

Eleventh Floor, Menzies Building
Monash University, Wellington Road
CLAYTON Vic 3800 AUSTRALIA

Telephone:
(03) 9905 2398, (03) 9905 5112

Fax:
(03) 9905 2426
e-mail:

Internet home page:

from overseas:
61 3 9905 2398 or
61 3 9905 5112

61 3 9905 2426
impact@buseco.monash.edu.au
<http://www.monash.edu.au/policy/>

PHILGEM: A SAM-based
Computable General Equilibrium
Model of the Philippines

by

Erwin L. Corong and J. Mark Horridge

Centre of Policy Studies, Monash University

General Paper No. G-227 April 2012

ISSN 1 031 9034

ISBN 978 1 921654 35 0

The Centre of Policy Studies (COPS) is a research centre at Monash University devoted to economy-wide modelling of economic policy issues.

CENTRE of

POLICY

STUDIES and

the IMPACT

PROJECT

PHILGEM: A SAM-based Computable General Equilibrium Model of the Philippines

Erwin L. Corong and J. Mark Horridge¹

Centre of Policy Studies, Monash University, Australia

Abstract: This paper describes the structure of PHILGEM, a single country computable general equilibrium (CGE) model of the Philippine economy. PHILGEM offers a good starting point for model development, especially for researchers who may want to extend their ORANI-G models to draw on supplementary data coming from a social accounting matrix (SAM). A generic version of the model is described here designed for expository purposes and for adaptation to other countries. The description of PHILGEM's equations and database is closely integrated with an explanation of how the model is solved using the GEMPACK system. Computer files are freely available, which contain a complete model specification and database.

Bibliographical note: The PHILGEM model and its documentation draws from:
Horridge, J.M. (2003) *ORANI-G: A Generic Single-Country Computable General Equilibrium Model*, Centre of Policy Studies, Monash University.

Latest PHILGEM or ORANI-G related material will be found at:
<http://www.monash.edu.au/policy/oranig.htm>

¹ Respectively PhD candidate; and Professor and Director of GEMPACK software, Centre of Policy Studies (CoPS), Monash University, Australia.

Contents

1. Introduction	1
2. Model Structure and Interpretation of Results	2
2.1. A comparative-static interpretation of model results	2
3. The Percentage-Change Approach to Model Solution	3
3.1. Levels and linearised systems compared: a small example	5
3.2. The initial solution	6
4. The Equations of PHILGEM	6
4.1. The TABLO language	7
4.2. The naming system	8
4.3. The model's data base	9
4.4. Dimensions of the model	15
4.5. Core data coefficients and related variables	17
4.6. The equation system	22
4.7. Structure of production	22
4.8. Demands for primary factors	23
4.9. Sourcing of intermediate inputs	26
4.10. Top production nest	28
4.11. Industry costs and production taxes	29
4.12. From industry outputs to commodity outputs	30
4.13. Export and local market versions of each good	31
4.14. Demands for investment goods	32
4.15. Household demands	34
4.16. Export demands	38
4.17. Other final demands	40
4.18. Demands for margins	41
4.19. Formulae for sales aggregates	41
4.20. Market-clearing equations	42
4.21. Purchasers' prices	43
4.22. Indirect taxes	44
4.23. GDP from the income and expenditure sides	46
4.24. The trade balance and other aggregates	49
4.25. Primary factor aggregates	50
4.26. Rates of return and investment	51
4.27. The labour market	53
4.28. Miscellaneous equations	54
4.29. Adding variables for explaining results	54
4.30. Sales decomposition	55
4.31. The Fan decomposition	55
4.32. The expenditure side GDP decomposition	56
4.33. Checking the data	59
4.34. Summarizing the data	60
4.35. Import shares and short-run supply elasticities	61
4.36. Storing Data for Other Computations	62
5. Equations for SAM extension	64
5.1. Gross Operating Surplus	64
5.2. Enterprises Account	65
5.3. Labour income of households	67
5.4. Household income	69
5.5. Household consumption function, savings and other transfers	71
5.6. Government Income	71
5.7. Government expenditure	72

5.8. Private investment expenditure	73
5.9. Rest of the world	74
5.10. SAM consistency	74
5.11. Revenue Neutrality	75
5.12. Household results for reporting purposes	79
5.13. Deriving a Macro SAM	76
5.14. Condensation	78
6. Closing the Model	81
7. Using GEMPACK to Solve the Model	85
8. Conclusion	87
References	88
Appendix A: Percentage-Change Equations of a CES Nest	89
Appendix B: PHILGEM on the Web	91
Appendix C: Hardware and Software Requirements for Using GEMPACK	91
Appendix D: Main differences between full-size PHILGEM and the version described here	92
Appendix E: Deriving Percentage-Change Forms	92
Appendix F: Algebra for the Linear Expenditure System	93
Appendix G: Making your own AGE model from PHILGEM	95
Appendix H: Formal Checks on Model Validity; Debugging Strategies	97
Appendix I: Short-Run Supply Elasticity	98
Appendix J: List of Variables	100

List of Figures

Figure 1. Comparative-static interpretation of results	2
Figure 2. Linearisation error	4
Figure 3. Multistep process to reduce linearisation error	4
Figure 4. The PHILGEM Flows Database	10
Figure 5. PHILGEM Aggregate Social Accounting Matrix (SAM) Database	12
Figure 6. PHILGEM Aggregate SAM (in millions of pesos)	14
Figure 7. Structure of Production	23
Figure 8. Demand for different types of labour	24
Figure 9. Primary Factor Demand	25
Figure 10. Intermediate input sourcing decision	27
Figure 11. Composition of output	30
Figure 12. Structure of Investment Demand	33
Figure 13. Structure of Consumer Demand	35
Figure 14. Building a model-specific EXE file	85
Figure 15. Using the model-specific EXE to run a simulation	86
Figure 16. The GEMSIM alternative	87

1. Introduction

This document describes PHILGEM, a single-country computable general equilibrium (CGE) model of the Philippine economy. PHILGEM extends the well-known ORANI-G model of the Australian economy² by introducing (a) multiple households, and (b) additional equations to facilitate use of data coming from a social accounting matrix (SAM). As a result, PHILGEM illuminates the linkage between producing sectors and the rest of the economy and tracks how income is generated and consequently distributed and transferred.

Like ORANI-G, PHILGEM is formulated as a system of linear equations and solved using GEMPACK (Harrison and Pearson, 1994), a flexible system for solving computable or applied general equilibrium (AGE) models. GEMPACK automates the process of translating the model specification into a model solution program. The GEMPACK user needs no programming skills. Instead, he/she creates a TAB text file, listing the equations of the model. The syntax of this file resembles ordinary algebraic notation. The GEMPACK program TABLO then translates this text file into a model-specific program which solves the model.

PHILGEM offers a good starting point for model development especially for researchers who may want to extend their ORANI-G-based models to draw on supplementary data coming from a SAM. Indeed as will be shown in this document, this extension simply entails adding a few equations at the bottom of the original ORANI-G model specification. Hence, most details of ORANI-G are unchanged.

Nevertheless, for the sake of completeness and convenience, this paper describes the *whole* PHILGEM model, borrowing large sections unchanged from the ORANI-G documentation³. It consists of:

- an outline of the structure of the model and of the appropriate interpretations of the results of comparative-static and forecasting simulations;
- a description of the solution procedure;
- a brief description of the data, emphasising the general features of the data structure required for such a model;
- a complete description of the theoretical specification of the model framed around the TABLO Input file which implements the model in GEMPACK; and
- a guide to the GEMPACK system.

Sections altered or added for PHILGEM are Sections 4.2 to 4.4, 4.15, the whole of Section 5, parts of Section 6 and parts of Appendices G and J. The remainder is essentially identical to the ORANI-G document.

A downloadable set of computer files complements this document—see Appendix B. The files contain the PHILGEM TABLO Input file and a 25-sector database with 2 occupational and 20 household categories. Some version of GEMPACK is required to solve the model—see Appendix C.

² ORANI-G is a generic version of the ORANI applied general equilibrium (AGE) model of the Australian economy, which was first developed in the late 1970s as part of the government-sponsored IMPACT project. See: Powell, 1977; Dixon, Parmenter, Ryland and Sutton, 1977; Dixon, Parmenter, Sutton and Vincent (DPSV/Green Book), 1982. ORANI-G has also been used as a launching pad for developing new CGE models for other countries. Among others, these include Brazil, Finland, Malaysia, South Africa, Vietnam, Indonesia, South Korea, Thailand, the Philippines, Pakistan, Denmark, Uganda, both Chinas and Fiji.

³ The ORANI-G documentation may be downloaded from <http://www.monash.edu.au/policy/oranig.htm>

2. Model Structure and Interpretation of Results

PHILGEM has a theoretical structure which is typical of a static AGE model. It consists of equations describing, for some time period:

- producers' demands for produced inputs and primary factors;
- producers' supplies of commodities;
- demands for inputs to capital formation;
- household demands;
- export demands;
- government demands;
- the relationship of basic values to production costs and to purchasers' prices;
- market-clearing conditions for commodities and primary factors; and
- numerous macroeconomic variables and price indices.

To facilitate use of SAM data, PHILGEM includes behavioural equations and identities describing, for some time period:

- household income and transfers
- enterprises income;
- government income and transfers;
- domestic receipts from and transfers to the rest of the world (ROW)

Demand and supply equations for private-sector agents are derived from the solutions to the optimisation problems (cost minimisation, utility maximisation, etc.) which are assumed to underlie the behaviour of the agents in conventional neoclassical microeconomics. The agents are assumed to be price-takers, with producers operating in competitive markets which prevent the earning of pure profits.

2.1. A comparative-static interpretation of model results

Like the majority of AGE models, PHILGEM is designed for comparative-static simulations. Its equations and variables, which are described in detail in Section 4, all refer implicitly to the economy at some future time period.

This interpretation is illustrated by Figure 1, which graphs the values of some variable, say employment, against time. A is the level of employment in the base period (period 0) and B is the level which it would attain in T years time if some policy—say a tariff change—were *not* implemented. With the tariff change, employment would reach C, all other things being equal. In a comparative-static simulation, PHILGEM might generate the percentage change in employment $100(C-B)/B$, showing how employment *in period T* would be affected by the tariff change alone.

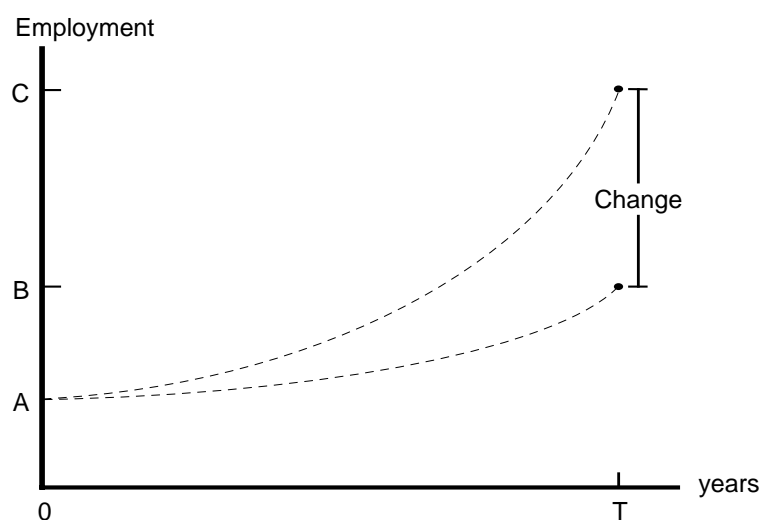


Figure 1. Comparative-static interpretation of results

3. The Percentage-Change Approach to Model Solution

Many of the PHILGEM equations are non-linear—demands depend on price ratios, for example. However, following Johansen (1960), the model is solved by representing it as a series of linear equations relating percentage changes in model variables. This section explains how the linearised form can be used to generate exact solutions of the underlying, non-linear, equations, as well as to compute linear approximations to those solutions⁴.

A typical AGE model can be represented in the levels as:

$$\mathbf{F}(\mathbf{Y}, \mathbf{X}) = \mathbf{0}, \quad (1)$$

where \mathbf{Y} is a vector of endogenous variables, \mathbf{X} is a vector of exogenous variables and \mathbf{F} is a system of non-linear functions. The problem is to compute \mathbf{Y} , given \mathbf{X} . Normally we cannot write \mathbf{Y} as an explicit function of \mathbf{X} .

Several techniques have been devised for computing \mathbf{Y} . The linearised approach starts by assuming that we already possess some solution to the system, $\{\mathbf{Y}^0, \mathbf{X}^0\}$, i.e.,

$$\mathbf{F}(\mathbf{Y}^0, \mathbf{X}^0) = \mathbf{0}. \quad (2)$$

Normally the initial solution $\{\mathbf{Y}^0, \mathbf{X}^0\}$ is drawn from historical data—we assume that our equation system was true for some point in the past. With conventional assumptions about the form of the \mathbf{F} function it will be true that for small changes $d\mathbf{Y}$ and $d\mathbf{X}$:

$$\mathbf{F}_Y(\mathbf{Y}, \mathbf{X})d\mathbf{Y} + \mathbf{F}_X(\mathbf{Y}, \mathbf{X})d\mathbf{X} = \mathbf{0}, \quad (3)$$

where \mathbf{F}_Y and \mathbf{F}_X are matrices of the derivatives of \mathbf{F} with respect to \mathbf{Y} and \mathbf{X} , evaluated at $\{\mathbf{Y}^0, \mathbf{X}^0\}$. For reasons explained below, we find it more convenient to express $d\mathbf{Y}$ and $d\mathbf{X}$ as small percentage changes \mathbf{y} and \mathbf{x} . Thus \mathbf{y} and \mathbf{x} , some typical elements of \mathbf{y} and \mathbf{x} , are given by:

$$y = 100dY/Y \quad \text{and} \quad x = 100dX/X. \quad (4)$$

Correspondingly, we define:

$$\mathbf{G}_Y(\mathbf{Y}, \mathbf{X}) = \mathbf{F}_Y(\mathbf{Y}, \mathbf{X})\hat{\mathbf{Y}} \quad \text{and} \quad \mathbf{G}_X(\mathbf{Y}, \mathbf{X}) = \mathbf{F}_X(\mathbf{Y}, \mathbf{X})\hat{\mathbf{X}}, \quad (5)$$

where $\hat{\mathbf{Y}}$ and $\hat{\mathbf{X}}$ are diagonal matrices. Hence the linearised system becomes:

$$\mathbf{G}_Y(\mathbf{Y}, \mathbf{X})\mathbf{y} + \mathbf{G}_X(\mathbf{Y}, \mathbf{X})\mathbf{x} = \mathbf{0}. \quad (6)$$

Such systems are easy for computers to solve, using standard techniques of linear algebra. But they are accurate only for small changes in \mathbf{Y} and \mathbf{X} . Otherwise, linearisation error may occur. The error is illustrated by Figure 2, which shows how some endogenous variable Y changes as an exogenous variable X moves from X^0 to X^F . The true, non-linear relation between X and Y is shown as a curve. The linear, or first-order, approximation:

$$y = -\mathbf{G}_Y(\mathbf{Y}, \mathbf{X})^{-1}\mathbf{G}_X(\mathbf{Y}, \mathbf{X})x \quad (7)$$

leads to the Johansen estimate Y^J —an approximation to the true answer, Y^{exact} .

Figure 2 suggests that, the larger is x , the greater is the proportional error in y . This observation leads to the idea of breaking large changes in X into a number of steps, as shown in Figure 3. For each sub-change in X , we use the linear approximation to derive the consequent sub-change in Y . Then, using the new values of X and Y , we recompute the coefficient matrices \mathbf{G}_Y and \mathbf{G}_X . The process is repeated for each step. If we use 3 steps (see Figure 3), the final value of Y , Y^3 , is closer to Y^{exact} than was the Johansen estimate Y^J . We can show, in fact, that given sensible restrictions on the derivatives of $\mathbf{F}(\mathbf{Y}, \mathbf{X})$, we can obtain a solution as accurate as we like by dividing the process into sufficiently many steps.

⁴ For a detailed treatment of the linearised approach to AGE modelling, see the Black Book. Chapter 3 contains information about Euler's method and multistep computations.

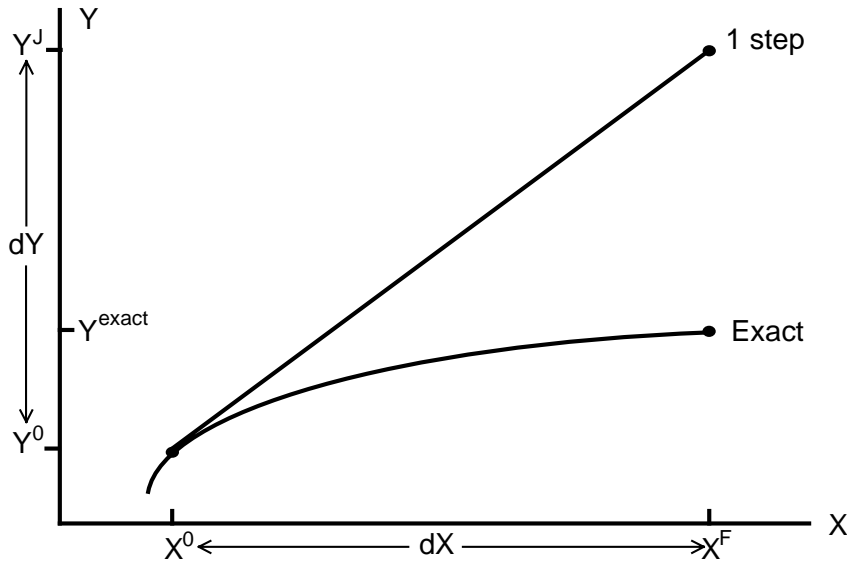


Figure 2. Linearisation error

The technique illustrated in Figure 3, known as the Euler method, is the simplest of several related techniques of numerical integration—the process of using differential equations (change formulae) to move from one solution to another. GEMPACK offers the choice of several such techniques. Each requires the user to supply an initial solution $\{Y^0, X^0\}$, formulae for the derivative matrices G_Y and G_X , and the total percentage change in the exogenous variables, x . The levels functional form, $F(Y, X)$, need not be specified, although it underlies G_Y and G_X .

