q:60 P:2
  I:PL  Q:40 P:2
```

d)

```
$PROD:X s:1
  O:PX  Q:100 P:2
  I:PK  Q:60
  I:PL  Q:40
```

e)

```
$PROD:X s:1
  O:PX  Q:100 P:2
  I:PK  Q:60 P:2
  I:PL  Q:40 P:2
```

represent the same technology $X(K, L) = Z K^{0.6} L^{0.4}$, where from equation (42) $Z = 1.96$. However, a modeller should be careful, because rescaling of coefficients in the MPSGE blocks gives a different level of output.

Also remember that reference prices and quantities are used for calibration only and not transferred to a solver as initial values. Another important issue is that only *relative* prices matter for technology representation. As such, use of P:1 or P:2 for both inputs gives the same marginal rate of substitution. Price of output does not matter in this example because we have one output only. In the case of two outputs relative prices would matter as they determine marginal rate of transformation.

Now consider a production block:

```
$PROD:X s:1
  O:PX  Q:1
  I:PK  Q:0.6 P:2
  I:PL  Q:0.4
```

This production block is described by the function $X(K, L) = Z K^{0.75} L^{0.25}$, because the value share from equation (41) is $a = \frac{1.2}{1.2+0.4} = 0.75$, and from equation (42) $Z = 1/(0.6 * 0.75 * 0.4 * 0.25) = 1.844$.

Leontief production function can be represented either as

```
$PROD:X s:0
  O:PX  Q:100
  I:PK  Q:60
  I:PL  Q:40
```

or as

```

$PROD:X
O:PX   Q:100
I:PK   Q:60
I:PL   Q:40

```

because default value for top level elasticity is zero.

Preferences:

General form of a nested CES utility function is $U(X, Y) = (aX^\rho + (1 - a)Y^\rho)^{\frac{1}{\rho}}$ where ρ is an elasticity parameter. Elasticity of substitution σ is related to elasticity parameter as $\sigma = \frac{1}{1-\rho}$. \$DEMAND block represents utility derived from consumption of goods and consumer's endowments. If quantities demanded and endowments are all equal to 100, then demand block in MPSGE is:

```

$DEMAND:CONS s:(1/1-RHO)
D:PX   Q:100
D:PY   Q:100
E:PL   Q:100
E:PK   Q:100

```

There is a particular utility function which is widely used in a dynamic analysis. It is a Constant Intertemporal Elasticity of Substitution utility function (CIES), which has a form: $U(C_t) = \frac{C_t^{1-\theta}-1}{1-\theta}$, where $\sigma = \frac{1}{\theta}$ is the elasticity of substitution.

As $\theta \rightarrow 0$, $U(C)$ approaches a linear case $U(C_t) = C_t - 1$ (which represent the same preferences as $U(C_t) = C_t$). As $\theta \rightarrow 1$, then $U(C_t) = \ln(C_t)$.

In order to see a relationship between CES and CIES utility functions, we start with CES form: $U(C_t) = (aC_t^\rho)^{\frac{1}{\rho}}$. Then substitute $\rho = 1 - \theta$ because $\sigma = \frac{1}{\theta}$ and $\sigma = \frac{1}{1-\rho}$. It leads to $U(C) = (aC^{1-\theta})^{\frac{1}{1-\theta}}$. Maximization of $(aC^{1-\theta})^{\frac{1}{1-\theta}}$ and $\frac{aC^{1-\theta}}{1-\theta}$ produces the same result for optimal value of C (because $\max X^{\frac{1}{b}} \Leftrightarrow \max X^b \Leftrightarrow \max \frac{X}{b}$). For utility maximization over infinite horizon we sum over instantaneous utilities and adjust for a discount factor, which leads to a traditional form of utility function:

$$U = \sum_{t=0}^{\infty} \left(\frac{1}{1+\rho}\right)^t \frac{C_t^{1-\theta}-1}{1-\theta}.$$

If a consumer has an endowment of labor and capital and elasticity of intertemporal substitution SIGMA between consumption in different periods is defined as SIGMA=1/THETA; to reflect the relationship between σ and θ , then \$DEMAND block for CIES utility function has the following form:

```

$DEMAND:CONS s:SIGMA
D:PC(T)      Q:QREF(T)   P:PREF(T)
E:PL(T)      Q:QREF(T)
E:PK(TFIRST) Q:KO

```

Note that if consumer gets utility from two goods (for example, in a Cobb-Douglas form) we can keep the same \$DEMAND block as above and add a production block