The accuracy of multistep solution techniques can be improved by extrapolation. Suppose the same experiment were repeated using 4-step, 8-step and 16-step Euler computations, yielding the following estimates for the total percentage change in some endogenous variable Y :

- y(4-step) = 4.5%,
- y(8-step) = 4.3% (0.2% less), and
- y(16-step) = 4.2% (0.1% less).

Extrapolation suggests that the 32-step solution would be:

y(32-step) = 4.15% (0.05% less),

and that the exact solution would be:

y(∞ -step) = 4.1%.

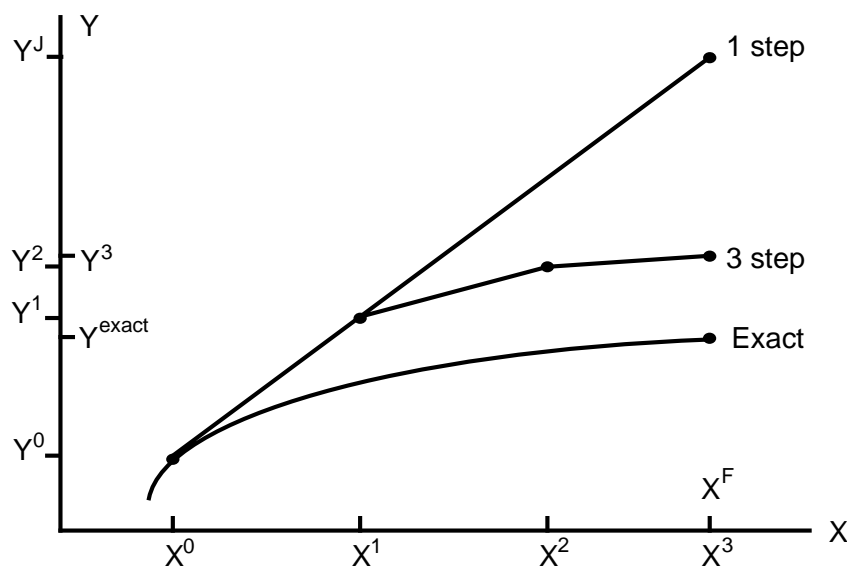


Figure 3. Multistep process to reduce linearisation error

The extrapolated result requires 28 (=4+8+16) steps to compute but would normally be more accurate than that given by a single 28-step computation. Alternatively, extrapolation enables us to obtain given accuracy with fewer steps. As we noted above, each step of a multi-step solution requires: computation from data of the percentage-change derivative matrices \mathbf{G}_Y and \mathbf{G}_X ; solution of the linear system (6); and use of that solution to update the data (\mathbf{X}, \mathbf{Y}) .

In practice, for typical AGE models, it is unnecessary, during a multistep computation, to record values for every element in \mathbf{X} and \mathbf{Y} . Instead, we can define a set of *data coefficients* \mathbf{V} , which are functions of \mathbf{X} and \mathbf{Y} , i.e., $\mathbf{V} = \mathbf{H}(\mathbf{X}, \mathbf{Y})$. Most elements of \mathbf{V} are simple cost or expenditure flows such as appear in input-output tables. \mathbf{G}_Y and \mathbf{G}_X turn out to be simple functions of \mathbf{V} ; often indeed identical to elements of \mathbf{V} . After each small change, \mathbf{V} is updated using the formula $\mathbf{v} = \mathbf{H}_Y(\mathbf{X}, \mathbf{Y})\mathbf{y} + \mathbf{H}_X(\mathbf{X}, \mathbf{Y})\mathbf{x}$. The advantages of storing \mathbf{V} , rather than \mathbf{X} and \mathbf{Y} , are twofold:

- the expressions for \mathbf{G}_Y and \mathbf{G}_X in terms of \mathbf{V} tend to be simple, often far simpler than the original \mathbf{F} functions; and
- there are fewer elements in \mathbf{V} than in \mathbf{X} and \mathbf{Y} (e.g., instead of storing prices and quantities separately, we store merely their products, the values of commodity or factor flows).

3.1. Levels and linearised systems compared: a small example

To illustrate the convenience of the linear approach⁵, we consider a very small equation system: the CES input demand equations for a producer who makes output Z from N inputs X_k , $k=1-N$, with prices P_k . In the levels the equations are (see Appendix A):

$$X_k = Z \delta_k^{1/(\rho+1)} \left[\frac{P_k}{P_{ave}} \right]^{-1/(\rho+1)}, \quad k=1, N \quad (8)$$

$$\text{where } P_{ave} = \left(\sum_{i=1}^N \delta_i^{1/(\rho+1)} P_i^{\rho/(\rho+1)} \right)^{(\rho+1)/\rho}. \quad (9)$$

The δ_k and ρ are behavioural parameters. To solve the model in the levels, the values of the δ_k are normally found from historical flows data, $V_k = P_k X_k$, presumed consistent with the equation system and with some externally given value for ρ . This process is called calibration. To fix the X_k , it is usual to assign arbitrary values to the P_k , say 1. This merely sets convenient units for the X_k (base-period-dollars-worth). ρ is normally given by econometric estimates of the elasticity of substitution, $\sigma (=1/(\rho+1))$. With the P_k , X_k , Z and ρ known, the δ_k can be deduced.

In the solution phase of the levels model, δ_k and ρ are fixed at their calibrated values. The solution algorithm attempts to find P_k , X_k and Z consistent with the levels equations and with other exogenous restrictions. Typically this will involve repeated evaluation of both (8) and (9)—corresponding to $\mathbf{F}(\mathbf{Y}, \mathbf{X})$ —and of derivatives which come from these equations—corresponding to \mathbf{F}_Y and \mathbf{F}_X .

The percentage-change approach is far simpler. Corresponding to (8) and (9), the linearised equations are (see Appendices A and E):

$$x_k = z - \sigma (p_k - p_{ave}), \quad k=1, N \quad (10)$$

$$\text{and } p_{ave} = \sum_{i=1}^N S_i p_i, \quad \text{where the } S_i \text{ are cost shares, eg, } S_i = V_i / \sum_{k=1}^N V_k \quad (11)$$

Since percentage changes have no units, the calibration phase—which amounts to an arbitrary choice of units—is not required. For the same reason the δ_k parameters do not appear. However, the flows data V_k again form the starting point. After each change they are updated by:

$$V_{k,new} = V_{k,old} + V_{k,old}(x_k + p_k)/100 \quad (12)$$

GEMPACK is designed to make the linear solution process as easy as possible. The user specifies the linear equations (10) and (11) and the update formulae (12) in the TABLO language—which resembles algebraic notation. Then GEMPACK repeatedly:

⁵ For a comparison of the levels and linearised approaches to solving AGE models see Hertel, Horridge & Pearson (1992).

- evaluates \mathbf{G}_Y and \mathbf{G}_X at given values of \mathbf{V} ;
- solves the linear system to find \mathbf{y} , taking advantage of the sparsity of \mathbf{G}_Y and \mathbf{G}_X ; and
- updates the data coefficients \mathbf{V} .

The housekeeping details of multistep and extrapolated solutions are hidden from the user.

Apart from its simplicity, the linearised approach has two further advantages.

- It allows free choice of which variables are to be exogenous or endogenous. Many levels algorithms do not allow this flexibility.
- To reduce AGE models to manageable size, it is often necessary to use model equations to substitute out matrix variables of large dimensions. In a linear system, we can always make any variable the subject of any equation in which it appears. Hence, substitution is a simple mechanical process. In fact, because GEMPACK performs this routine algebra for the user, the model can be specified in terms of its original behavioural equations, rather than in a reduced form. This reduces the potential for error and makes model equations easier to check.

3.2. The initial solution

Our discussion of the solution procedure has so far assumed that we possess an initial solution of the model— $\{\mathbf{Y}^0, \mathbf{X}^0\}$ or the equivalent \mathbf{V}^0 —and that results show percentage deviations from this initial state.

In practice, the PHILGEM database does not, like B in Figure 1, show the expected state of the economy at a future date. Instead the most recently available historical data, A, are used. At best, these refer to the present-day economy. Note that, for the atemporal static model, A provides a solution for period T. In the static model, setting all exogenous variables at their base-period levels would leave all the endogenous variables at their base-period levels. Nevertheless, A may not be an empirically plausible control state for the economy at period T and the question therefore arises: are estimates of the B-to-C percentage changes much affected by starting from A rather than B? For example, would the percentage effects of a tariff cut inflicted in 1994 differ much from those caused by a 2005 cut? Probably not. First, balanced growth, i.e., a proportional enlargement of the model database, just scales equation coefficients equally; it does not affect PHILGEM results. Second, compositional changes, which do alter percentage-change effects, happen quite slowly. So for short- and medium-run simulations A is a reasonable proxy for B, (Dixon, Parmenter and Rimmer, 1986).⁶

4. The Equations of PHILGEM

In this section we provide a formal description of the linear form of the model. Our description is organised around the TABLO file which implements the model in GEMPACK. We present the complete text of the TABLO Input file divided into a sequence of excerpts and supplemented by tables, figures and explanatory text.

The TABLO language in which the file is written is essentially conventional algebra, with names for variables and coefficients chosen to be suggestive of their economic interpretations. Some practice is required for readers to become familiar with the TABLO notation but it is no more complex than alternative means of setting out the model—the notation employed in DPSV (1982), for example. Acquiring the familiarity allows ready access to the GEMPACK programs used to conduct simulations with the model and to convert the results to human-readable form. Both the input and the output of these programs employ the TABLO notation. Moreover, familiarity with the TABLO format is essential for users who may wish to make modifications to the model's structure.

⁶ We claim here that, for example, the estimate that a reduction in the textile tariff would reduce textile employment 5 years hence by, say, 7%, is not too sensitive to the fact that our simulation started from today's database rather than a database representing the economy in 5 years time. Nevertheless, the social implications of a 7% employment loss depend closely on whether textile employment is projected to grow in the absence of any tariff cut. To examine this question we need a forecasting model, such as MONASH. If a MONASH control scenario had textile employment grow annually by 1.5%, the 7% reduction could be absorbed without actually firing any textile workers.

Another compelling reason for using the TABLO Input file to document the model is that it ensures that our description is complete and accurate: complete because the only other data needed by the GEMPACK solution process is numerical (the model's database and the exogenous inputs to particular simulations); and accurate because GEMPACK is nothing more than an equation solving system, incorporating no economic assumptions of its own.

We continue this section with a short introduction to the TABLO language—other details may be picked up later, as they are encountered. Then we describe the input-output database which underlies the model. This structures our subsequent presentation.

4.1. The TABLO language

The TABLO model description defines the percentage-change equations of the model. For example, the CES demand equations, (10) and (11), would appear as:

```
Equation E_x # input demands #
  (all, f, FAC) x(f) = z - SIGMA*[p(f) - p_f];
Equation E_p_f # input cost index #
  V_F*p_f = sum{f,FAC, V(f)*p(f)};
```

The first word, 'Equation', is a keyword which defines the statement type. Then follows the identifier for the equation, which must be unique. The descriptive text between '#' symbols is optional—it appears in certain report files. The expression '(all, f, FAC)' signifies that the equation is a matrix equation, containing one scalar equation for each element of the set FAC.⁷

Within the equation, the convention is followed of using lower-case letters for the percentage-change variables (x , z , p and p_f), and upper case for the coefficients (SIGMA, V and V_F). Since GEMPACK ignores case, this practice assists only the human reader. An implication is that we cannot use the same sequence of characters, distinguished only by case, to define a variable and a coefficient. The '(f)' suffix indicates that variables and coefficients are vectors, with elements corresponding to the set FAC. A semi-colon signals the end of the TABLO statement.

To facilitate portability between computing environments, the TABLO character set is quite restricted—only alphanumeric characters and a few punctuation marks may be used. The use of Greek letters and subscripts is precluded, and the asterisk, '*', must replace the multiplication symbol '×'.

Sets, coefficients and variables must be explicitly declared, *via* statements such as:

```
Set FAC # inputs # (capital, labour, energy);
Coefficient
  (all,f,FAC) V(f) # cost of inputs #;
      V_F # total cost #;
      SIGMA # substitution elasticity #;
Variable
  (all,f,FAC) p(f) # price of inputs #;
  (all,f,FAC) x(f) # demand for inputs #;
      z # output #;
      p_f # input cost index #;
```

As the last two statements in the 'Coefficient' block and the last three in the 'Variable' block illustrate, initial keywords (such as 'Coefficient' and 'Variable') may be omitted if the previous statement was of the same type.

Coefficients must be assigned values, either by reading from file:

```
Read V from file FLOWDATA;
Read SIGMA from file PARAMS;
```

or in terms of other coefficients, using formulae:

```
Formula V_F = sum{f, FAC, V(f)}; ! used in cost index equation !
```

⁷ For equation E_x we could have written: (all, j, FAC) x(j) = z - SIGMA*[p(j) - p_f], without affecting simulation results. Our convention that the index, (f), be the same as the initial letter of the set it ranges over, aids comprehension but is not enforced by GEMPACK. By contrast, GAMS (a competing software package) enforces consistent usage of set indices by rigidly connecting indices with the corresponding sets.

The right hand side of the last statement employs the TABLO summation notation, equivalent to the Σ notation used in standard algebra. It defines the sum over an index f running over the set FAC of the input-cost coefficients, $V(f)$. The statement also contains a comment, i.e., the text between exclamation marks (!). TABLO ignores comments.

Some of the coefficients will be updated during multistep computations. This requires the inclusion of statements such as:

```
Update (all, f, FAC) V(f) = x(f)*p(f);
```

which is the default update statement, causing $V(f)$ to be increased after each step by $[x(f) + p(f)]\%$, where $x(f)$ and $p(f)$ are the percentage changes computed at the previous step.

The sample statements listed above introduce most of the types of statement required for the model. But since all sets, variables and coefficients must be defined before they are used, and since coefficients must be assigned values before appearing in equations, it is necessary for the order of the TABLO statements to be almost the reverse of the order in which they appear above. The PHILGEM TABLO Input file is ordered as follows:

- definition of sets;
- declarations of variables;
- declarations of often-used coefficients which are read from files, with associated Read and Update statements;
- declarations of other often-used coefficients which are computed from the data, using associated Formulae; and
- groups of topically-related equations, with some of the groups including statements defining coefficients which are used only within that group.

4.2. The naming system

PHILGEM's naming convention is the same as ORANI-G. The TABLO Input file defines a multitude of variables and coefficients that are used in the model's equations. It can be difficult to remember the names of all these variables and coefficients⁸. Fortunately, their names follow a pattern. Although GEMPACK does not require that names conform to any pattern, we find that systematic naming reduces the burden on (human) memory. As far as possible, names for variables and coefficients conform to a system in which each name consists of 2 or more parts, as follows:

- first, a letter or letters indicating the type of variable, for example,
- a technical change
 - del ordinary (rather than percentage) change
 - f shift variable
 - H indexing parameter
 - p price, local currency
 - pf price, foreign currency
 - S input share
 - SIGMA elasticity of substitution
 - t tax
 - V levels value, local currency
 - w percentage-change value, local currency
 - x input quantity;
- second, one of the digits 0 to 6 indicating user, that is,
- 1 current production
 - 2 investment
 - 3 consumption

⁸ GEMPACK's TABmate editor offers some comfort to the forgetful. With the TABLO Input file open in TABmate, you may click on any variable or coefficient name, then click the Gloss button. A list will appear, starting with a description of that variable and then showing all statements in the TABLO Input file where it is used.

- 4 export
 5 government
 6 inventories
 0 all users, or user distinction irrelevant;
 third (optional), three or more letters giving further information, for example,
- | | |
|------|--|
| bas | (often omitted) basic—not including margins or taxes |
| cap | capital |
| cif | imports at border prices |
| imp | imports (duty paid) |
| lab | labour |
| lnd | land |
| lux | linear expenditure system (supernumerary part) |
| mar | margins |
| oct | other cost tickets |
| prim | all primary factors (land, labour or capital) |
| pur | at purchasers' prices |
| sub | linear expenditure system (subsistence part) |
| tar | tariffs |
| tax | indirect taxes |
| tot | total or average over all inputs for some user |
| hou | households |
| gov | government |
| ent | enterprises |
| row | rest of the world |
| gos | gross operating surplus; |

fourth (optional), an underscore character, indicating that this variable is an aggregate or average, with subsequent letters showing over which sets the underlying variable has been summed or averaged, for example,

- _c over COM (commodities),
 _s over SRC (dom + imp),
 _i over IND (industries),
 _io over IND and OCC (skills).

Although GEMPACK does not distinguish between upper and lower case, we use:

- lower case for variable names and set indices;
 upper case for set and coefficient names; and
 initial letter upper case for TABLO keywords.

4.3. The model's data base

Figure 4 is a schematic representation of PHILGEM's input-output table database. It reveals the basic structure of the model. The column headings in the main part of the figure (an absorption matrix or Supply-Use Table) identify the following demanders:

- (1) domestic producers divided into I industries;
- (2) investors divided into I industries;
- (3) a single representative household;
- (4) an aggregate foreign purchaser of exports;
- (5) government demands; and
- (6) changes in inventories.

The entries in each column show the structure of the purchases made by the agents identified in the column heading. Each of the C commodity types identified in the model can be obtained locally or imported from overseas. The source-specific commodities are used by industries as inputs to current production and capital formation, are consumed by households and governments, are exported, or are added

to or subtracted from inventories. Only domestically produced goods appear in the export column. M of the domestically produced goods are used as margins services (wholesale and retail trade, and transport) which are required to transfer commodities from their sources to their users. Commodity taxes are payable on the purchases. As well as intermediate inputs, current production requires inputs of three categories of primary factors: labour (divided into O occupations), fixed capital, and agricultural land. Production taxes include output taxes or subsidies that are not user-specific. The 'other costs' category covers various miscellaneous taxes on firms, such as municipal taxes or charges.

		Absorption Matrix					
		1	2	3	4	5	6
		Producers	Investors	Household	Export	Government	Change in Inventories
Size		← I →	← I →	← 1 →	← 1 →	← 1 →	← 1 →
Basic Flows	↑ C×S ↓	V1BAS	V2BAS	V3BAS	V4BAS	V5BAS	V6BAS
Margins	↑ C×S×M ↓	V1MAR	V2MAR	V3MAR	V4MAR	V5MAR	n/a
Taxes	↑ C×S ↓	V1TAX	V2TAX	V3TAX	V4TAX	V5TAX	n/a
Labour	↑ O ↓	V1LAB	C = Number of Commodities I = Number of Industries S = 2: Domestic, Imported, O = Number of Occupation Types M = Number of Commodities used as Margins				
Capital	↑ 1 ↓	V1CAP					
Land	↑ 1 ↓	V1LND					
Production Tax	↑ 1 ↓	V1PTX					
Other Costs	↑ 1 ↓	V1OCT					

		Joint Production Matrix	
Size		←	→
↑	C	MAKE	
↓			

		Import Duty	
Size		←	→
↑	C	V0TAR	
↓			

Figure 4. The PHILGEM Flows Database

Each cell in the illustrative absorption matrix in Figure 4 contains the name of the corresponding data matrix. For example, V2MAR is a 4-dimensional array showing the cost of M margins services on the flows of C goods, both domestically produced and imported (S), to I investors.

In principle, each industry is capable of producing any of the C commodity types. The MAKE matrix at the bottom of Figure 4 shows the value of output of each commodity by each industry. Finally, tariffs on imports are assumed to be levied at rates which vary by commodity but not by user. The revenue obtained is represented by the tariff vector VOTAR.

Figure 5 presents a schematic representation of the entire database of PHILGEM in the form of a Social Accounting Matrix (SAM), while Figure 6 presents the value flows of the SAM which is based on the Input-Output Table and National Accounts data (NSCB 2006), both for the year 2000.

A SAM is an integrated framework that records all transactions in an economy in a given year (Round, 2003). It provides information on the economic and social structure; illuminates the interaction of various agents; and captures economic flows at both micro and macro levels.

A SAM tracks how income is generated and distributed. It is a square matrix, composed of different accounts, with entries along the rows representing receipts while column entries track expenditures. A SAM adheres to double entry accounting in which a flow is both recorded as a receipt and an expense. A residual savings row allows the row sum for each account to equal the corresponding column sum. The Philippine SAM shown in Figure 5 is based on a combination of data coming from the input-output table, national income and product accounts, household survey and the labour force survey.

Figure 5 reveals that the first 8 rows of the SAM correspond to the Input-Output table database shown in Figure 4, while entries found along the intersection of government row and column accounts representing commodity tax, production tax and tariff are computed directly. On the other hand, cells shaded in gray represent data drawn elsewhere or not found in the input-output table.

Entries in the SAM are named based on the row and column in which they appear. For example, VHOUGOS represents the value of household income from gross operating surplus (or capital), while VTAXENT corresponds to the value of direct income taxes paid by enterprises. VGOVROW shows the value of foreign aid received by the government whereas VGOVENT displays the value of dividend income received by government from enterprises which include both public and private corporations. Section 5 contains more details of the SAM and associated equations and variables.

The capital income shares of these agents are sourced from the national income accounts for the year 2000 (NSCB 2006). In turn, each household's share in total capital income earned by all households in the economy is taken from the 2003 household survey, known locally as the family income and expenditure survey (NSO 2003).

Depending on data availability, additional flows could be introduced in the SAM. For example, flows representing labour payments from the rest of the world can be added on the intersection of labour row and rest of the world row column.

		1 Industries	2 Domestic Commodities	3 Imported commodities	4 Labour	5 Capital	6 Production Tax	7 Commodity Tax	8 Tariff
		← I →	← C →	← C →	← O →	← K →	← I →	← I →	← C →
1 Industries	↑ I ↓		MAKE						
2 Domestic Commodities	↑ C ↓	V1BAS("dom") + V1MAR							
3 Imported commodities	↑ C ↓	V1BAS("imp")							
4 Labour	↑ O ↓	V1LAB							
5 Capital	↑ K ↓	V1CAP + V1LND							
6 Production Tax	↑ I ↓	V1PTX + V1OCT							
7 Commodity Tax	↑ I ↓	V1TAX							
8 Tariff	↑ C ↓			V0TAR					
9 Direct Tax	↑ I ↓								
10 Households	↑ H ↓				V1LAB_I	VHOUGOS			
11 Enterprises	↑ I ↓					VENTGOS			
12 Government	↑ I ↓					VGOVGOS	V1PTX + V1OCT	V0TAX	V0TAR
13 Government Investment	↑ I ↓								
14 Savings	↑ I ↓								
15 Stocks	↑ I ↓								
16 Rest of the World	↑ I ↓			V0CIF					
17 Total	↑ I ↓	Output	Supply of domestic Commodities	Supply of Imported Commodities	Wage Costs	Cost of Capital	Production Tax	Commodity Tax	Tariff

Legend: I – No. of Industries; C – No. of Commodities; O – No. of Occupation Types; K – No. of Types of Capital; H – No. of Household Types.

Figure 5. PHILGEM Aggregate Social Accounting Matrix (SAM) Database

9 Direct Tax	10 Households	11 Enterprises	12 Government	13 Government Investment	14 Private Investment	15 Stocks	16 Rest of the World	17 Total
← I →	← H →	← I →	← I →	← I →	← I →	← I →	← I →	← I →
								Sales
	V3BAS("dom") + V3MAR		V5BAS("dom") + V5MAR	V2BAS("dom") + V2MAR	V2BAS("dom") + V2MAR	V6BAS("dom")	V4BAS("dom") + V4MAR	Demand for Domestic Commodities
	V3BAS("imp")		V5BAS("imp")	V2BAS("imp")	V2BAS("imp")	V6BAS("imp")		Demand for Imported Commodities
								Wage Income
								Capital Income
								Production Tax
	V3TAX		V5TAX	V2TAX	V2TAX		V4TAX	Commodity Tax
								Tariff
	VTAXHOU	VTAXENT						Income Tax
	VHOUHOU	VHOUENT	VHOUGOV				VHOUROW	Household Income
	VENTHOU						VENTROW	Enterprises' Income
VTAXHOU + VTAXENT	VGOVHOU	VGOVENT					VGOWROW	Government Income
			VGOVINV					Government Investment
	VSAVHOU	VSAVENT	VSAVGOV				VSAVROW	Savings
					VSTKINV			Stocks
		VROWENT	VROWGOV					Foreign Exchange Receipts
Income Tax	Household Expenditures	Enterprises' Expenditure	Government Expenditure	Government Investment	Private Investment	Stocks	Foreign Exchange Receipts	

Note: Shaded cells represent additional data from the SAM (i.e., not found in Figure 4).

Figure 5 (continued) . PHILGEM Aggregate Social Accounting Matrix (SAM) Database

		1 Industries	2 Domestic Commodities	3 Imported commodities	4 Labour	5 Capital	6 Production Tax	7 Commodity Tax	8 Tariff
		← I →	← C →	← C →	← O →	← K →	← I →	← I →	← C →
1 Industries	↑ I ↓		7,795,925						
2 Domestic Commodities	↑ C ↓	2,720,240							
3 Imported commodities	↑ C ↓	1,119,442							
4 Labour	↑ O ↓	1,338,209							
5 Capital	↑ K ↓	2,543,469							
6 Production Tax	↑ I ↓	1,733							
7 Commodity Tax	↑ I ↓	72,831							
8 Tariff	↑ C ↓			91,246					
9 Direct Tax	↑ I ↓								
10 Households	↑ H ↓				1,338,209	1,653,543			
11 Enterprises	↑ I ↓					875,259			
12 Government	↑ I ↓					14,667	1,733	182,359	91,246
13 Government Investment	↑ I ↓								
14 Savings	↑ I ↓								
15 Stocks	↑ I ↓								
16 Rest of the World	↑ I ↓			1,696,826					
17 Total	↑ I ↓	7,795,925	7,795,925	1,788,072	1,338,209	2,543,469	1,733	182,359	91,246

Legend: I – No. of Industries; C – No. of Commodities; O – No. of Occupation Types; K – No. of Types of Capital; H – No. of Household Types.

Figure 6. PHILGEM Aggregate SAM (in millions of pesos)

9 Direct Tax	10 Households	11 Enterprises	12 Government	13 Government Investment	14 Private Investment	15 Stocks	16 Rest of the World	17 Total
← I →	← H →	← I →	← I →	← I →	← I →	← I →	← I →	← I →
								7,795,925
	2,541,153		437,149	268,847	188,654	185,078	1,454,804	7,795,925
	284,121		796	84,680	59,421	239,612		1,788,072
								1,338,209
								2,543,469
								1,733
	90,420		1,181	3,071	2,155		12,702	182,359
								91,246
	81,622	113,932						195,554
	0	65,242	175,606				17,224	3,249,824
	40,994						78,289	994,542
195,554	98,768	44,336					5,011	633,674
			356,598					356,598
	112,747	676,720	-392,680				278,134	674,921
					424,690			424,690
		94,312	55,025					1,846,163
195,554	3,249,824	994,542	633,674	356,598	674,921	424,690	1,846,163	

Note: Shaded cells represent additional data from the SAM (i.e., not found in Figure 4).

Figure 6. (continued). PHILGEM Aggregate Social Accounting Matrix (SAM) Database

4.4. Dimensions of the model

Excerpt 1 of the TABLO Input file begins by defining logical names for input and output files. Initial data are stored in the BASEDATA input file. The SUMMARY output file is used to store summary and diagnostic information. Note that BASEDATA and SUMMARY are *logical* names. The actual locations of these files (disk, folder, filename) are chosen by the model user.

The rest of Excerpt 1 defines sets: lists of descriptors for the components of vector variables. Set names appear in upper-case characters. For example, the first Set statement is to be read as defining a set named 'COM' which contains commodity descriptors. The elements of COM (a list of commodity names) are read from the input file (this allows the model to use databases with different numbers of sectors). By contrast the two elements of the set SRC—dom and imp—are listed explicitly.

```

! Excerpt 1 of TABLO input file: !
! Files and sets !
File BASEDATA # Input data file #;
(new) SUMMARY # Output for summary and checking data #;

Set                                                    !Index!
COM # Commodities # read elements from file BASEDATA header "COM";      ! c !
SRC # Source of commodities # (dom,imp);                                ! s !
IND # Industries # read elements from file BASEDATA header "IND";        ! i !
OCC # Occupations # read elements from file BASEDATA header "OCC";       ! o !
MAR # Margin commodities # read elements from file BASEDATA header "MAR"; ! m !
HOU # Household # read elements from file BASEDATA header "HOU";         ! m !

Subset MAR is subset of COM;
Set NONMAR # Non-margin commodities # = COM - MAR;                       ! n !

```

The commodity, industry, occupational, and household classifications of PHILGEM described here are aggregates of the classifications used in the original version of PHILGEM, which had 240 industries and commodities, 10 labour occupations, and 42,092 households.

The industry classification differs slightly from the commodity classification. Both are listed in Table 1. In this aggregated version of the model, multiproduction is confined to the first industry (RiceCorn), which produce the first two commodities (Paddy rice and Corn). Each of the remaining industries produces a unique commodity. Labour is disaggregated into skill-based occupational categories described by the set OCC.

The central column of Table 1 lists the elements of the set COM which are read from file. GEMPACK uses the element names to label the rows and columns of results and data tables. The element names cannot be more than 12 letters long, nor contain spaces. The IND elements are the same as elements 2-25 of COM. The representative households are IThe set HOU contains elements sh the representative households by demographic charac are also read from file

Elements of the set MAR are margins commodities, i.e., they are required to facilitate the flows of other commodities from producers (or importers) to users. Hence, the costs of margins services, together with indirect taxes, account for differences between *basic* prices (received by producers or importers) and *purchasers'* prices (paid by users).

TABLO does not prevent elements of two sets from sharing the same name; nor, in such a case, does it automatically infer any connection between the corresponding elements. The Subset statement which follows the definition of the set MAR is required for TABLO to realize that the two elements of MAR, Trade and Transport, are the same as the 19th and 20th elements of the set COM.

The statement for NONMAR defines that set as a complement. That is, NONMAR consists of all those elements of COM which are not in MAR. In this case TABLO is able to deduce that NONMAR must be a subset of COM.

Table 1 Commodity and Industry Classification

Commodity Description		Elements of Set COM	Industry Description	
1	Paddy rice	Paddy		
2	Corn	Corn	1	Paddy rice and Corn
3	Fruits and vegetables	FruitsVege	2	Fruits and vegetables
4	Other crops	OtherCrops	3	Other crops
5	Livestock and poultry	LvstkPoultry	4	Livestock and poultry
6	Other agriculture	OtherAgric	5	Other agriculture
7	Mining	Mining	6	Mining
8	Processed food	ProcFood	7	Processed food
9	Rice, corn, sugar milling	Milling	8	Rice, corn, sugar milling
10	Tobacco and alcohol	TobacAlchl	9	Tobacco and alcohol
11	Textile, garments and footwear	TextGarmFoot	10	Textile, garments and footwear
12	Metal products	OtherManuf	11	Metal products
13	Transport equipment	PetroChem	12	Transport equipment
14	Other machinery	Metals	13	Other machinery
15	Other manufacturing	Machines	14	Other manufacturing
16	Electric related appliances	ElecRelAppli	15	Electric related appliances
17	Semiconductors	Semicon	16	Semiconductors
18	Utilities	Utilities	17	Utilities
19	Retail & wholesale trade	Trade (Margin)	18	Retail & wholesale trade
20	Transport	Transport (Margin)	19	Transport
21	Communication	Communicatn	20	Communication
22	Construction	Construction	21	Construction
23	Ownership of dwellings	Dwellings	22	Ownership of dwellings
24	Public services	PublicSrvcs	23	Public services
25	Private services	PrivateSrvcs	24	Private services

The next statement relates to the household set. Initially, the elements of the set GENDER (female and male) are listed explicitly while the set DEC is specified to contain 10 ordered elements representing income deciles 1 to 10. On the other hand, a new set HOU is generated as a set (Cartesian) product of the sets GENDER and DEC, resulting in HOU containing all possible ordered pairs coming from the sets GENDER and DEC respectively. Accordingly, set HOU would now contain elements: female_dec01, male_dec01, female_dec02, male_dec02, ..., female_dec10, male_dec10. Note that under this operation, the elements of the set GENDER vary faster while those of the set DEC vary slower; and a joining character "_" that distinguishes the elements of the first and second set is automatically created (see section 9.1.4 of GEMPACK user manual for more details).

4.5. Core data coefficients and related variables

The next excerpts of the TABLO file contains statements indicating coefficient names data to be read from file. The data items defined in these statements appear as coefficients in the model's equations. The statements define coefficient names (which all appear in upper-case characters), the locations from which the data are to be read, variable names (in lower-case), and formulae for the data updates which are necessary in computing multi-step solutions to the model (see Section 3).

4.5.1. Basic flows

The excerpts group the data according to the rows of Figure. Thus, Excerpt 2 begins by defining coefficients representing the basic commodity flows corresponding to row 1 (direct flows) of the figure, i.e., the flow matrices V1BAS, V2BAS, and so on. Preceding the coefficient names are their dimensions, indicated using the "all" qualifier and the sets defined in Excerpt 1. For example, the first 'Coefficient' statement defines a data item V1BAS(c,s,i) which is the basic value (indicated by 'BAS') of a flow of intermediate inputs (indicated by 'I') of commodity c from source s to user industry i. The first 'Read' statement indicates that this data item is stored on file BASEDATA with header '1BAS'. (A GEMPACK data file consists of a number of data items such as arrays of real numbers. Each data item is identified by a unique key or 'header').

Each of these flows is the product of a price and a quantity. The excerpt goes on to define these variables. Unless otherwise stated, all variables are percentage changes—to indicate this, their names

appear in lower-case letters. Preceding the names of the variables are their dimensions, indicated using the sets defined in Excerpt 1. For example, the first variable statement defines a matrix variable x_1 (indexed by commodity, source, and using industry) the elements of which are percentage changes in the direct demands by producers for source-specific intermediate inputs. This is the quantity variable corresponding to V1BAS.

The last in the group of quantity variables, $delx_6$, is preceded by the 'Change' qualifier to indicate that it is an ordinary (rather than percentage) change. Changes in inventories may be either positive or negative. Our multistep solution procedure requires that large changes be broken into a sequence of small changes. However, no sequence of small *percentage* changes allows a number to change sign—at least one change must exceed -100%. Thus, for variables that may, in the levels, change sign, we prefer to use ordinary changes. The names of ordinary change variables often start with the letters "del".