```

$PROD:C(T) s:1
O:PC(T)      Q:CO
I:PX(T)      Q:XO
I:PY(T)      Q:YO

```

Appendix 2. MPSGE and algebraic formulation

Appendix 2 contains the MPSGE and algebraic (MCP) versions for the following problem (Shoven and Whalley, Applied General-Equilibrium Models of Taxation and International Trade: An Introduction and Survey, Journal of Economic Literature, Vol. 22, No. 3. (Sep., 1984), pp. 1007-1051). It is provided for illustration purposes. One can compare the degree of difficulty in both versions. If you are interested in an MCP version of a dynamic Ramsey model, see Rutherford's [1997] paper. I have some preliminary notes on conversion of MPSGE into MCP. Check with me if I have finished them yet.

In this model, there are two final commodities (M, N), two factors of production (K, L) and two consumers (R,P). Producers are competitive, minimizing unit cost taking market prices as given. Production functions are of the constant-elasticity-of-substitution (CES) form:

$$Q_i = \phi_i [\delta_i L_i^{\rho_i} + (1 - \delta) K_i^{\rho_i}]^{1/\rho_i} \quad i \in M, N$$

where $\rho_i = (\sigma_i - 1)/\sigma_i$, and σ_i is an elasticity of substitution between capital and labor. Parameters of the production function are given by:

Sector	ϕ_i	δ_i	σ_i
Manufacturing (M)	1.5	0.6	2.0
Non-Manufacturing (N)	2.0	0.7	0.5

Consumers are endowed with primary factors from which they earn income. This income is used to purchase goods to maximize utility. The CES utility functions are:

$$U_c = \left[\sum_i \alpha_{ic}^{1/\sigma_c} X_{ic}^{\rho_c} \right]^{1/\rho_c} \quad c \in R, P$$

where $\rho_c = (\sigma_c - 1)/\sigma_c$, and σ_c is an elasticity of substitution between goods M and N. Commodity endowments and parameters of the utility function are given by:

Household	K	L	α_M	α_N	σ_c
Rich	25	0	0.5	0.5	1.5
Poor	0	60	0.3	0.7	0.75

After computing the benchmark equilibrium, a single counterfactual experiment is done in which a 50% ad-valorem tax is levied on each unit of capital services employed in the manufacturing sector. Tax revenue is returned as lump-sum income for the two households, 40% to the rich and 60% to the poor.

`$title Compare MPSGE and its algebraic representation`

```
set i /M, N/;
set c /RICH, POOR/;
set rich(c) /rich/;
set poor(c) /poor/;
```

parameters

```
sigma /m 2, n 0.5/,
phi /m 1.5, n 2/,
delta /m 0.6, n 0.7/,
sig /rich 1.5, poor 0.75/,
capital /rich 25, poor 0/,
labor /rich 0, poor 60/;
```

```

table alpha(c,i)
      M      N
RICH  0.5    0.5
POOR  0.3    0.7 ;

parameter taxrate(i);

$ontext
$model:work

$SECTORS:
      Y(i)      !Production

$COMMODITIES:
      P(i)      !Price
      W          !Price of labor
      R          !Price of capital

$CONSUMERS:
      cons(c) ! Representative agent

$prod:y(i)      s:sigma(i)
      o:p(i)    q:phi(i)
      i:w       q:1          p:delta(i)
      i:r       q:1          p:(1-delta(i))
+
+
      a:cons("rich") t:(0.4*taxrate(i))
      a:cons("poor") t:(0.6*taxrate(i))

$demand:cons(c) s:sig(c)
      d:p(i)    q:1          p:(alpha(c,i)**(1/sig(c)))
      e:r       q:capital(c)
      e:w       q:labor(c)

$offtext
$sysinclude mpsgeset work

taxrate(i) = 0;
work.iterlim = 5000;
$include work.gen
solve work using mcp;

taxrate("M") = 0.5;
$include work.gen
solve work using mcp;

* MCP version

alias(i,j);

taxrate("M") = 0;

```

```

parameter pk0(i);
pk0(i) = 1+taxrate(i);

parameter thetak(i), thetal(i);
thetak(i) = 1-delta(i);
thetal(i) = delta(i);

parameter abeta(c,i);
abeta(c,i) = alpha(c,i)**(1/sig(c));

equations
    prf_y(i)
    mkt_y(i)
    mkt_k
    mkt_l
    i_cons(c);

* Zero profit

prf_y(i)..
    (thetak(i)*(r*(1+taxrate(i))/(pk0(i)*(1-delta(i))))** (1-sigma(i))
    + thetal(i)*(w/delta(i))** (1-sigma(i)))* (1/(1-sigma(i))))
    =e= p(i) * phi(i);