Next come two price variables. A matrix variable p_0 (indexed by commodity and source), shows percentage changes in the basic prices which are common to all local users. These basic prices do not include the cost of margins and taxes. Exports have their own basic prices, pe . Potentially, the pe could be different from the domestic part of p_0 ⁹.

```
! Excerpt 2 of TABLO input file: !
! Data coefficients and variables relating to basic commodity flows !
Coefficient ! Basic flows of commodities (excluding margin demands)!
(all,c,COM)(all,s,SRC)(all,i,IND) V1BAS(c,s,i) # Intermediate basic flows #;
(all,c,COM)(all,s,SRC)(all,i,IND) V2BAS(c,s,i) # Investment basic flows #;
      (all,c,COM)(all,s,SRC) V3BAS(c,s) # Household basic flows #;
      (all,c,COM) V4BAS(c) # Export basic flows #;
(all,c,COM)(all,s,SRC) V5BAS(c,s) # Government basic flows #;
(all,c,COM)(all,s,SRC) V6BAS(c,s) # Inventories basic flows #;

Read
V1BAS from file BASEDATA header "1BAS";
V2BAS from file BASEDATA header "2BAS";
V3BAS from file BASEDATA header "3BAS";
V4BAS from file BASEDATA header "4BAS";
V5BAS from file BASEDATA header "5BAS";
V6BAS from file BASEDATA header "6BAS";

Variable ! Variables used to update above flows !
(all,c,COM)(all,s,SRC)(all,i,IND) x1(c,s,i) # Intermediate basic demands #;
(all,c,COM)(all,s,SRC)(all,i,IND) x2(c,s,i) # Investment basic demands #;
      (all,c,COM)(all,s,SRC) x3(c,s) # Household basic demands #;
      (all,c,COM) x4(c) # Export basic demands #;
      (all,c,COM)(all,s,SRC) x5(c,s) # Government basic demands #;
(change)(all,c,COM)(all,s,SRC) delx6(c,s) # Inventories demands #;
      (all,c,COM)(all,s,SRC) p0(c,s) # Basic prices for local users #;
      (all,c,COM) pe(c) # Basic price of exportables #;
(change)(all,c,COM)(all,s,SRC) delV6(c,s) # Value of inventories #;

Update
(all,c,COM)(all,s,SRC)(all,i,IND) V1BAS(c,s,i) = p0(c,s)*x1(c,s,i);
(all,c,COM)(all,s,SRC)(all,i,IND) V2BAS(c,s,i) = p0(c,s)*x2(c,s,i);
      (all,c,COM)(all,s,SRC) V3BAS(c,s) = p0(c,s)*x3(c,s);
      (all,c,COM) V4BAS(c) = pe(c)*x4(c);
      (all,c,COM)(all,s,SRC) V5BAS(c,s) = p0(c,s)*x5(c,s);
(change)(all,c,COM)(all,s,SRC) V6BAS(c,s) = delV6(c,s);
```

Finally, the variable $delV_6$ is used in the update statements which appear next.

⁹ Exports (V4BAS) are valued with price vector pe . Unless we activate the optional CET transformation between goods destined for export and those for local use, the pe are identical to the domestic part of p_0 . See Excerpt 13.

The first 'Update' statement indicates that the flow V1BAS(c,s,i) should be updated using the default update formula, which is used for a data item which is a product of two (or more) of the model's variables. For an item of the form $V = PX$, the formula for the updated value V^U is:

$$\begin{aligned} V^U &= V^0 + \Delta(PX) = V^0 + X^0\Delta P + P^0\Delta X \\ &= V^0 + P^0X^0\left(\frac{\Delta P}{P^0} + \frac{\Delta X}{X^0}\right) = V^0 + V^0\left(\frac{p}{100} + \frac{x}{100}\right) \end{aligned} \quad (13)$$

where V^0 , P^0 and X^0 are the pre-update values, and p and x are the percentage changes of the variables P and X . For the data item V1BAS(c,s,i) the relevant percentage-change variables are $p0(c,s)$ (the basic-value price of commodity c from source s) and $x1(c,s,i)$ (the demand by user industry i for intermediate inputs of commodity c from source s).

Not all of the model's data items are amenable to update *via* default Updates. For example, the inventories flows, V6BAS, might change sign, and so must not be updated with percentage change variables. In such a case, the Update statement must contain an explicit formula for the ordinary change in the data item: this is indicated by the word 'Change' in parentheses. For V6BAS we represent the change by an ordinary-change variable, delV6. The Update formula (13) then becomes simply:

$$V^U = V^0 + \Delta V. \quad (14)$$

An equation defining the delV6 variable appears later on.

4.5.2. Margin flows

The coefficients and variables of Excerpt 3 are associated with row 2 (margins) of Figure 4, i.e., the flow matrices V1MAR, V2MAR, and so on. These are the quantities of retail and wholesale services or transport needed to deliver each basic flow to the user. For example V3MAR(c,s,m) is the value of margin type m used to deliver commodity type c from source s to households (user 3). The model assumes that margin services are domestically produced and are valued at basic prices—represented by the variable $p0dom$, which (we shall see later) is simply a synonym for the domestic part of the basic price matrix, $p0$ [i.e., $p0dom(c) = p0(c, "dom")$].

```
! Excerpt 3 of TABLO input file: !
! Data coefficients and variables relating to margin flows !
Coefficient
(all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
                                V1MAR(c,s,i,m) # Intermediate margins #;
(all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
                                V2MAR(c,s,i,m) # Investment margins #;
(all,c,COM)(all,s,SRC)(all,m,MAR) V3MAR(c,s,m) # Households margins #;
                                (all,c,COM)(all,m,MAR) V4MAR(c,m) # Export margins #;
(all,c,COM)(all,s,SRC)(all,m,MAR) V5MAR(c,s,m) # Government margins #;
Read
V1MAR from file BASEDATA header "1MAR";
V2MAR from file BASEDATA header "2MAR";
V3MAR from file BASEDATA header "3MAR";
V4MAR from file BASEDATA header "4MAR";
V5MAR from file BASEDATA header "5MAR";

Variable ! Variables used to update above flows !
(all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
                                x1mar(c,s,i,m)# Intermediate margin demand #;
(all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
                                x2mar(c,s,i,m)# Investment margin demands #;
(all,c,COM)(all,s,SRC)(all,m,MAR) x3mar(c,s,m) # Household margin demands #;
                                (all,c,COM)(all,m,MAR) x4mar(c,m) # Export margin demands #;
(all,c,COM)(all,s,SRC)(all,m,MAR) x5mar(c,s,m) # Government margin demands #;
(all,c,COM) p0dom(c) # Basic price of domestic goods = p0(c,"dom") #;

Update
(all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
                                V1MAR(c,s,i,m) = p0dom(m)*x1mar(c,s,i,m);
(all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
```

```

                                V2MAR(c,s,i,m) = p0dom(m)*x2mar(c,s,i,m);
(all,c,COM)(all,s,SRC)(all,m,MAR) V3MAR(c,s,m)   = p0dom(m)*x3mar(c,s,m);
(all,c,COM)(all,m,MAR)             V4MAR(c,m)     = p0dom(m)*x4mar(c,m);
(all,c,COM)(all,s,SRC)(all,m,MAR) V5MAR(c,s,m)   = p0dom(m)*x5mar(c,s,m);

```

4.5.3. User-specific commodity taxes

Excerpt 4 contains coefficients and variables associated with row 3 (commodity taxes) of Figure 4, i.e., the flow matrices V1TAX, V2TAX, and so on. These all have the same dimensions as the corresponding basic flows.

```

! Excerpt 4 of TABLO input file: !
! Data coefficients and variables relating to commodity taxes !
Coefficient ! Taxes on Basic Flows!
(all,c,COM)(all,s,SRC)(all,i,IND) V1TAX(c,s,i) # Taxes on intermediate #;
(all,c,COM)(all,s,SRC)(all,i,IND) V2TAX(c,s,i) # Taxes on investment #;
(all,c,COM)(all,s,SRC)             V3TAX(c,s)   # Taxes on households #;
(all,c,COM)                         V4TAX(c)     # Taxes on export #;
(all,c,COM)(all,s,SRC)             V5TAX(c,s)   # Taxes on government #;
Read
V1TAX from file BASEDATA header "1TAX";
V2TAX from file BASEDATA header "2TAX";
V3TAX from file BASEDATA header "3TAX";
V4TAX from file BASEDATA header "4TAX";
V5TAX from file BASEDATA header "5TAX";
Variable
(change)(all,c,COM)(all,s,SRC)(all,i,IND) delV1TAX(c,s,i) # Interm tax rev #;
(change)(all,c,COM)(all,s,SRC)(all,i,IND) delV2TAX(c,s,i) # Invest tax rev #;
(change)(all,c,COM)(all,s,SRC)           delV3TAX(c,s)   # H'hold tax rev #;
(change)(all,c,COM)                       delV4TAX(c)     # Export tax rev #;
(change)(all,c,COM)(all,s,SRC)           delV5TAX(c,s)   # Govmnt tax rev #;
Update
(change)(all,c,COM)(all,s,SRC)(all,i,IND) V1TAX(c,s,i) = delV1TAX(c,s,i);
(change)(all,c,COM)(all,s,SRC)(all,i,IND) V2TAX(c,s,i) = delV2TAX(c,s,i);
(change)(all,c,COM)(all,s,SRC)           V3TAX(c,s)   = delV3TAX(c,s);
(change)(all,c,COM)                       V4TAX(c)     = delV4TAX(c);
(change)(all,c,COM)(all,s,SRC)           V5TAX(c,s)   = delV5TAX(c,s);

```

PHILGEM treats commodity taxes in great detail—the tax levied on each basic flow is separately identified. Published input-output tables are usually less detailed—so to construct the initial model database we enforce plausible assumptions: e.g., that all intermediate usage of, say, Coal, is taxed at the same rate. However, the disaggregated data structure still allows us to simulate the effects of commodity-and-user-specific tax changes, such as an increased tax on Coal used by the Iron industry.

The tax flows are updated by corresponding ordinary change variables: equations determining these appear later on.

4.5.4. Factor payments and other flows data

Excerpt 5 of the TABLO Input file corresponds to the remaining rows of Figure 4. There are coefficient matrices for payments to labour, capital, and land, and 2 sorts of production tax. Then are listed the corresponding price and quantity variables. For the production tax V1PTX, the corresponding ordinary change variable delV1PTX is used in a Change Update statement: an equation determining this variable appears later. The other flow coefficients in this group are simply the products of prices and quantities. Hence, they can be updated *via* default Update statements.

Excerpt 5 also defines the import duty vector V0TAR. Treatment of the last item in the flows database, the multiproduction matrix MAKE, showing output of commodities by each industry, is deferred to a later section.

```

! Excerpt 5 of TABLO input file: !
! Data coefficients for primary-factor flows, other industry costs, and tariffs!
Coefficient

```

```

(all,i,IND)(all,o,OCC) V1LAB(i,o) # Wage bill matrix #;
(all,i,IND)           V1CAP(i)   # Capital rentals #;
(all,i,IND)           V1LND(i)   # Land rentals #;
(all,i,IND)           V1PTX(i)   # Production tax #;
(all,i,IND)           V1OCT(i)   # Other cost tickets #;
Read
V1LAB from file BASEDATA header "1LAB";
V1CAP from file BASEDATA header "1CAP";
V1LND from file BASEDATA header "1LND";
V1PTX from file BASEDATA header "1PTX";
V1OCT from file BASEDATA header "1OCT";
Variable
(all,i,IND)(all,o,OCC) x1lab(i,o) # Employment by industry and occupation #;
(all,i,IND)(all,o,OCC) p1lab(i,o) # Wages by industry and occupation #;
(all,i,IND) x1cap(i)           # Current capital stock #;
(all,i,IND) p1cap(i)           # Rental price of capital #;
(all,i,IND) x1lnd(i)           # Use of land #;
(all,i,IND) p1lnd(i)           # Rental price of land #;
(change)(all,i,IND) delV1PTX(i) # Ordinary change in production tax revenue #;
(all,i,IND) x1oct(i)           # Demand for "other cost" tickets #;
(all,i,IND) p1oct(i)           # Price of "other cost" tickets #;
Update
(all,i,IND)(all,o,OCC) V1LAB(i,o) = p1lab(i,o)*x1lab(i,o);
(all,i,IND)           V1CAP(i)   = p1cap(i)*x1cap(i);
(all,i,IND)           V1LND(i)   = p1lnd(i)*x1lnd(i);
(change)(all,i,IND)   V1PTX(i)   = delV1PTX(i);
(all,i,IND)           V1OCT(i)   = p1oct(i)*x1oct(i);

! Data coefficients relating to import duties !
Coefficient (all,c,COM) V0TAR(c) # Tariff revenue #;
Read V0TAR from file BASEDATA header "0TAR";
Variable (all,c,COM) (change) delV0TAR(c) # Ordinary change in tariff revenue #;
Update (change) (all,c,COM) V0TAR(c) = delV0TAR(c);

```

4.5.5. Purchasers' values

Excerpt 6 defines the values at purchasers' prices of the commodity flows identified in Figure 4. These aggregates will be used in several different equation blocks. The definitions use the TABLO summation notation, explained in Section 4.1. For example, the first formula in Excerpt 6 contains the term:

$$\text{sum}\{m, \text{MAR}, V1\text{MAR}(c, s, i, m)\}$$

This defines the sum, over an index m running over the set of margins commodities (MAR), of the input-output data flows $V1\text{MAR}(c, s, i, m)$. This sum is the total value of margins commodities required to facilitate the flow of intermediate inputs of commodity c from source s to user industry i . Adding this sum to the basic value of the intermediate-input flow and the associated indirect tax, gives the purchaser's-price value of the flow.

Next are defined purchasers' price variables, which include basic, margin and tax components. Equations to determine these variables appear in Excerpt 22 below.

```

! Excerpt 6 of TABLO input file: !
! Coefficients and variables for purchaser's prices (basic + margins + taxes) !
Coefficient ! Flows at purchasers prices !
(all,c,COM)(all,s,SRC)(all,i,IND) V1PUR(c,s,i) # Intermediate purch. value #;
(all,c,COM)(all,s,SRC)(all,i,IND) V2PUR(c,s,i) # Investment purch. value #;
(all,c,COM)(all,s,SRC)           V3PUR(c,s)   # Households purch. value #;
(all,c,COM)                       V4PUR(c)    # Export purch. value #;
(all,c,COM)(all,s,SRC)           V5PUR(c,s) # Government purch. value #;
Formula
(all,c,COM)(all,s,SRC)(all,i,IND)
  V1PUR(c,s,i) = V1BAS(c,s,i) + V1TAX(c,s,i) + sum{m, MAR, V1MAR(c,s,i,m)};
(all,c,COM)(all,s,SRC)(all,i,IND)
  V2PUR(c,s,i) = V2BAS(c,s,i) + V2TAX(c,s,i) + sum{m, MAR, V2MAR(c,s,i,m)};
(all,c,COM)(all,s,SRC)

```

$$\begin{aligned}
 & \text{V3PUR}(c,s) = \text{V3BAS}(c,s) + \text{V3TAX}(c,s) + \text{sum}\{m,\text{MAR}, \text{V3MAR}(c,s,m)\}; \\
 & (\text{all},c,\text{COM}) \\
 & \text{V4PUR}(c) = \text{V4BAS}(c) + \text{V4TAX}(c) + \text{sum}\{m,\text{MAR}, \text{V4MAR}(c,m)\}; \\
 & (\text{all},c,\text{COM})(\text{all},s,\text{SRC}) \\
 & \text{V5PUR}(c,s) = \text{V5BAS}(c,s) + \text{V5TAX}(c,s) + \text{sum}\{m,\text{MAR}, \text{V5MAR}(c,s,m)\};
 \end{aligned}$$

Variable ! Purchasers prices !

$$\begin{aligned}
 & (\text{all},c,\text{COM})(\text{all},s,\text{SRC})(\text{all},i,\text{IND}) \text{p1}(c,s,i) \# \text{Purchaser's price, intermediate \#}; \\
 & (\text{all},c,\text{COM})(\text{all},s,\text{SRC})(\text{all},i,\text{IND}) \text{p2}(c,s,i) \# \text{Purchaser's price, investment \#}; \\
 & (\text{all},c,\text{COM})(\text{all},s,\text{SRC}) \text{p3}(c,s) \# \text{Purchaser's price, household \#}; \\
 & (\text{all},c,\text{COM}) \text{p4}(c) \# \text{Purchaser's price, exports,loc\$ \#}; \\
 & (\text{all},c,\text{COM})(\text{all},s,\text{SRC}) \text{p5}(c,s) \# \text{Purchaser's price, government \#};
 \end{aligned}$$

4.6. The equation system

The rest of the TABLO Input file is an algebraic specification of the linear form of the model, with the equations organised into a number of blocks. Each Equation statement begins with a name and (optionally) a description. For PHILGEM, the equation name normally consists of the characters E_ followed by the name of the left-hand-side variable. Except where indicated, the variables are percentage changes. Variables are in lower-case characters and coefficients in upper case. Variables and coefficients are defined as the need arises. Readers who have followed the TABLO file so far should have no difficulty in reading the equations in the TABLO notation. We provide some commentary on the theory underlying each of the equation blocks.