* Market clearance

mkt_y(i)..
    y(i) * phi(i) =e=
    sum(c, 1 *
    (((sum(j, abeta(c,j) * (p(j)/abeta(c,j))
    ** (1-sig(c))))** (1/(1-sig(c)))) * abeta(c,i) / p(i)) ** sig(c)) *
    cons(c) / ( ((sum(j, abeta(c,j) * (p(j)/abeta(c,j))
    ** (1-sig(c))))** (1/(1-sig(c)))) ) );

mkt_k..
    sum(c, capital(c)) =e=
    sum(i,
    ((thetal(i)*(w/delta(i))** (1-sigma(i))
    + thetak(i)*(r*(1+taxrate(i))/(pk0(i)*(1-delta(i))))** (1-sigma(i))
    ** (1/(1-sigma(i))) / (r*(1+taxrate(i))/(pk0(i)*(1-delta(i))))
    ** (sigma(i)) * y(i));

mkt_l..
    sum(c, labor(c)) =e=
    sum(i,
    ((thetal(i)*(w/delta(i))** (1-sigma(i))
    + thetak(i)*(r*(1+taxrate(i))/(pk0(i)*(1-delta(i))))** (1-sigma(i))
    ** (1/(1-sigma(i))) * delta(i)/w)** (sigma(i)) * y(i));

i_cons(c)..
    cons(c) =e= r*capital(c) + w*labor(c)

```

```

+ (sum(i, 0.6 * taxrate(i) * y(i) * r *
((thetal(i)*(w/delta(i))**(1-sigma(i))
+ thetak(i)*(r*(1+taxrate(i))/(pk0(i)*(1-delta(i))))**(1-sigma(i)))
** (1/(1-sigma(i)))
/ (r*(1+taxrate(i))/(pk0(i)*(1-delta(i))))**(sigma(i))))$poor(c)
+ (sum(i, 0.4 * taxrate(i) * y(i) * r *
((thetal(i)*(w/delta(i))**(1-sigma(i))
+ thetak(i)*(r*(1+taxrate(i))/(pk0(i)*(1-delta(i))))**(1-sigma(i)))
** (1/(1-sigma(i)))
/ (r*(1+taxrate(i))/(pk0(i)*(1-delta(i))))**(sigma(i))))$rich(c);

model obana /prf_y.y, mkt_y.p, mkt_k.r, mkt_l.w, i_cons.cons/;

cons.fx("rich") = 34.3368;
cons.fx("poor") = 60.0000;
obana.iterlim = 5000;
solve obana using mcp;

* Introduce tax

taxrate("M") = 0.5;
cons.fx("rich") = 29.0935;
cons.fx("poor") = 61.3484;
solve obana using mcp;

```

Appendix 3. Program listings

Appendix 3 contains GAMS programs which can be copied and executed. The following files are located here.

move1_1.gms move1_2.gms move1_3.gms move1_4.gms move1_5.gms	Static model	The Wonderland example no welfare block initial quantities in the listing report block vector syntax
move2_1.gms move2_2.gms move3_1.gms	Dynamic model	on a steady-state in a baseyear alternative formulation of capital block not on a steady-state in a baseyear

```
$TITLE: move1_1.gms - A Simple Static Model (The Wonderland example)
```

```
$ONTEXT
```

```
$MODEL:MOVE1_1
```

```
$SECTORS:
```

```
    X      ! Activity level for sector X
    Y      ! Activity level for sector Y
    W      ! Activity level for sector W (welfare index)
```

```
$COMMODITIES:
```

```
    PX     ! Price index for commodity X
    PY     ! Price index for commodity Y
    PL     ! Price index for primary factor L
    PK     ! Price index for primary factor K
    PW     ! Price index for welfare (expenditure function)
```

```
$CONSUMERS:
```

```
    CONS   ! Income level for consumer CONS
```

```
$PROD:X s:1
```

```
    O:PX   Q:100
    I:PL   Q:40
    I:PK   Q:60
```

```
$PROD:Y s:1
```

```
    O:PY   Q:100
    I:PL   Q:60
    I:PK   Q:40
```

```
$PROD:W s:1
```

```
    O:PW   Q:200
    I:PX   Q:100
    I:PY   Q:100
```

```
$DEMAND:CONS
```

```
    D:PW   Q:200
    E:PL   Q:100
    E:PK   Q:100
```

```
$OFFTEXT
```

```
$SYSINCLUDE mpsgeset MOVE1_1
```

```
MOVE1_1.ITERLIM = 0;
```

```
$INCLUDE MOVE1_1.GEN
```

```
SOLVE MOVE1_1 USING MCP;
```

```
$TITLE: move1_2.gms - A Simple Static Model (no welfare block)
```

```
$ONTEXT
```

```
$MODEL:MOVE1_2
```

```
$SECTORS:
```

```
    X      ! Activity level for sector X  
    Y      ! Activity level for sector Y
```

```
$COMMODITIES:
```

```
    PX      ! Price index for commodity X  
    PY      ! Price index for commodity Y  
    PL      ! Price index for primary factor L  
    PK      ! Price index for primary factor K
```

```
$CONSUMERS:
```

```
    CONS    ! Income level for consumer CONS
```

```
$PROD:X s:1
```

```
    O:PX    Q:100  
    I:PL    Q:40  
    I:PK    Q:60
```

```
$PROD:Y s:1
```

```
    O:PY    Q:100  
    I:PL    Q:60  
    I:PK    Q:40
```

```
$DEMAND:CONS s:1
```

```
    D:PX    Q:100  
    D:PY    Q:100  
    E:PL    Q:100  
    E:PK    Q:100
```

```
$OFFTEXT
```

```
$SYSINCLUDE mpsgeset MOVE1_2
```

```
MOVE1_2.ITERLIM = 0;
```

```
$INCLUDE MOVE1_2.GEN
```

```
SOLVE MOVE1_2 USING MCP;
```

```
$TITLE: move1_3.gms. Simple Static Model: Formulation for Initial Quantities
```

```
$ONTEXT
```

```
$MODEL:MOVE1_3
```

```
$SECTORS:
```

```
    X      ! Activity level for sector X
    Y      ! Activity level for sector Y
    W      ! Activity level for sector W (welfare index)
```

```
$COMMODITIES:
```

```
    PX     ! Price index for commodity X
    PY     ! Price index for commodity Y
    PL     ! Price index for primary factor L
    PK     ! Price index for primary factor K
    PW     ! Price index for welfare (expenditure function)
```

```
$CONSUMERS:
```

```
    CONS   ! Income level for consumer CONS
```

```
$PROD:X s:1
```

```
    O:PX   Q:1
    I:PL   Q:0.4
    I:PK   Q:0.6
```

```
$PROD:Y s:1
```

```
    O:PY   Q:1
    I:PL   Q:0.6
    I:PK   Q:0.4
```

```
$PROD:W s:1
```

```
    O:PW   Q:1
    I:PX   Q:0.5
    I:PY   Q:0.5
```

```
$DEMAND:CONS
```

```
    D:PW   Q:200
    E:PL   Q:100
    E:PK   Q:100
```

```
$OFFTEXT
```

```
$SYSINCLUDE mpsgeset MOVE1_3
```

```
X.L=100;
```

```
Y.L=100;
```

```
W.L=200;
```

```
MOVE1_3.ITERLIM = 0;
```

```
$INCLUDE MOVE1_3.GEN
```

```
SOLVE MOVE1_3 USING MCP;
```

\$TITLE: move1_4.gms - A Simple Static Model (report block)

\$ONTEXT

\$MODEL:MOVE1_4

\$SECTORS:

X ! Activity level for sector X
 Y ! Activity level for sector Y
 W ! Activity level for sector W (welfare index)

\$COMMODITIES:

PX ! Price index for commodity X
 PY ! Price index for commodity Y
 PL ! Price index for primary factor L
 PK ! Price index for primary factor K
 PW ! Price index for welfare (expenditure function)

\$CONSUMERS:

CONS ! Income level for consumer CONS

\$PROD:X s:1

O:PX Q:100
 I:PL Q:40
 I:PK Q:60

\$PROD:Y s:1

O:PY Q:100
 I:PL Q:60
 I:PK Q:40

\$PROD:W s:1

O:PW Q:200
 I:PX Q:100
 I:PY Q:100

\$DEMAND:CONS

D:PW Q:200
 E:PL Q:100
 E:PK Q:100

\$REPORT:

V:X_OUT	O:PX	PROD:X
V:K_X	I:PK	PROD:X
V:L_X	I:PL	PROD:X
V:Y_OUT	O:PY	PROD:Y
V:K_Y	I:PK	PROD:Y
V:L_Y	I:PL	PROD:Y
V:W_OUT	O:PW	PROD:W
V:X_W	I:PX	PROD:W
V:Y_W	I:PY	PROD:W
V:C	D:PW	DEMAND:CONS

```
$OFFTEXT

$SYSINCLUDE mpsgeset MOVE1_4

MOVE1_4.ITERLIM = 0;
$INCLUDE MOVE1_4.GEN
SOLVE MOVE1_4 USING MCP;

parameter report;

report("px","X") = x_out.l;
report("py","y") = y_out.l;
report("pw","w") = w_out.l;
report("pl","X") = -l_x.l;
report("pk","X") = -k_x.l;
report("pl","y") = -l_y.l;
report("pk","y") = -k_y.l;
report("px","w") = -x_w.l;
report("py","w") = -y_w.l;
report("pl","cons") = l_x.l+l_y.l;
report("pk","cons") = k_x.l+k_y.l;
report("pw","cons") = -w_out.l;

display report;
```

```
$TITLE: move1_5.gms - A Simple Static Model (vector syntax)
```

```
set      i Sectors /x,y/,
         f Factors /k,l/;
```

```
TABLE sam(*,*) Benchmark data
```

	X	Y	W	CONS
X	100		-100	
Y		100	-100	
W			200	-200
L	-40	-60		100
K	-60	-40		100 ;