4.7. Structure of production

PHILGEM allows each industry to produce several commodities, using as inputs domestic and imported commodities, labour of several types, land, and capital. In addition, commodities destined for export are distinguished from those for local use. The multi-input, multi-output production specification is kept manageable by a series of separability assumptions, illustrated by the nesting shown in Figure 7. For example, the assumption of *input-output separability* implies that the generalised production function for some industry:

$$F(\text{inputs},\text{outputs}) = 0 \tag{15}$$

may be written as:

$$G(\text{inputs}) = \text{X1TOT} = H(\text{outputs}) \tag{16}$$

where X1TOT is an index of industry activity. Assumptions of this type reduce the number of estimated parameters required by the model. Figure 7 shows that the H function in (16) is derived from two nested CET (constant elasticity of transformation) aggregation functions, while the G function is broken into a sequence of nests. At the top level, commodity composites, a primary-factor composite and 'other costs' are combined using a Leontief production function. Consequently, they are all demanded in direct proportion to X1TOT. Each commodity composite is a CES (constant elasticity of substitution) function of a domestic good and the imported equivalent. The primary-factor composite is a CES aggregate of land, capital and composite labour. Composite labour is a CES aggregate of occupational labour types. Although all industries share this common production structure, input proportions and behavioural parameters may vary between industries.

The nested structure is mirrored in the TABLO equations—each nest requiring 2 sets of equations. We begin at the bottom of Figure 7 and work upwards.

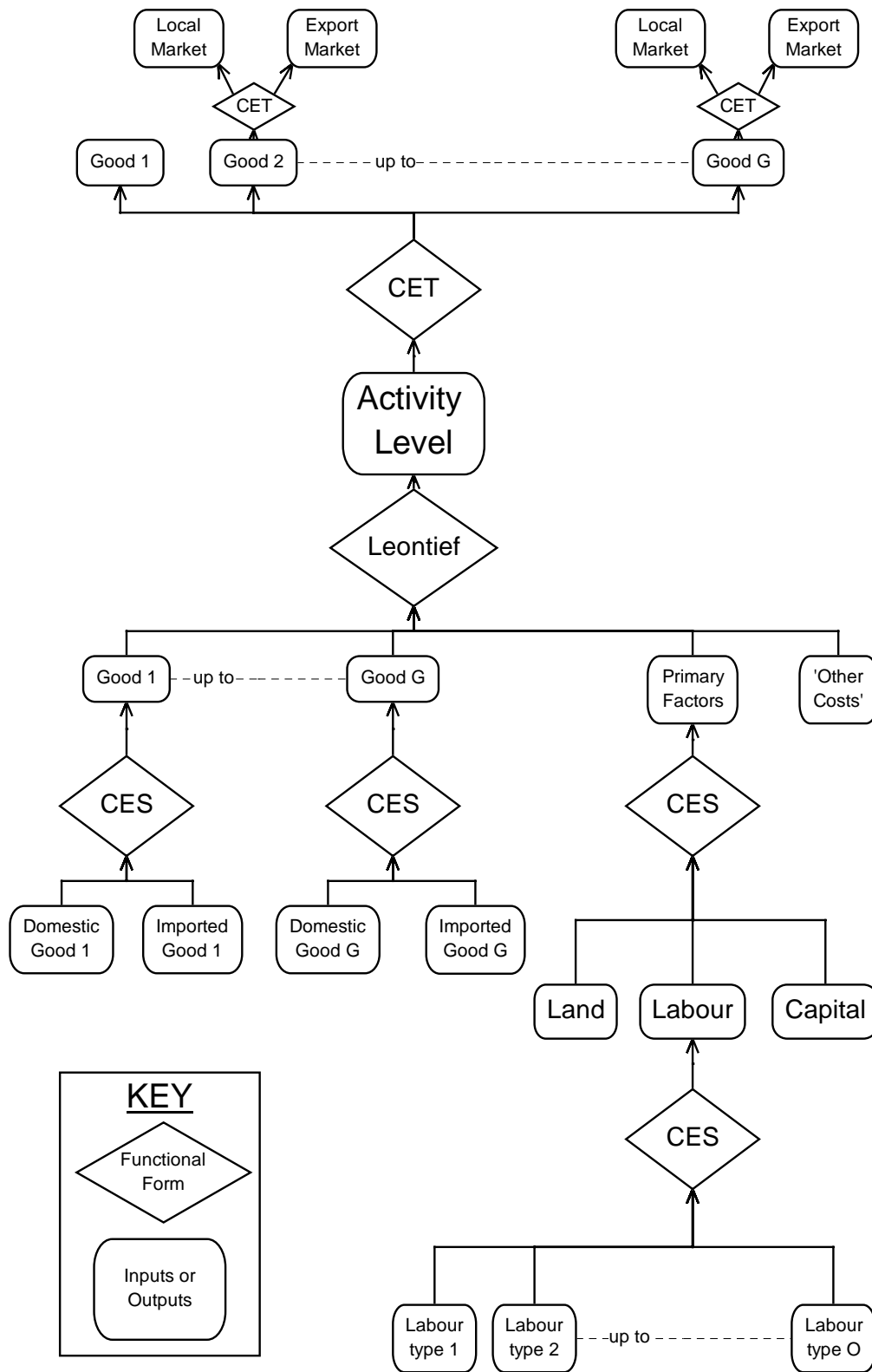


Figure 7. Structure of Production

4.8. Demands for primary factors

Excerpt 7 shows the equations determining the occupational composition of labour demand in each industry. For each industry i , the equations are derived from the following optimisation problem:

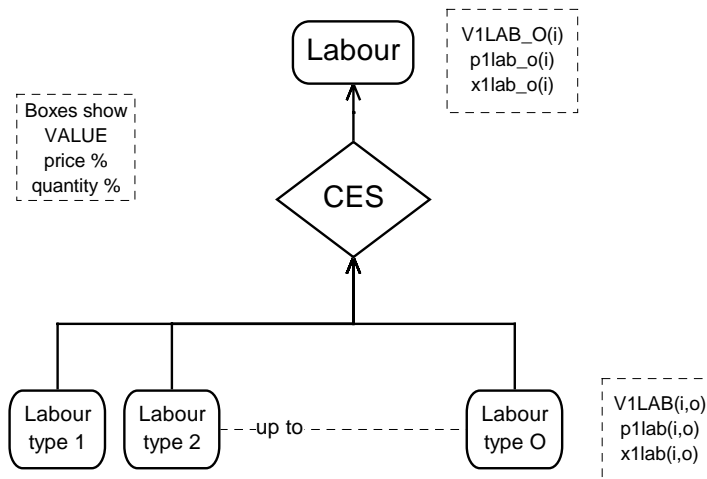


Figure 8. Demand for different types of labour

Choose inputs of occupation-specific labour,

$$X1LAB(i,o),$$

to minimize total labour cost,

$$\text{Sum}\{o, OCC, P1LAB(i,o)*X1LAB(i,o)\},$$

where

$$X1LAB_O(i) = \text{CES}[\text{All}, o, OCC: X1LAB(i,o)],$$

regarding as exogenous to the problem

$$P1LAB(i,o) \text{ and } X1LAB_O(i).$$

Note that the problem is formulated in the levels of the variables. Hence, we have written the variable names in upper case. The notation CES[] represents a CES function defined over the set of variables enclosed in the square brackets.

```
! Excerpt 7 of TABLO input file: !
! Occupational composition of labour demand !
Coefficient
(parameter)(all,i,IND) SIGMA1LAB(i) # CES substitution between skill types #;
(all,i,IND) V1LAB_O(i) # Total labour bill in industry i #;
          TINY # Small number to prevent zerodivides or singular matrix #;
Read SIGMA1LAB from file BASEDATA header "SLAB";
Formula
(all,i,IND) V1LAB_O(i) = sum{o,OCC, V1LAB(i,o)};
          TINY = 0.000000000001; !NB TINY+NUM=NUM, if NUM significant!
Variable
(all,i,IND) p1lab_o(i) # Price to each industry of labour composite #;
(all,i,IND) x1lab_o(i) # Effective labour input #;
Equation
E_x1lab # Demand for labour by industry and skill group #
(all,i,IND)(all,o,OCC)
x1lab(i,o) = x1lab_o(i) - SIGMA1LAB(i)*[p1lab(i,o) - p1lab_o(i)];
E_p1lab_o # Price to each industry of labour composite #
(all,i,IND) [TINY+V1LAB_O(i)]*p1lab_o(i) = sum{o,OCC, V1LAB(i,o)*p1lab(i,o)};
```

The solution of this problem, in percentage-change form, is given by equations E_x1lab and E_p1lab_o (see Appendix A for derivation). The first of the equations indicates that demand for labour type o is proportional to overall labour demand, X1LAB_O, and to a price term. In change form, the price term is composed of an elasticity of substitution, SIGMA1LAB(i), multiplied by the percentage change in a price ratio [p1lab(i,o)-p1lab_o(i)] representing the wage of occupation o relative to the average wage for labour in industry i. Changes in the relative prices of the occupations induce substitution in favour of relatively cheapening occupations. The percentage change in the average wage, p1lab_o(i), is given by the second of the equations. This could be rewritten:

$p1lab_o(i) = \text{sum}\{o, OCC, S1LAB(i,o)*p1lab(i,o)\}$,
 if $S1LAB(i,o)$ were the value share of occupation o in the total wage bill of industry i . In other words, $p1lab_o(i)$ is a Divisia index of the $p1lab(i,o)$.

It is worth noting that if the individual equations of E_x1lab were multiplied by corresponding elements of $S1LAB(i,o)$, and then summed together, all price terms would disappear, giving:

$$x1lab_o(i) = \text{sum}\{o, OCC, S1LAB(i,o)*x1lab(i,o)\}.$$

This is the percentage-change form of the CES aggregation function for labour.

For an industry which does not use labour (housing services is a common example), $V1LAB(i,o)$ would contain only zeros so that $p1lab_o(i)$ would be undefined. To prevent this, we add the coefficient TINY (set to some very small number) to the left hand side of equation E_p1lab_o . With $V1LAB_O(i)$ zero, equation E_p1lab_o becomes:

$$p1lab_o(i) = 0.$$

The same procedure is used extensively in later equations.

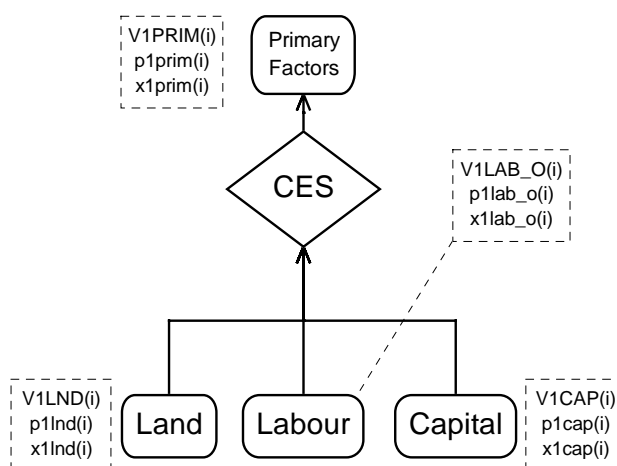


Figure 9. Primary Factor Demand

Excerpt 8 contains equations determining the composition of demand for primary factors. Their derivation follows a pattern similar to that underlying the previous nest. In this case, total primary factor costs are minimised subject to the production function:

$$X1PRIM(i) = CES \left[\frac{X1LAB_O(i)}{A1LAB_O(i)}, \frac{X1CAP(i)}{A1CAP(i)}, \frac{X1LND(i)}{A1LND(i)} \right].$$

Because we may wish to introduce factor-saving technical changes, we include explicitly the coefficients $A1LAB_O(i)$, $A1CAP(i)$, and $A1LND(i)$.

The solution to this problem, in percentage-change form, is given by equations E_x1lab_o , E_x1cap and E_x1lnd , and E_p1prim . Ignoring the technical-change terms, we see that demand for each factor is proportional to overall factor demand, $X1PRIM$, and to a price term. In change form the price term is an elasticity of substitution, $SIGMA1PRIM(i)$, multiplied by the percentage change in a price ratio representing the cost of an effective unit of the factor relative to the overall, effective cost of primary factor inputs to industry i . Changes in the relative prices of the primary factors induce substitution in favour of relatively cheapening factors. The percentage change in the average effective cost, $p1prim(i)$, given by equation E_p1prim , is again a cost-weighted Divisia index of individual prices and technical changes.

```
! Excerpt 8 of TABLO input file: !
! Primary factor proportions !
Coefficient
(parameter)(all,i,IND) SIGMA1PRIM(i) # CES substitution, primary factors #;
Read SIGMA1PRIM from file BASEDATA header "P028";
Coefficient (all,i,IND) V1PRIM(i) # Total factor input to industry i#;
Formula      (all,i,IND) V1PRIM(i) = TINY + V1LAB_O(i)+ V1CAP(i) + V1LND(i);
Variable
(all,i,IND) p1prim(i) # Effective price of primary factor composite #;
```

```
(all,i,IND) x1prim(i) # Primary factor composite #;
(all,i,IND) a1lab_o(i) # Labor-augmenting technical change #;
(all,i,IND) a1cap(i) # Capital-augmenting technical change #;
(all,i,IND) a1lnd(i) # Land-augmenting technical change #;
(change)(all,i,IND) delV1PRIM(i)# Ordinary change in cost of primary factors #;
```

Equation

```
E_x1lab_o # Industry demands for effective labour #
(all,i,IND) x1lab_o(i) - a1lab_o(i) =
x1prim(i) - SIGMA1PRIM(i)*[p1lab_o(i) + a1lab_o(i) - p1prim(i)];

E_p1cap # Industry demands for capital #
(all,i,IND) x1cap(i) - a1cap(i) =
x1prim(i) - SIGMA1PRIM(i)*[p1cap(i) + a1cap(i) - p1prim(i)];

E_p1lnd # Industry demands for land #
(all,i,IND) x1lnd(i) - a1lnd(i) =
x1prim(i) - SIGMA1PRIM(i)*[p1lnd(i) + a1lnd(i) - p1prim(i)];

E_p1prim # Effective price term for factor demand equations #
(all,i,IND) V1PRIM(i)*p1prim(i) = V1LAB_O(i)*[p1lab_o(i) + a1lab_o(i)]
+ V1CAP(i)*[p1cap(i) + a1cap(i)] + V1LND(i)*[p1lnd(i) + a1lnd(i)];

E_delV1PRIM # Ordinary change in total cost of primary factors #
(all,i,IND) 100*delV1PRIM(i) = V1CAP(i) * [p1cap(i) + x1cap(i)]
+ V1LND(i) * [p1lnd(i) + x1lnd(i)]
+ sum{o,OCC, V1LAB(i,o)* [p1lab(i,o) + x1lab(i,o)]};
```

Appendix A contains a formal derivation of CES demand equations with technical-change terms. The technical-change terms appear in a predictable pattern. Imagine that the percentage-change equations lacked these terms, as in the previous, occupational-demand, block. We could add them in by:

- replacing each quantity (x) variable by (x-a);
- replacing each price (p) variable by (p+a); and
- rearranging terms.

The last equation defines DelV1PRIM, the ordinary change in total cost of primary factors to each industry. This variable is used later in computing industries' total production costs.

4.9. Sourcing of intermediate inputs

We adopt the Armington (1969; 1970) assumption that imports are imperfect substitutes for domestic supplies. Excerpt 9 shows equations determining the import/domestic composition of intermediate commodity demands. They follow a pattern similar to the previous nest. Here, the total cost of imported and domestic good *i* are minimised subject to the production function:

$$X1_S(c,i) = CES\left[All,s,src: \frac{X1(c,s,i)}{A1(c,s,i)}\right], \quad (17)$$

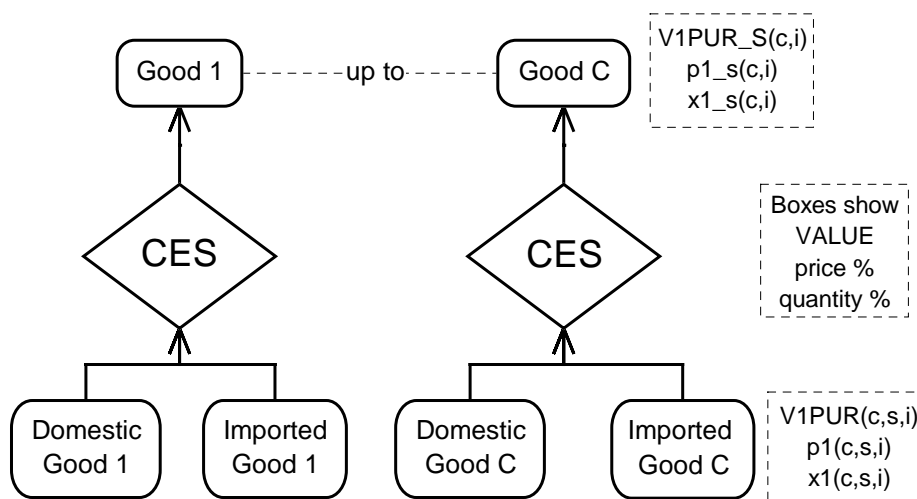


Figure 10. Intermediate input sourcing decision

Commodity demand from each source is proportional to demand for the composite, $X1_S(c,i)$, and to a price term. The change form of the price term is an elasticity of substitution, $SIGMA1(i)$, multiplied by the percentage change in a price ratio representing the effective price from the source relative to the effective cost of the import/domestic composite. Lowering of a source-specific price, relative to the average, induces substitution in favour of that source. The percentage change in the average effective cost, $p1_s(i)$, is again a cost-weighted Divisia index of individual prices and technical changes.

```

! Excerpt 9 of TABLO input file: !
! Import/domestic composition of intermediate demands !
Variable
(all,c,COM)(all,s,SRC)(all,i,IND) a1(c,s,i) # Intermediate basic tech change #;
(all,c,COM)(all,i,IND) x1_s(c,i) # Intermediate use of imp/dom composite #;
(all,c,COM)(all,i,IND) p1_s(c,i) # Price, intermediate imp/dom composite #;
(all,i,IND) p1mat(i) # Intermediate cost price index #;
(all,i,IND) p1var(i) # Short-run variable cost price index #;
Coefficient
(parameter)(all,c,COM) SIGMA1(c) # Armington elasticities: intermediate #;
(all,c,COM)(all,i,IND) V1PUR_S(c,i) # Dom+imp intermediate purch. value #;
(all,c,COM)(all,s,SRC)(all,i,IND) S1(c,s,i) # Intermediate source shares #;
(all,i,IND) V1MAT(i) # Total intermediate cost for industry i #;
(all,i,IND) V1VAR(i) # Short-run variable cost for industry i #;
Read SIGMA1 from file BASEDATA header "1ARM";
Zerodivide default 0.5;
Formula
(all,c,COM)(all,i,IND) V1PUR_S(c,i) = sum{s,SRC, V1PUR(c,s,i)};
(all,c,COM)(all,s,SRC)(all,i,IND) S1(c,s,i) = V1PUR(c,s,i) / V1PUR_S(c,i);
(all,i,IND) V1MAT(i) = sum{c,COM, V1PUR_S(c,i)};
(all,i,IND) V1VAR(i) = V1MAT(i) + V1LAB_0(i);
Zerodivide off;

Equation E_x1 # Source-specific commodity demands #
(all,c,COM)(all,s,SRC)(all,i,IND)
x1(c,s,i)-a1(c,s,i) = x1_s(c,i) -SIGMA1(c)*[p1(c,s,i) +a1(c,s,i) -p1_s(c,i)];

Equation E_p1_s # Effective price of commodity composite #
(all,c,COM)(all,i,IND)
p1_s(c,i) = sum{s,SRC, S1(c,s,i)*[p1(c,s,i) + a1(c,s,i)]};

Equation E_p1mat # Intermediate cost price index #
(all,i,IND)
p1mat(i) = sum{c,COM, sum{s,SRC, (V1PUR(c,s,i)/ID01[V1MAT(i)])*p1(c,s,i)}};

Equation E_p1var # Short-run variable cost price index #

```

```
(all,i,IND)
p1var(i) = [1/V1VAR(i)]*[V1MAT(i)*p1mat(i) + V1LAB_O(i)*p1lab_o(i)];
```

Following the pattern established for factor demands, we could have written Equation E_p1_s as:

$$V1PUR_S(c,i)*p1_s(c,i)=\text{Sum}\{s,\text{SRC},V1PUR(c,s,i)*[p1(c,s,i)+a1(c,s,i)]\};$$

where $V1PUR_S(c,i)$ is the sum over domestic and imported of $V1PUR(c,s,i)$. However, this equation would have left $p1_s(c,i)$ undefined when $V1PUR_S(c,i)$ is zero—not all industries use all commodities.

In computing the share:

$$S1(c,s,i)=V1PUR(c,s,i)/V1PUR_S(c,i),$$

(see again Excerpt 9) we used the Zerodivide statement to instruct GEMPACK to set import and domestic shares (arbitrarily) to 0.5 in such cases. This device avoids a numerical error in computing, without any other substantive consequence.

The last 2 equations defines 2 variables used to explain results. P1mat is an industry-specific index of the price of intermediate inputs¹⁰. P1var is an index of short-run variable cost: it includes the cost of all industry inputs except capital and land (which are fixed in the short-run). Changes in p1var correspond to vertical shifts in shortrun industry supply schedules. Equations for both p1mat and p1var are written with no coefficient on the LHS—this facilitates the use of GEMPACK’s AnalyseGE tool. AnalyseGE is able to ‘decompose’ the RHS of these equations to show the contribution of each individual price change to the total change in p1mat or p1var.

4.10. Top production nest

Excerpt 10 covers the topmost input-demand nest of Figure 7. Commodity composites, the primary-factor composite and 'other costs' are combined using a Leontief production function, given by:

$$X1TOT(i) = \frac{1}{A1TOT(i)} \times \text{MIN}\left[All,c,COM: \frac{X1_S(c,i)}{A1_S(c,i)}, \frac{X1PRIM(i)}{A1PRIM(i)}, \frac{X1OCT(i)}{A1OCT(i)}\right]. \quad (18)$$

Consequently, each of these three categories of inputs identified at the top level is demanded in direct proportion to $X1TOT(i)$.

The Leontief production function is equivalent to a CES production function with the substitution elasticity set to zero. Hence, the demand equations resemble those derived from the CES case but lack price (substitution) terms. The $a1tot(i)$ are Hicks-neutral technical-change terms, affecting all inputs equally.

```
! Excerpt 10 of TABLO input file: !
! Top nest of industry input demands !
Variable
(all,i,IND) x1tot(i)          # Activity level or value-added #;
(all,i,IND) a1prim(i)        # All factor augmenting technical change #;
(all,i,IND) a1tot(i)         # All input augmenting technical change #;
(all,i,IND) p1tot(i)         # Average input/output price #;
(all,i,IND) a1oct(i)         # "Other cost" ticket augmenting technical change#;
(all,c,COM)(all,i,IND) a1_s(c,i) # Tech change, int'mdiate imp/dom composite #;

Equation E_x1_s # Demands for commodity composites #
(all,c,COM)(all,i,IND) x1_s(c,i) - [a1_s(c,i) + a1tot(i)] = x1tot(i);

Equation E_x1prim # Demands for primary factor composite #
(all,i,IND) x1prim(i) - [a1prim(i) + a1tot(i)] = x1tot(i);

Equation E_x1oct # Demands for other cost tickets #
(all,i,IND) x1oct(i) - [a1oct(i) + a1tot(i)] = x1tot(i);
```

¹⁰ Equation E_p1mat uses the ID01 function built-in to GEMPACK: another way of avoiding divide-by-zero problems. If $x=0$, $ID01(x)=1$; otherwise $ID01(x)=x$.

4.11. Industry costs and production taxes

Excerpt 11 computes levels and changes in the total cost of production both excluding (V1CST) and including (V1TOT) an ad valorem production tax, V1PTX. The equations are arranged to facilitate changing the base of the production tax. For example, a comment line underneath equation E_delV1PTX shows how a value-added tax might be implemented: by making V1PRIM (rather than V1CST) the tax base.

```
! Excerpt 11 of TABLO input file: !
! Output cost inclusive of production tax !
Coefficient
(all,i,IND) V1CST(i)      # Total cost of industry i #;
(all,i,IND) V1TOT(i)     # Total industry cost plus tax #;
(all,i,IND) PTXRATE(i)   # Rate of production tax #;
Formula
(all,i,IND) V1CST(i)     = V1PRIM(i) + V1OCT(i) + V1MAT(i);
(all,i,IND) V1TOT(i)     = V1CST(i) + V1PTX(i);
(all,i,IND) PTXRATE(i)   = V1PTX(i)/V1CST(i); !VAT: V1PTX/V1PRIM !
Write PTXRATE to file SUMMARY header "PTXR";
Variable
(change)(all,i,IND) delV1CST(i) # Change in ex-tax cost of production #;
(change)(all,i,IND) delV1TOT(i) # Change in tax-inc cost of production #;
(change)(all,i,IND) delPTXRATE(i) # Change in rate of production tax #;

Equation
E_delV1CST (all,i,IND) delV1CST(i) = delV1PRIM(i) +
  sum{c,COM,sum{s,SRC, 0.01*V1PUR(c,s,i)*[p1(c,s,i) + x1(c,s,i)]}}
  + 0.01*V1OCT(i) * [p1oct(i) + x1oct(i)];

E_delV1PTX (all,i,IND) delV1PTX(i) =
  PTXRATE(i)*delV1CST(i) + V1CST(i) * delPTXRATE(i);
! VAT alternative: PTXRATE(i)*delV1PRIM(i) + V1PRIM(i)* delPTXRATE(i); !

E_delV1TOT (all,i,IND) delV1TOT(i) = delV1CST(i) + delV1PTX(i);

E_p1tot (all,i,IND) V1TOT(i)*[p1tot(i) + x1tot(i)] = 100*delV1TOT(i);

Variable (all,i,IND) p1cst(i) # Index of production costs (for AnalyseGE) #;
Equation E_p1cst (all,i,IND) p1cst(i) = [1/V1CST(i)]*[
  sum{c,COM,sum{s,SRC, V1PUR(c,s,i)*p1(c,s,i)}}
  + V1OCT(i) *p1oct(i)
  + V1CAP(i) *p1cap(i)
  + V1LND(i) *p1lnd(i)
  + sum{o,OCC, V1LAB(i,o) *p1lab(i,o)}];
```

The penultimate equation defines $p1tot(i)$ as the percentage change in the unit cost of production for industry i . Given the constant returns to scale which characterise the model's production technology, $p1tot$ is also the percentage change in marginal cost. We enforce the competitive *Zero Pure Profits* condition (price = marginal cost) by assuming that the $p1tot$ are also equal to the average price received by each industry.

Under constant returns to scale, we could combine equations listed previously to deduce:

$$p1tot(i) = \sum S_k p_k + \text{tax and technical change terms}$$

where S_k and p_k are respectively the share of input k in total cost, and the percent change in its price.

The final equation defines another variable used to explain results. $P1cst$ is an index of production costs which can be decomposed using AnalyseGE to show the contribution of individual price changes to the total change in input costs. This index ignores the effects of technical change.

4.12. From industry outputs to commodity outputs

PHILGEM allows for each industry to produce a mixture of all the commodities. For each industry, the mix varies, according to the relative prices of commodities¹¹. The first two equations of Excerpt 12 determine the commodity composition of industry output—the final nest of Figure 7. Here, the total revenue from all outputs is *maximised* subject to the production function:

$$X1TOT(i) = CET[All,c,COM: Q1(c,i)]. \tag{19}$$

The CET (constant elasticity of transformation) aggregation function is identical to CES, except that the transformation parameter in the CET function has the opposite sign to the substitution parameter in the CES function. In equation E_q1, an increase in a commodity price, relative to the average, induces transformation in favour of that output. The symbol, p1tot, defined in E_x1tot as average unit revenue, is the same as that used in the previous equation group to refer to the effective price of a unit of activity. This confirms our interpretation of equation E_p1tot as a *Zero Pure Profits* condition.

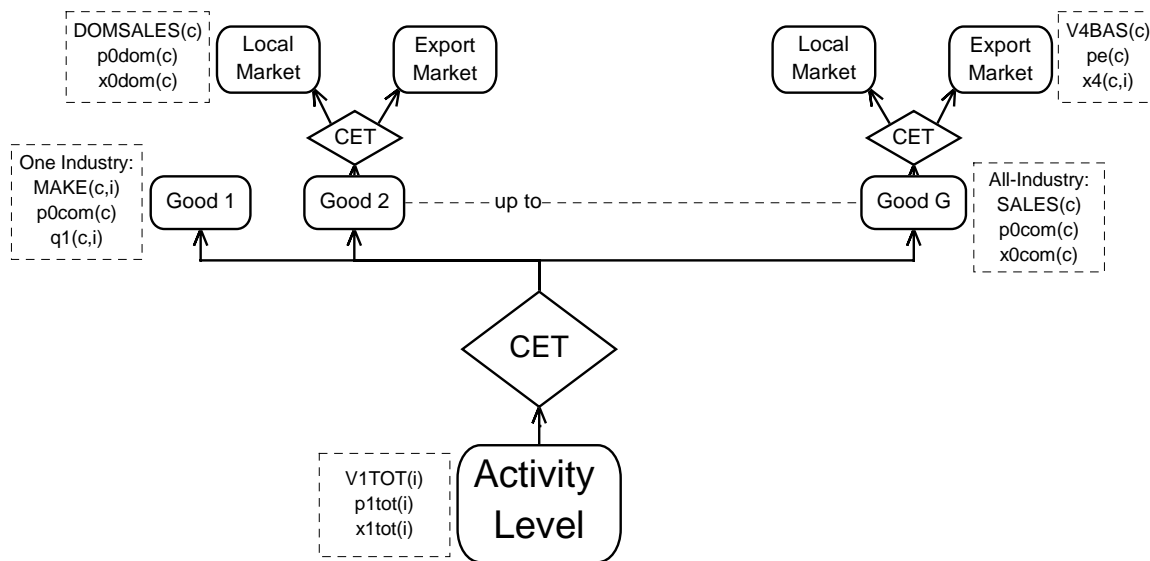


Figure 11. Composition of output

Equation E_pq1 states that all industries that produce, say, Corn, receive the same unit price, p0com("Corn"). Corn produced by different industries are deemed to be perfect substitutes¹². Equation E_x0com simply adds up all industries' output of each commodity to get the total supply, x0com.

It is also possible to enforce a one-to-one correspondence between industries and commodities. This is implied whenever all off-diagonal elements of the MAKE matrix are zero. In this case, the equations of Excerpt 12 just equate corresponding elements of p0com and p1tot. Similarly, x1tot and x0com become, in effect, the same variable. The computational overhead of including the MAKE mechanism is very slight.

```
! Excerpt 12 of TABLO input file: !
! Output mix of commodities !
Coefficient (all,c,COM)(all,i,IND) MAKE(c,i) # Multiproduction matrix #;
Variable
  (all,c,COM)(all,i,IND) q1(c,i) # Output by commodity and industry #;
  (all,c,COM)(all,i,IND) pq1(c,i) # Price of com c produced by ind i #;
```

¹¹ Multiproduction can be useful even where each industry produces just one commodity. For example we could split electricity generation into 2 parts: oil-fired and nuclear, each producing the same commodity, electricity.

¹² The perfect substitute assumption causes problems when we have two industries just producing the same commodity, and neither industry has a fixed factor; eg, in a long-run closure with mobile capital, and two electricity industries (say, nuclear and coal-fired) both making electric power. The model will find it hard to decide what proportion of generation should be nuclear. To cover such a case, the TABLO input file contains optional equations (commented out in Excerpt 12), which allow (for example) nuclear electricity to be an imperfect substitute for coal-fired electricity.

```

(all,c,COM) p0com(c) # General output price of locally-produced commodity #;
Read MAKE from file BASEDATA header "MAKE";
Update (all,c,COM)(all,i,IND) MAKE(c,i)= pq1(c,i)*q1(c,i);
Variable
(all,c,COM) x0com(c) # Output of commodities #;
Coefficient
(parameter)(all,i,IND) SIGMA1OUT(i) # CET transformation elasticities #;
Read SIGMA1OUT from file BASEDATA header "SCET";
Equation E_q1 # Supplies of commodities by industries #
(all,c,COM)(all,i,IND)
q1(c,i) = x1tot(i) + SIGMA1OUT(i)*[pq1(c,i) - p1tot(i)];
Coefficient
(all,i,IND) MAKE_C(i) # All production by industry i #;
(all,c,COM) MAKE_I(c) # Total production of commodities #;
Formula
(all,i,IND) MAKE_C(i) = sum{c,COM, MAKE(c,i)};
(all,c,COM) MAKE_I(c) = sum{i,IND, MAKE(c,i)};

Equation E_x1tot # Average price received by industries #
(all,i,IND) p1tot(i) = sum{c,COM, [MAKE(c,i)/MAKE_C(i)]*pq1(c,i)};

!*****!
! ORANIG assumes that, for example, Wheat produced by Industry 1
! is a perfect substitute for Wheat produced by Industry 2 !

Equation
E_pq1 # Each industry gets the same price for a given commodity #
(all,c,COM)(all,i,IND) pq1(c,i) = p0com(c);
E_x0com # Total output of commodities (as simple addition) #
(all,c,COM) x0com(c) = sum{i,IND, [MAKE(c,i)/MAKE_I(c)]*q1(c,i)};

! The perfect substitute assumption causes problems when we have two industries
! just producing the same commodity, and neither industry has a fixed factor.
! EG, in a long-run closure with mobile capital, and two electricity industries
! (say, nuclear and coal-fired) both making electric power. The model will
! find it hard to decide what proportion of generation should be nuclear.
! In such a case, the previous two equations could be commented out and
! replaced with the next two equations, which allow (for example) nuclear
! electricity to be an imperfect substitute for coal-fired electricity. !
!*****!
![[[
Coefficient (parameter)(all,c,COM) CESMAKE(c) # Inter-MAKE Elasticities #;
Read CESMAKE from file BASEDATA header "CMAK"; ! could have values like 5.0 !
Equation E_pq1 # Demands for commodities from industries #
(all,c,COM)(all,i,IND) q1(c,i) = x0com(c) - CESMAKE(c)*[pq1(c,i) - p0com(c)];
Equation E_x0com # Total output of commodities (as CES quantity index) #
(all,c,COM) x0com(c) = sum{i,IND, [MAKE(c,i)/MAKE_I(c)]*q1(c,i)};!
! for low values [<1] of CESMAKE use next equation instead of above !
!(all,c,COM) MAKE_I(c)*p0com(c) = sum{i,IND, MAKE(c,i)*pq1(c,i)};! !]]!
!*****!

```

4.13. Export and local market versions of each good

Excerpt 13 allows for the possibility that goods destined for export are not the the same as those for local use¹³. Conversion of an undifferentiated commodity into goods for both destinations is governed by a CET transformation frontier. Conceptually, the system is the same as Excerpt 12, but it is expressed a little differently; partly because there are only two outputs; and partly to facilitate switching the system off. This is achieved by setting TAU to zero, so that p0com, p0dom and pe are all equal.