```
PARAMETER      YO(I)           Benchmark sectoral output,
                FDO(F,I)       Benchmark factor demands,
                CO(I)          Benchmark consumption demand,
                EO(F)          Factor endowments,
                WO             Benchmark total consumption;
```

```
*      Extract data from the original format into model-specific arrays:
```

```
YO(I)          = SAM(I,I);
FDO(F,I)       = -SAM(F,I);
CO(I)          = -SAM(I,"W");
WO             = SUM(I, CO(I));
EO(F)          = SAM(F,"CONS");
```

```
$ONTEXT
```

```
$MODEL:MOVE1_5
```

```
$SECTORS:
```

```
    OUT(i)      ! Production Activity Level
    W           ! Welfare Index
```

```
$COMMODITIES:
```

```
    P(i)        ! Price index for commodities
    PF(f)       ! Price index for factors
    PW          ! Utility Price Index
```

```
$CONSUMERS:
```

```
    CONS        ! Income level for consumer CONS
```

```
$PROD:OUT(i)
```

```
    O:P(i)      Q:YO(i)
    I:PF(f)     Q:FDO(f,i)
```

```
$PROD:W s:1
```

```
      O:PW          Q:W0
      I:P(i)        Q:CO(i)

$DEMAND:CONS
      D:PW          Q:W0
      E:PF(f)      Q:EO(f)

$REPORT:
      V:OUT_o(i)    O:P(i)      PROD:OUT(i)
      V:INP(f,i)    I:PF(f)     PROD:OUT(i)
      V:OUT_w       O:PW        PROD:W
      V:INP_w(i)    I:P(i)      PROD:W
      V:C           D:PW        DEMAND:CONS

$OFFTEXT

$SYSINCLUDE mpsgeset MOVE1_5

MOVE1_5.ITERLIM = 0;
$INCLUDE MOVE1_5.GEN
SOLVE MOVE1_5 USING MCP;

parameter report;

report(i,i) = out_o.l(i);
report("w","w") = out_w.l;
report(i,"w") = -inp_w.l(i);
report(f,i) = inp.l(f,i);
report(f,"cons") = sum(i, inp.l(f,i));
report("w","cons") = -out_w.l;

display report;
```

\$TITLE: move2_1.gms. A Simple Ramsey Dynamic Model

SET T /1*10/,
TFIRST(T),
TLAST(T);

TFIRST(T) = YES\$(ORD(T) EQ 1);
TLAST(T) = YES\$(ORD(T) EQ CARD(T));

SCALAR	DELTA	DISCOUNT FACTOR	/0.05/
	R	INTEREST RATE	/0.05/
	G	GROWTH RATE	/0.02/
	VK	INITIAL CAPITAL EARNINGS	/100.0/
	KO	INITIAL CAPITAL STOCK	
	RKO	INITIAL RETURN TO CAPITAL	
	IO	INITIAL INVESTMENT;	

PARAMETER QREF(T) QUANTITIES
PREF(T) PRICES;

RKO = DELTA+R;
KO = VK/RKO;
IO = (DELTA + G) * KO;
QREF(T) = (1 + G) ** (ORD(T) - 1);
PREF(T) = (1/(1 + R)) ** (ORD(T) - 1);

\$ONTEXT
\$MODEL:MOVE2_1

\$SECTORS:
X(T) ! Activity level for sector X
Y(T) ! Activity level for sector Y
W(T) ! Activity level for sector W (welfare index)
I(T) ! Investment Sector
K(T) ! Capital Accumulation

\$COMMODITIES:
PX(T) ! Price index for commodity X
PY(T) ! Price index for commodity Y
PL(T) ! Price index for primary factor L
PK(T) ! Price index for primary factor K
PW(T) ! Price index for welfare (expenditure function)
RK(T) ! Rental rate for capital
PKT ! Post-terminal capital constraint

\$CONSUMERS:
CONS ! Income level for consumer CONS

\$AUXILIARY:
TK ! Terminal Capital Stock

```

$PROD:X(T) s:1
    O:PX(T)          Q:100
    I:PL(T)          Q:40
    I:RK(T)          Q:60

$PROD:Y(T) s:1
    O:PY(T)          Q:100
    I:PL(T)          Q:60
    I:RK(T)          Q:40

$PROD:K(T)
    O:PK(T+1)        Q:((1-DELTA)*KO)
    O:PKT$TLAST(T)  Q:((1-DELTA)*KO)
    O:RK(T)          Q:(KO*(DELTA+R))
    I:PK(T)          Q:KO