¹³ This feature is not part of the ORANI tradition, but appears in some other applied GE models.

```

! Excerpt 13 of TABLO input file: !
! CET between outputs for local and export markets !
Variable
  (all,c,COM) x0dom(c) # Output of commodities for local market #;
Coefficient
  (all, c,COM) EXPSHR(c) # Share going to exports #;
  (all, c,COM) TAU(c) # 1/Elast. of transformation, exportable/locally used #;
Zerodivide default 0.5;
Formula
  (all,c,COM) EXPSHR(c) = V4BAS(c)/MAKE_I(c);
  (all,c,COM) TAU(c) = 0.0; ! if zero, p0dom = pe, and CET is nullified !
Zerodivide off;
Equation E_x0dom # Supply of commodities to export market #
  (all,c,COM) TAU(c)*[x0dom(c) - x4(c)] = p0dom(c) - pe(c);
Equation E_pe # Supply of commodities to domestic market #
  (all,c,COM) x0com(c) = [1.0-EXPSHR(c)]*x0dom(c) + EXPSHR(c)*x4(c);
Equation E_p0com # Zero pure profits in transformation #
  (all,c,COM) p0com(c) = [1.0-EXPSHR(c)]*p0dom(c) + EXPSHR(c)*pe(c);

```

The names of the prices, quantities and flows in the two CET nests of Excerpt 12 are shown below:

Joint Production CET Nest					
Type of Variable	Industry Output	Commodity Outputs	Undifferentiated Commodity	Local Destination	Export Destination
%Δ quantity	x1tot(i)	q1(c,i)	x0com(c)	x0dom(c)	x4(c)
%Δ price	p1tot(i)	p0com(c)	p0com(c)	p0dom(c) = p0(c,"dom")	pe(c)
Value of flow	V1TOT(i)	MAKE(c,i)	SALES(c)	DOMSALES(c)	V4BAS(c)
Export/Domestic CET nest					

Note that $V1TOT(i) = \sum_c MAKE(c,i)$ and $SALES(c) = \sum_i MAKE(c,i) = DOMSALES(c) + V4BAS(c)$

4.14. Demands for investment goods

Figure 12 shows the nesting structure for the production of new units of fixed capital. Capital is assumed to be produced with inputs of domestically produced and imported commodities. The production function has the same nested structure as that which governs intermediate inputs to current production. No primary factors are used directly as inputs to capital formation.

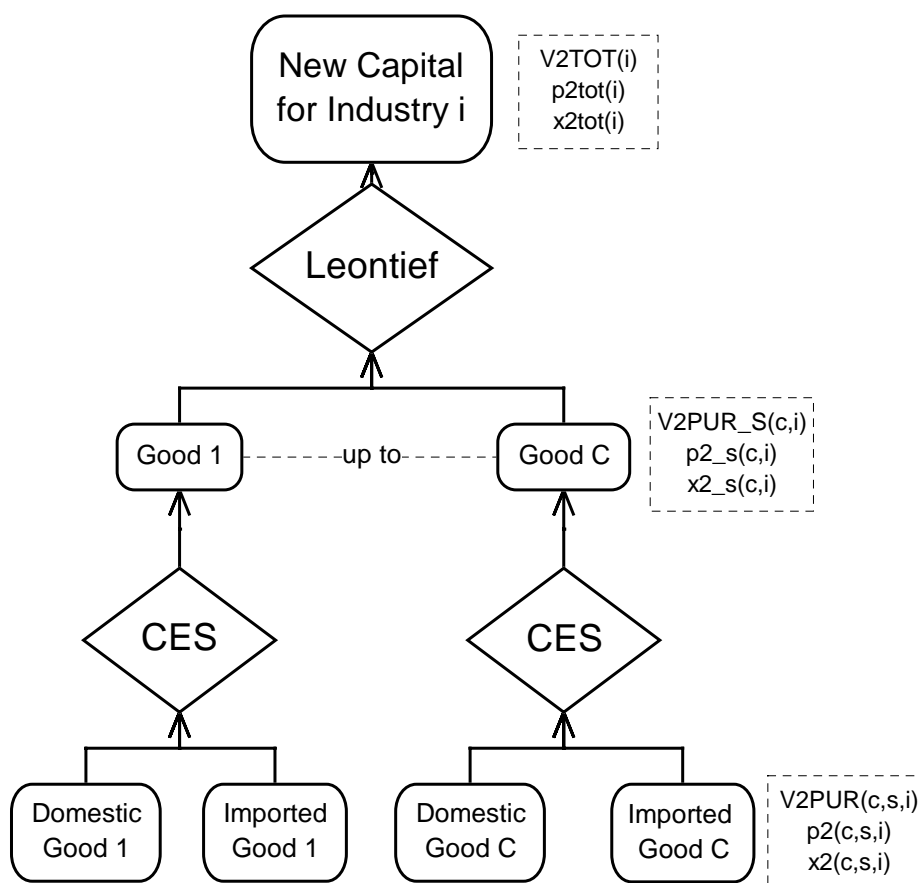


Figure 12. Structure of Investment Demand

The investment demand equations (see Excerpt 14) are derived from the solutions to the investor's two-part cost-minimisation problem. At the bottom level, the total cost of imported and domestic good *i* is minimised subject to the CES production function:

$$X2_S(c, i) = CES[All,s, SRC: \frac{X2(c,s,i)}{A2(c,s,i)}], \tag{20}$$

while at the top level the total cost of commodity composites is minimised subject to the Leontief production function:

$$X2TOT(i) = \frac{1}{A2TOT(i)} \text{MIN}[All,c, COM: \frac{X2_S(c,i)}{A2_S(c,i)}]. \tag{21}$$

The total amount of investment in each industry, $X2TOT(i)$, is exogenous to the above cost-minimisation problem. It is determined by other equations, covered later on.

The equations in Excerpt 14 describing the demand for source-specific inputs (E_{x2} and E_{p2_s}) and for composites (E_{x2_s}) are very similar to the corresponding intermediate demand equations in previous excerpts. The source-specific demand equation (E_{x2}) requires an elasticity of substitution, $SIGMA2(i)$. Also included is an equation which determines the price of new units of capital as the average cost of producing a unit—a *Zero Pure Profits* condition.

```
! Excerpt 14 of TABLO input file: !
! Investment demands !
Variable
(all,c,COM)(all,i,IND) x2_s(c,i) # Investment use of imp/dom composite #;
(all,c,COM)(all,i,IND) p2_s(c,i) # Price, investment imp/dom composite #;
(all,c,COM)(all,s, SRC)(all,i,IND) a2(c,s,i) # Investment basic tech change #;
Coefficient
(parameter) (all,c,COM) SIGMA2(c) # Armington elasticities: investment #;
Read SIGMA2 from file BASEDATA header "2ARM";
Coefficient ! Source Shares in Flows at Purchaser's prices !
(all,c,COM)(all,i,IND) V2PUR_S(c,i) # Dom+imp investment purch. value #;
```

```

(all,c,COM)(all,s,SRC)(all,i,IND) S2(c,s,i) # Investment source shares #;
Zerodivide default 0.5;
Formula
(all,c,COM)(all,i,IND) V2PUR_S(c,i) = sum{s,SRC, V2PUR(c,s,i)};
(all,c,COM)(all,s,SRC)(all,i,IND) S2(c,s,i) = V2PUR(c,s,i) / V2PUR_S(c,i);
Zerodivide off;

Equation E_x2 # Source-specific commodity demands #
(all,c,COM)(all,s,SRC)(all,i,IND)
x2(c,s,i)-a2(c,s,i) - x2_s(c,i) = - SIGMA2(c)*[p2(c,s,i)+a2(c,s,i) - p2_s(c,i)];

Equation E_p2_s # Effective price of commodity composite #
(all,c,COM)(all,i,IND)
p2_s(c,i) = sum{s,SRC, S2(c,s,i)*[p2(c,s,i)+a2(c,s,i)]};

! Investment top nest !

Variable
(all,i,IND) a2tot(i) # Neutral technical change - investment #;
(all,i,IND) p2tot(i) # Cost of unit of capital #;
(all,i,IND) x2tot(i) # Investment by using industry #;
(all,c,COM)(all,i,IND) a2_s(c,i) # Tech change, investment imp/dom composite #;
Coefficient (all,i,IND) V2TOT(i) # Total capital created for industry i #;
Formula (all,i,IND) V2TOT(i) = sum{c,COM, V2PUR_S(c,i)};
Equation
E_x2_s (all,c,COM)(all,i,IND) x2_s(c,i) - [a2_s(c,i) + a2tot(i)] = x2tot(i);
E_p2tot (all,i,IND) p2tot(i)
= sum{c,COM, (V2PUR_S(c,i)/ID01[V2TOT(i)])*[p2_s(c,i) +a2_s(c,i) +a2tot(i)]};

```

4.15. Household demands

As Figure 13 shows, the nesting structure for household demand is nearly identical to that for investment demand. The only difference is that commodity composites are aggregated by a Klein-Rubin, rather than a Leontief, function, leading to the linear expenditure system (LES).

The equations for the lower, import/domestic nest (see Excerpt 15) are similar to the corresponding equations for intermediate and investment demands.

```

! Excerpt 15 of TABLO input file: !
! Import/domestic composition of household demands !
Variable
(all,c,COM)(all,s,SRC) a3(c,s) # Household basic taste change #;
(all,c,COM) x3_s(c) # Household use of imp/dom composite #;
(all,c,COM) p3_s(c) # Price, household imp/dom composite #;
Coefficient
(parameter)(all,c,COM) SIGMA3(c) # Armington elasticities: households #;
Read SIGMA3 from file BASEDATA header "3ARM";

Coefficient ! Source Shares in Flows at Purchaser's prices !
(all,c,COM) V3PUR_S(c) # Dom+imp households purch. value #;
(all,c,COM)(all,s,SRC) S3(c,s) # Household source shares #;
Zerodivide default 0.5;
Formula
(all,c,COM) V3PUR_S(c) = sum{s,SRC, V3PUR(c,s)};
(all,c,COM)(all,s,SRC) S3(c,s) = V3PUR(c,s) / V3PUR_S(c);
Zerodivide off;

Equation E_x3 # Source-specific commodity demands #
(all,c,COM)(all,s,SRC)
x3(c,s)-a3(c,s) = x3_s(c) - SIGMA3(c)*[ p3(c,s)+a3(c,s) - p3_s(c) ];

Equation E_p3_s # Effective price of commodity composite #
(all,c,COM) p3_s(c) = sum{s,SRC, S3(c,s)*[p3(c,s)+a3(c,s)]};

```

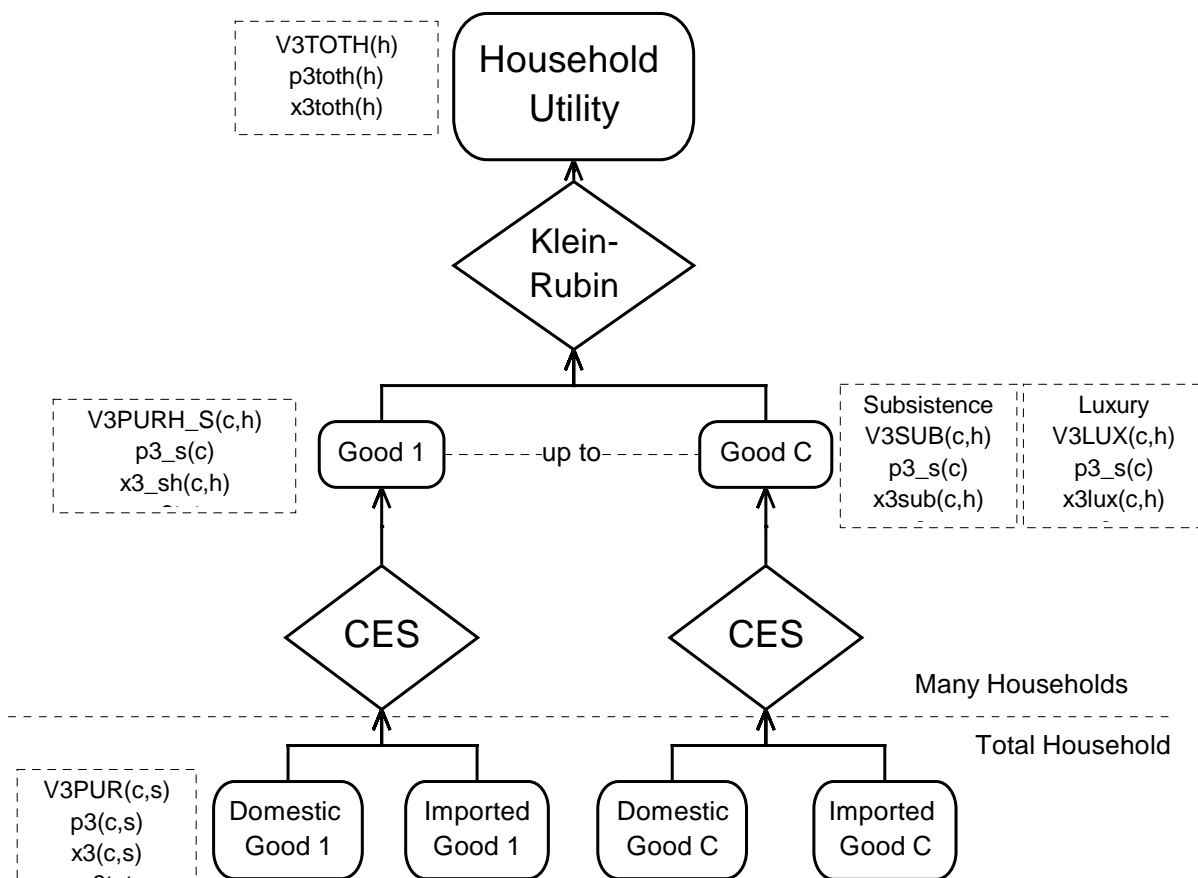



Figure 13. Structure of Consumer Demand

Excerpt 16 shows an important feature of PHILGEM: its ability to handle multiple household categories (originally, ORANI-G had just one). As far as possible, the necessary changes are localized to Excerpt 16. The changes include: new variables $p3toth$, $x3toth$, $w3luxh$, $w3toth$, qh , utility and $x3_sh$ with household dimension; and a satellite data matrix size $COM*HOU$, which must add up (over HOU) to the "core" data matrix $V3PUR (COM*SRC)$ —which is ensured by invoking the Assertion command in Excerpt 16. An implication of this approach is that we assume household imp/dom shares are independent of household type.

Excerpt 16 of the TABLO input file determines the allocation of household expenditure between commodity composites. This is derived from the Klein-Rubin utility function for each household category:¹⁴

$$\text{Utility per household} = \frac{1}{Q} \prod_c \{X3_S(c) - X3SUB(c)\}^{S3LUX(c)}, \tag{22}$$

The $X3SUB$ and $S3LUX$ are behavioural coefficients—the $S3LUX$ must sum to unity. QH is the number of households per household type. The demand equations that arise from this utility function are:

$$X3_SH(c) = X3SUB(c) + S3LUX(c)*V3LUX_C/P3_S(c), \tag{23}$$

where:

$$V3LUX_C = V3TOT - \sum X3SUB(c)*P3_S(c) \tag{24}$$

The name of the linear expenditure system derives from its property that expenditure on each good is a linear function of prices ($P3_S$) and expenditure ($V3TOT$). The form of the demand equations gives rise to the following interpretation. The $X3SUB$ are said to be the 'subsistence' requirements of each

¹⁴ For simplicity, the household index h is omitted.

good—these quantities are purchased regardless of price. V3LUX_C is what remains of the consumer budget after subsistence expenditures are deducted—we call this 'luxury' or 'supernumerary' expenditure. The S3LUX are the shares of this remnant allocated to each good—the marginal budget shares. Such an interpretation facilitates our transition to percentage change form, which begins from the levels equations:

$$X3_S(c) = X3SUB(c) + X3LUX(c) \quad (25)$$

$$X3LUX(c)*P3_S(c) = S3LUX(c)*V3LUX_C \quad (26)$$

$$X3SUB(c) = Q*A3SUB(c) \quad (27)$$

As equation (25) makes plain, the X3LUX are luxury usages, or the difference between the subsistence quantities and total demands. Equation (26) states that luxury expenditures follow the marginal budget shares S3LUX. Together, equations (25) and (26) are equivalent to (23). Equation (27) is necessary because our demand system applies to aggregate instead of to individual households. It states that total subsistence demand for each good c is proportional to the number of households, Q, and to the individual household subsistence demands, A3SUB(c). The percentage change forms of equations (27), (26) and (25) appear as the first three equation items in Excerpt 16 of the Tablo Input File. Note that a3lux(c) and w3lux are respectively the percentage changes in S3LUX(c) and V3LUX_C.

```
! Excerpt 16 of TABLO input file: !
! Household demands for composite commodities !
Variable
(all,h,HOU)          p3toth(h) # Consumer price index #;
(all,h,HOU)          x3toth(h) # Real household consumption #;
(all,h,HOU)          w3toth(h) # Nominal total household consumption #;
(all,h,HOU)          w3luxh(h) # Nominal luxury consumption #;
(all,h,HOU)          qh(h) # Number of households #;
(all,h,HOU)          utilityh(h) # Utility per household #;
(all,c,COM)(all,h,HOU) x3lux(c,h) # Household - supernumerary demands #;
(all,c,COM)(all,h,HOU) x3sub(c,h) # Household - subsistence demands #;
(all,c,COM)(all,h,HOU) a3lux(c,h) # Taste change, supernumerary demands #;
(all,c,COM)(all,h,HOU) a3sub(c,h) # Taste change, subsistence demands #;
(all,c,COM)(all,h,HOU) a3_s(c,h) # Taste change, hhold imp/dom composite #;
(all,c,COM)(all,h,HOU) x3_sh(c,h) # Consumption, hhold imp/dom composite #;

Coefficient
(all,h,HOU) V3TOTH(h) # Total purchases for each household #;
              V3TOT      # Total purchases by households #;
(all,h,HOU) FRISCH(h) # Frisch LES 'parameter'= - (total/luxury) #;
(all,c,COM)(all,h,HOU) EPS(c,h) # Household expenditure elasticities #;
(all,c,COM)(all,h,HOU) S3_S(c,h) # Household average budget shares #;
(all,c,COM)(all,h,HOU) B3LUX(c,h) # Ratio, (supernumerary /total expenditure)#;
(all,c,COM)(all,h,HOU) S3LUX(c,h) # Marginal household budget shares #;
(all,c,COM)(all,h,HOU) V3PURH_S(c,h) # Consumption, hhold imp/dom composite #;

Read  FRISCH from file BASEDATA header "P21h";
      EPS from file BASEDATA header "XPEL";
      V3PURH_S from file BASEDATA header "3PUR";

! First, check that V3PUR_SH(c,h) adds to V3PUR_S(c) !
Coefficient
(all,c,COM) V3PURH_SH(c) # Total Household expenditure by good #;
(all,c,COM) V3PURH_CHECK(c) # must be zero #;

Formula
(all,c,COM) V3PURH_SH(c) = sum{h,HOU, V3PURH_S(c,h)};
(all,c,COM) V3PURH_CHECK(c)= V3PURH_SH(c) - V3PUR_S(c);

Assertion # Check that V3PUR_SH(c,h) adds to V3PUR_S(c) #
```

(all,c,COM) ABS[V3PURH_CHECK(c)]<0.5;

Update

(change)(all,h,HOU) FRISCH(h) = FRISCH(h)*[w3toth(h) - w3luxh(h)]/100.0;
 (all,c,COM)(all,h,HOU) V3PURH_S(c,h) = p3_s(c)*x3_sh(c,h);
 (change)(all,c,COM)(all,h,HOU) EPS(c,h) = EPS(c,h)*
 [x3lux(c,h)-x3_sh(c,h)+w3toth(h) - w3luxh(h)]/100.0;

Formula

(all,h,HOU) V3TOT(h) = sum{c,COM, V3PURH_S(c,h)};
 V3TOT = sum{h,HOU, V3TOT(h)};
 (all,c,COM)(all,h,HOU) S3_S(c,h) = V3PURH_S(c,h)/V3TOT(h);

Coefficient (all,h,HOU) EPSTOTH(h) # Average Engel elasticity: should = 1 #;

Formula (all,h,HOU) EPSTOTH(h) = sum{c,COM, S3_S(c,h)*EPS(c,h)};

! Below is optional and slightly unorthodox: EPS is reinitialized to scaled value; otherwise EPS values drift in many-period simulation !