$PROD:I(T)
    O:PK(T+1)        Q:IO
    O:PKT$TLAST(T)  Q:IO
    I:PY(T)          Q:(0.5*IO)
    I:PX(T)          Q:(0.5*IO)

$PROD:W(T) s:1
    O:PW(T)          Q:130
    I:PX(T)          Q:65
    I:PY(T)          Q:65

$DEMAND:CONS s:1
    D:PW(T)          Q:(130*QREF(T)) P:PREF(T)
    E:PL(T)          Q:(100*QREF(T))
    E:PK(TFIRST)    Q:KO
    E:PKT           Q:(-1)   R:TK

$CONSTRAINT:TK
    SUM(T$TLAST(T), I(T)/I(T-1) - Y(T)/Y(T-1)) =G= 0;

$OFFTEXT

$SYSINCLUDE mpsgeset MOVE2_1

X.L(T) = QREF(T);
Y.L(T) = QREF(T);
W.L(T) = QREF(T);
I.L(T) = QREF(T);
K.L(T) = QREF(T);
TK.L   = KO * (1 + G) ** CARD(T);

PX.L(T) = PREF(T);
PY.L(T) = PREF(T);
PL.L(T) = PREF(T);
PK.L(T) = (1+R)*PREF(T);

```

```
PW.L(T) = PREF(T);  
RK.L(T) = PREF(T);  
PKT.L   = SUM(TLAST, PK.L(TLAST)/(1+R));
```

```
MOVE2_1.ITERLIM = 0;  
$INCLUDE MOVE2_1.GEN  
SOLVE MOVE2_1 USING MCP;
```

\$TITLE move2_2.gms Simple Dynamic Model

* No $RK0=DELTA+R$ in $0:RK(T)$, but in initial $RK.L$ and
 * in production blocks

SET T /1*10/,
 TFIRST(T),
 TLAST(T);

TFIRST(T) = YES\$(ORD(T) EQ 1);
 TLAST(T) = YES\$(ORD(T) EQ CARD(T));

SCALAR	DELTA	DISCOUNT FACTOR	/0.05/
	R	INTEREST RATE	/0.05/
	G	GROWTH RATE	/0.02/
	VK	INITIAL CAPITAL EARNINGS	/100.0/
	KO	INITIAL CAPITAL STOCK	
	RK0	INITIAL RETURN TO CAPITAL	
	IO	INITIAL INVESTMENT;	

PARAMETER QREF(T) QUANTITIES
 PREF(T) PRICES;

$RK0 = DELTA+R$;
 $KO = VK/RK0$;
 $IO = (DELTA + G) * KO$;
 $QREF(T) = (1 + G) ** (ORD(T) - 1)$;
 $PREF(T) = (1/(1 + R)) ** (ORD(T) - 1)$;

\$ONTEXT
 \$MODEL:MOVE2_2

\$SECTORS:

X(T) ! Activity level for sector X
 Y(T) ! Activity level for sector Y
 W(T) ! Activity level for sector W (welfare index)
 I(T) ! Investment sector
 K(T) ! Capital accumulation

\$COMMODITIES:

PX(T) ! Price index for commodity X
 PY(T) ! Price index for commodity Y
 PL(T) ! Price index for primary factor L
 PK(T) ! Price index for primary factor K
 PW(T) ! Price index for welfare (expenditure function)
 RK(T) ! Rental rate for capital
 PKT ! Post-terminal capital constraint

\$CONSUMERS:

```

      CONS      ! Income level for consumer CONS

$AUXILIARY:
      TK        ! Terminal Capital Stock

$PROD:X(T) s:1
      O:PX(T)          Q:100
      I:PL(T)          Q:40
      I:RK(T)          Q:(60/RK0)      P:RK0

$PROD:Y(T) s:1
      O:PY(T)          Q:100
      I:PL(T)          Q:60
      I:RK(T)          Q:(40/RK0)      P:RK0

$PROD:K(T)
      O:PK(T+1)        Q:((1-DELTA)*KO)
      O:PKT$TLAST(T)  Q:((1-DELTA)*KO)
      O:RK(T)          Q:KO
      I:PK(T)          Q:KO

$PROD:I(T)
      O:PK(T+1)        Q:IO
      O:PKT$TLAST(T)  Q:IO
      I:PY(T)          Q:(0.5*IO)
      I:PX(T)          Q:(0.5*IO)

$PROD:W(T) s:1
      O:PW(T)          Q:130
      I:PX(T)          Q:65
      I:PY(T)          Q:65

$DEMAND:CONS s:1
      D:PW(T)          Q:(130*QREF(T)) P:PREF(T)
      E:PL(T)          Q:(100*QREF(T))
      E:PK(TFIRST)    Q:KO
      E:PKT           Q:(-1)      R:TK

$CONSTRAINT:TK
      SUM(T$TLAST(T), I(T)/I(T-1) - Y(T)/Y(T-1)) =G= 0;

$OFFTEXT

$SYSINCLUDE mpsgeset MOVE2_2

X.L(T) = QREF(T);
Y.L(T) = QREF(T);
W.L(T) = QREF(T);
I.L(T) = QREF(T);
K.L(T) = QREF(T);
TK.L   = KO * (1 + G) ** CARD(T);

```

```
PX.L(T) = PREF(T);
PY.L(T) = PREF(T);
PL.L(T) = PREF(T);
PK.L(T) = (1+R)*PREF(T);
PW.L(T) = PREF(T);
RK.L(T) = RKO*PREF(T);
PKT.L   = SUM(TLAST, PK.L(TLAST)/(1+R));
```

```
MOVE2_2.ITERLIM = 0;
$INCLUDE MOVE2_2.GEN
SOLVE MOVE2_2 USING MCP;
```

\$TITLE move3_1.gms. Simple Dynamic Model with off steady-state investment

\$ontext

Before dynamics. Static model SAM.

Markets	Production Sectors			Consumers
	X	Y	W	CONS
PX	100		-100	
PY		100	-100	
PW			200	-200
PL	-40	-60		100
PK	-60	-40		100

After dynamics. SAM is consistent with a steady-state in a base period.

	X	Y	W	Inv	CONS
PX	100		-65	-35	
PY		100	-65	-35	
PW			200		-200
PL	-40	-60			100
PK	-60	-40			100
sav			-70	70	

After dynamics. SAM is away of a steady-state in base period.

	X	Y	W	Inv	CONS
PX	100		-60	-40	
PY		100	-60	-40	
PW			200		-200
PL	-40	-60			100
PK	-60	-40			100
sav			-80	80	

\$OFFTEXT

```
SET      T          /1*100/,
         TFIRST(T),
         TLAST(T);
```

```
TFIRST(T) = YES$(ORD(T) EQ 1);
TLAST(T) = YES$(ORD(T) EQ CARD(T));
```

```
set      trep(t) period labels for investment report /1*5/,
         tdecade(t) /1,10,20,30,40,50,60,70,80,90,100/;
```

```
SCALAR  DELTA      DISCOUNT FACTOR          /0.05/
```

```

R      INTEREST RATE           /0.05/
G      GROWTH RATE             /0.02/
KZERO  INITIAL CAPITAL EARNINGS /100.0/
KO     INITIAL CAPITAL STOCK
RKO    INITIAL RETURN TO CAPITAL
RKG    GUESS ON INITIAL RETURN TO CAPITAL
IO     INITIAL INVESTMENT;

```

```

PARAMETER      QREF(T) QUANTITIES
                PREF(T) PRICES;

```

```

* IN THE ON-SS CASE, LABOR AND CAPITAL GROW AT A RATE G.
* INITIAL RETURN TO CAPITAL IS RKO = DELTA+R;
* IN THE OFF-SS CASE LABOR GROWS AT A RATE G, CAPITAL GROWS AT A
* DIFFERENT RATE, DETERMINED BY THE INITIAL RETURN TO CAPITAL
* (DENOTED AS RKG). WE DON'T KNOW RKG, SO WE MAKE A GUESS TRYING
* TO BE CONSISTENT WITH GIVEN INITIAL INVESTMENT.

```

```

RKO = DELTA+R;
KO = KZERO/RKO;
IO = (DELTA + G) * KO +10;

```

```

* RKO is unknown, we use formulas for RKO, KO, IO for consistency
* with the on SS dynamic model. We could also simply use the
* declarations as KO = 1000; IO = 80; (we know these values
* from SAM).

```

```
RKG = 0.12;
```

```

QREF(T) = (1 + G) ** (ORD(T) - 1);
PREF(T) = (1/(1 + R)) ** (ORD(T) - 1);

```

```

$ONTEXT
$MODEL:DYNAMIC

```

```
$SECTORS:
```

```

X(T)      ! ACTIVITY LEVEL FOR SECTOR X
Y(T)      ! ACTIVITY LEVEL FOR SECTOR Y
W(T)      ! ACTIVITY LEVEL FOR SECTOR W
I(T)      ! INVESTMENT SECTOR
K(T)      ! CAPITAL ACCUMULATION

```

```
$COMMODITIES:
```

```

PX(T)     ! PRICE INDEX FOR COMMODITY X
PY(T)     ! PRICE INDEX FOR COMMODITY Y
PL(T)     ! PRICE INDEX FOR PRIMARY FACTOR L
PK(T)     ! PRICE INDEX FOR PRIMARY FACTOR K
PW(T)     ! PRICE INDEX FOR WELFARE
RK(T)     ! RENTAL RATE FOR CAPITAL

```

```

        PKT          ! POST-TERMINAL CAPITAL CONSTRAINT

$CONSUMERS:
        CONS          ! INCOME LEVEL FOR CONSUMER CONS

$AUXILIARY:
        TK            ! TERMINAL CAPITAL STOCK

$PROD:X(T) S:1
        O:PX(T)       Q:100
        I:PL(T)       Q:40
        I:RK(T)       Q:(60/RKG)      P:RKG

$PROD:Y(T) S:1
        O:PY(T)       Q:100
        I:PL(T)       Q:60
        I:RK(T)       Q:(40/RKG)      P:RKG

$PROD:K(T)
        O:PK(T+1)     Q:(1-DELTA)
        O:PKT$TLAST(T) Q:(1-DELTA)
        O:RK(T)       Q:1
        I:PK(T)       Q:1

$PROD:I(T)
        O:PK(T+1)     Q:IO
        O:PKT$TLAST(T) Q:IO
        I:PY(T)       Q:(0.5*IO)
        I:PX(T)       Q:(0.5*IO)

$PROD:W(T) s:1
        O:PW(T)       Q:130
        I:PX(T)       Q:65
        I:PY(T)       Q:65

$DEMAND:CONS s:1
        D:PW(T)       Q:(130*QREF(T)) P:PREF(T)
        E:PL(T)       Q:(100*QREF(T))
        E:PK(TFIRST)  Q:(KZERO/RKG)
        E:PKT         Q:(-1)      R:TK

$REPORT:
        V:C(T)  D:PW(T)          DEMAND:CONS
        V:XX(T) O:PX(T)          PROD:X(T)
        V:YY(T) O:PY(T)          PROD:Y(T)

$CONSTRAINT:TK
        SUM(T$TLAST(T), I(T)/I(T-1) - Y(T)/Y(T-1)) =G= 0;

$OFFTEXT

```

```

$SYSINCLUDE MPSGESET DYNAMIC

X.L(T) = QREF(T);
Y.L(T) = QREF(T);
W.L(T) = QREF(T);
I.L(T) = QREF(T);
K.L(T) = KO*QREF(T);
TK.L   = KO * (1 + G) ** CARD(T);

PX.L(T) = PREF(T);
PY.L(T) = PREF(T);
PL.L(T) = PREF(T);
PK.L(T) = (1+R)*PREF(T);
PW.L(T) = PREF(T);
RK.L(T) = RKG*PREF(T);
PKT.L   = SUM(TLAST, PK.L(TLAST)/(1+R));

SET      ITER      JACOBI-SECANT ITERATIONS / ITER1*ITER6/;

PARAMETER
    INVEST  INVESTMENT PATH
    INTRATE INTEREST RATE
    CONSUM  CONSUMPTION LEVEL
    GROWTH  CONSUMPTION GROWTH RATE
    ERROR   ERROR ESTIMATE BY ITERATION
    RKOVAL  VALUE OF RKO;

LOOP(ITER,

$INCLUDE DYNAMIC.GEN
    SOLVE DYNAMIC USING MCP;

    INVEST(TREP,ITER) = I.L(TREP) * IO;
    INTRATE(TFIRST,ITER) = RKG-DELTA;
    INTRATE(T+1,ITER)$PX.L(T+1) = PX.L(T)/PX.L(T+1) - 1;
    CONSUM(T,ITER) = C.L(T);

*      ERROR(ITER) = C.L("1") - (200-IO);
*      ERROR(ITER) = I.L("1")*IO - IO;
ERROR(ITER) = C.L("1") - (XX.L("1")+YY.L("1")-IO);
RKOVAL(ITER) = RKG;

    IF (ORD(ITER)-1,
        RKG = RKG + ERROR(ITER)/(ERROR(ITER-1)-ERROR(ITER))
            * (RKOVAL(ITER)-RKOVAL(ITER-1));
    ELSE
        RKG = RKG + 0.01;
    );

);
INVEST(TREP,"IO") = IO;
DISPLAY INVEST, INTRATE, CONSUM;

```

```
$SETGLOBAL labels TREP  
$LIBINCLUDE PLOT INVEST
```

```
CONSUM(T,ITER)$ (ORD(T) GT CARD(T)-5) = NA;  
GROWTH(T,ITER)$ (CONSUM(T+1,ITER) NE NA) =  
    CONSUM(T+1,ITER)/CONSUM(T,ITER)-1;  
INTRATE(T,ITER)$ (ORD(T) GT CARD(T)-5) = NA;
```

```
$SETGLOBAL labels TDECADE  
$LIBINCLUDE PLOT CONSUM  
$LIBINCLUDE PLOT GROWTH  
$LIBINCLUDE PLOT INTRATE
```