! ensure average EPS=1 !

(initial)(all,c,COM)(all,h,HOU) EPS(c,h) = EPS(c,h)/EPSTOTH(h);
 (all,h,HOU) EPSTOTH(h) = sum{c,COM, S3_S(c,h)*EPS(c,h)};
 (all,c,COM)(all,h,HOU) B3LUX(c,h) = EPS(c,h)/ABS[FRISCH(h)];

! above, initial sign of Frisch ignored !

(all,c,COM)(all,h,HOU) S3LUX(c,h) = EPS(c,h)*S3_S(c,h);

Write

S3LUX to file SUMMARY header "LSHR";
 S3_S to file SUMMARY header "CSHR";
 V3PURH_CHECK to file SUMMARY header "V3CH";

Equation

E_x3sub # Subsistence demand for composite commodities #
 (all,c,COM)(all,h,HOU) x3sub(c,h) = qh(h) + a3sub(c,h);

E_x3lux # Luxury demand for composite commodities #
 (all,c,COM)(all,h,HOU) x3lux(c,h) + p3_s(c) = w3luxh(h) + a3lux(c,h);

E_x3_sh # Total household demand for composite commodities #
 (all,c,COM)(all,h,HOU)
 x3_sh(c,h) = B3LUX(c,h)*x3lux(c,h) + [1-B3LUX(c,h)]*x3sub(c,h);

E_utilityh # Change in utility disregarding taste change terms #
 (all,h,HOU) utilityh(h) + qh(h) = sum{c,COM, S3LUX(c,h)*x3lux(c,h)};

E_a3lux # Default setting for luxury taste shifter #
 (all,c,COM)(all,h,HOU)
 a3lux(c,h) = a3sub(c,h) - sum{k,COM, S3LUX(k,h)*a3sub(k,h)};

E_a3sub # Default setting for subsistence taste shifter #
 (all,c,COM)(all,h,HOU) a3sub(c,h) = a3_s(c,h)-sum{k,COM,S3_S(k,h)*a3_s(k,h)};

E_x3toth # Real consumption #
 (all,h,HOU) x3toth(h) = sum{c,COM, S3_S(c,h)*x3_sh(c,h)};

E_p3toth # Consumer price index #
 (all,h,HOU) p3toth(h) = sum{c,COM, S3_S(c,h)*p3_s(c)};

E_w3toth # Household budget constraint: determines w3lux #
 (all,h,HOU) w3toth(h) = x3toth(h) + p3toth(h);

! Aggregate Household !

Variable

p3tot # Consumer price index #;
 x3tot # Real household consumption #;

w3tot # Nominal total household consumption #;

Equation

E_x3tot # Real consumption #

0 = sum{h,HOU, V3TOT(h)*[x3toth(h)-x3tot]};

E_p3tot # Consumer price index #

0 = sum{h,HOU, V3TOT(h)*[p3toth(h)-p3tot]};

E_w3tot # Household budget constraint: determines w3lux #

w3tot = x3tot + p3tot;

E_x3_s # Total household demand for composite commodities #

(all,c,COM) sum{h,HOU, ID01[V3PURH_S(c,h)]*[x3_sh(c,h)- x3_s(c)]} = 0;

Formula (all,h,HOU) EPSTOTH(h) = sum{c,COM, S3_S(c,h)*EPS(c,h)};

Assertion (initial) # Check ave EPS =1 # (all,h,HOU) ABS[1-EPSTOTH(h)]<0.01;

!Assertion # Hou check # (all,c,COM)

ABS[sum{s, SRC, V3BAS(c,s)+V3TAX(c,s)+sum{m, MAR, V3MAR(c,s,m)}}
- sum{h,HOU, V3PURH_S(c,h)}]<0.5;!

Equation E_utility is the percentage-change form of the utility function (22). Equations E_a3sub and E_a3lux provide default settings for the taste-change variables, a3sub and a3lux, which allow the average budget shares to be shocked, *via* the a3_s, in a way that preserves the pattern of expenditure elasticities. See Appendix F for further details.

The final four equations in Excerpt 16 relates to the aggregate household behaviour as the equations just described determine the composition of household demands but do not determine total consumption. That could be done in a variety of ways: (i) set exogenously, (ii) determined by a consumption function, or (iii) determined *via* a balance of trade constraint.

The reader may wonder why there is no equation akin to (24) which would determine the variable w3luxh. The reason is that (24) can be deduced from (23) and the definition of V3TOT:

$$V3TOT = \sum X3_S(c)*P3_S(c) \tag{28}$$

The many-household percentage change form of (28) in fact appears as the final equation E_w3toth. Hence any additional equation defining w3luxh would be redundant.

4.16. Export demands

To model export demands, commodities in PHILGEM (ORANI-G) are divided into two groups:

- For an *individual export* commodity, foreign demand is inversely related to that commodity's price.
- For the remaining, *collective export*, commodities, foreign demand is inversely related to the average price of *all* collective export commodities.

! Excerpt 17 of TABLO input file: !

! Export demands !

Coefficient

(parameter)(all,c,COM) IsIndivExp(c) # >0.5 For individual export commodities#;

Read IsIndivExp from file BASEDATA header "ITEX";

! This way of defining a set facilitates aggregation of the data base !

Set

TRADEXP # Individual export commodities # = (all,c,COM: IsIndivExp(c)>0.5);

NTRADEXP # Collective Export Commodities # = COM - TRADEXP;

Write

(Set)TRADEXP to file SUMMARY header "TEXP";

(Set) NTRADEXP to file SUMMARY header "NTEXP";

Variable

(all,c,COM) pf4(c) # Foreign currency export prices #;

(all,c,COM) f4p(c) # Price (upward) shift in export demand schedule #;

```

(all,c,COM) f4q(c) # Quantity (right) shift in export demands #;
           phi # Exchange rate, local currency/$world #;
           x4_ntrad # Quantity, collective export composite #;
           f4p_ntrad # Uniform upward (price) demand shift for collective expts #;
           f4q_ntrad # Uniform right (quantity) demand shift for collective exports #;
           p4_ntrad # Average price of collective exports #;

Coefficient
(parameter)(all,c,COM) EXP_ELAST(c)
                        # Export demand elasticities: typical value -5.0 #;
(parameter) EXP_ELAST_NT # Collective export demand elasticity #;
           V4NTRADEXP # Total collective export earnings #;

Read
EXP_ELAST from file BASEDATA header "P018";
EXP_ELAST_NT from file BASEDATA header "EXNT";

Formula V4NTRADEXP = sum{c,NTRADEXP, V4PUR(c)};

Equation
E_x4A # Individual export demand functions # (all,c,TRADEXP)
      x4(c) - f4q(c) = -ABS[EXP_ELAST(c)]*[pf4(c) - f4p(c)];
! note: ABS function above fixes common mistake: positive EXP_ELAST values !

! To simulate exogenous export prices, exogenize pf4 instead of f4p !
E_pf4 # Foreign currency export prices # (all,c,COM)
      pf4(c) = p4(c) - phi;

E_x4B # Collective export demand functions # (all,c,NTRADEXP)
      x4(c) - f4q(c) = x4_ntrad;
!NB: if f4q(c) shocked, x4_ntrad is not total of the collective x4 !

E_p4_ntrad # Average price of collective exports #
      [TINY+V4NTRADEXP]*p4_ntrad = sum{c,NTRADEXP, V4PUR(c)*p4(c)};

E_x4_ntrad # Demand for collective export composite #
      x4_ntrad - f4q_ntrad = -ABS[EXP_ELAST_NT]*[p4_ntrad - phi - f4p_ntrad];

```

The individual export group, which would normally include all the main export commodities, is defined by the set TRADEXP¹⁵ in Excerpt 17. Equation E_x4A specifies downward-sloping foreign demand schedules for these commodities. In the levels, the equation would read:

$$X4(c) = F4Q(c) \left[\frac{P4(c)}{PHI * F4P(c)} \right]^{EXP_ELAST(c)}, \quad (29)$$

where EXP_ELAST(c) is a negative¹⁶ parameter—the constant elasticity of demand. That is, export volumes, X4(c), are declining functions of their prices in foreign currency, (P4(c)/PHI). The exchange rate PHI converts local to foreign currency units. The variables F4Q(i) and F4P(i) allow for horizontal (quantity) and vertical (price) shifts in the demand schedules.

The collective export group, defined by the set NTRADEXP, could include all those commodities for which the above equation is inappropriate. In PHILGEM, these commodities are limited to other crops and other agriculture.¹⁷

The commodity composition of aggregate collective exports is exogenised by treating collective exports as a Leontief aggregate (see equation E_x4B). Demand for the aggregate is related to its average

¹⁵ The original version of ORANI referred to individual exports as *traditional* exports; the rest were called *non-traditional* exports. For Australia, the traditional exports were all primary products.

¹⁶ The ABS (absolute value) function is used to neutralize mistakes about the sign of EXP_ELAST.

¹⁷ For models based on ORANI-G this has typically included the bulk of the service commodities, and other commodities (examples might be Air Transport or Gold) where export volumes do not seem to depend mainly on the corresponding price. The list would be country-specific.

price *via* a constant-elasticity demand curve, similar to those for individual exports (see equation E_x4_ntrad).

It is permissible for the set TRADEXP to contain *all* commodities, so that the set NTRADEXP is empty. This disactivates the collective export mechanism. In addition, exports of particular commodities may be fixed or shocked by exogenising the corresponding elements of the vector x4, while at the same time endogenising matching elements of the vector f4q.

4.17. Other final demands

Equations E_x5 and E_f5tot determine government usage. With both of the shift variables f5 and f5tot exogenous, the level and composition of government consumption is exogenously determined. Then equation E_f5tot merely determines the value of the endogenous variable f5tot2, which appears nowhere else. Alternatively, it is also possible to assume that, in the absence of shocks to the shift variables, aggregate government consumption moves with real aggregate household consumption, x3tot. This is achieved by *endogenising* f5tot and *exogenising* f5tot2. The trick of changing behavioural specifications by switching the exogenous/endogenous status of shift variables is used frequently in ORANI type models and hence, maintained in PHILGEM . It helps to avoid proliferation of model variants, allowing the same TABLO Input file to contain different versions of some equations. The choice of which shift variables are exogenous determines at run time which version is operative in the rest of the model.

```
! Excerpt 18 of TABLO input file: !
! Government and inventory demands !
Variable
  f5tot # Overall shift term for government demands #;
  f5tot2 # Ratio between f5tot and x3tot #;
  (all,c,COM)(all,s,SRC) f5(c,s) # Government demand shift #;
  (change) (all,c,COM)(all,s,SRC) fx6(c,s) # Shifter on rule for stocks #;
Equation
  E_x5 # Government demands # (all,c,COM)(all,s,SRC) x5(c,s) = f5(c,s) + f5tot;
  E_f5tot # Overall government demands shift # f5tot = x3tot + f5tot2;
! note: normally ONE of f5tot and f5tot2 is exogenous
  f5tot2 exogenous gives x5(c,s) = f5(c,s) + x3tot ...gov follows hou
  f5tot exogenous gives x5(c,s) = f5(c,s) + f5tot ...gov exog !

Coefficient (all,c,COM)(all,s,SRC) LEVP0(c,s) # Levels basic prices #;
Formula (initial) (all,c,COM)(all,s,SRC) LEVP0(c,s) = 1; ! arbitrary setting !
Update (all,c,COM)(all,s,SRC) LEVP0(c,s) = p0(c,s);
Equation
  E_delx6 # Stocks follow domestic output # (all,c,COM)(all,s,SRC)
  100*LEVP0(c,s)*delx6(c,s) = V6BAS(c,s)*x0com(c) + fx6(c,s);
  E_delV6 # Update formula for stocks # (all,c,COM)(all,s,SRC)
  delV6(c,s) = 0.01*V6BAS(c,s)*p0(c,s) + LEVP0(c,s)*delx6(c,s);
```

Equation E_delx6 shows one way that delx6, the change in the volume of goods going to inventories, might be endogenized. It states that the percentage change in the volume of each commodity, domestic or imported, going to inventories, is the same as the percentage change in domestic production of that commodity. Like the previous equation, E_delx6 can be insulated from the rest of the equation system—by leaving fx6 endogenous. A chief purpose for this equation is to facilitate the *real homogeneity test* described in Appendix I.

Equation E_delV6 defines ordinary changes in the values of inventory demands. These are used above to update the V6BAS coefficient. The form of the equation arises because delx6 is an ordinary (rather than percentage) change variable. It may be derived by:

$$\Delta V = P.X \left(\frac{\Delta P}{P} + \frac{\Delta X}{X} \right) = V \frac{p}{100} + P\Delta X. \quad (30)$$

Notice that we are now required to define and update the levels price, P, i.e., we are obliged to specify units of measurement for quantities. In the TABLO code the coefficient LEVP0 contains current basic

prices. The initial values of its elements are set (arbitrarily) to 1 via an earlier 'Formula (Initial)' statement. Thus, the unit of measurement of delx6 is *base-period-dollars-worth*.

4.18. Demands for margins

The equations in Excerpt 19 indicate that, in the absence of technical change, demands for margins are proportional to the commodity flows with which the margins are associated. The 'a' variables allow for technical change in margins usage.

```
! Excerpt 19 of TABLO input file: !
! Margin demands !
Variable
(all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
      a1mar(c,s,i,m) # Intermediate margin tech change #;
(all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
      a2mar(c,s,i,m) # Investment margin tech change #;
(all,c,COM)(all,s,SRC)(all,m,MAR) a3mar(c,s,m) # Household margin tech change#;
(all,c,COM)(all,m,MAR)          a4mar(c,m)   # Export margin tech change #;
(all,c,COM)(all,s,SRC)(all,m,MAR) a5mar(c,s,m) # Governmnt margin tech change#;

Equation
E_x1mar # Margins to producers # (all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
      x1mar(c,s,i,m) = x1(c,s,i) + a1mar(c,s,i,m);
E_x2mar # Margins to investment # (all,c,COM)(all,s,SRC)(all,i,IND)(all,m,MAR)
      x2mar(c,s,i,m) = x2(c,s,i) + a2mar(c,s,i,m);
E_x3mar # Margins to households # (all,c,COM)(all,s,SRC)(all,m,MAR)
      x3mar(c,s,m)   = x3(c,s) + a3mar(c,s,m);
E_x4mar # Margins to exports # (all,c,COM)(all,m,MAR)
      x4mar(c,m)     = x4(c) + a4mar(c,m);
E_x5mar # Margins to government # (all,c,COM)(all,s,SRC)(all,m,MAR)
      x5mar(c,s,m)   = x5(c,s) + a5mar(c,s,m);
```

4.19. Formulae for sales aggregates

The next two excerpts are concerned with computing aggregate demands for domestic and imported commodities. Excerpt 20 contains coefficients and formulae for these demands. The aggregate SALE matrix breaks down demand for domestic and imported goods, valued at basic prices, by their main destinations. This is written to the SUMMARY file for later use in analysing results. Note that margin use is identified separately.

The VOIMP, SALES and DOMSALES aggregates are simply subtotals of the SALE matrix.

```
! Excerpt 20 of TABLO input file: !
! Sales Aggregates !
Coefficient (all,c,COM) MARSALLES(c) # Total usage for margins purposes #;
Formula
(all,n,NONMAR) MARSALLES(n) = 0.0;
(all,m,MAR)    MARSALLES(m) = sum{c,COM, V4MAR(c,m) +
      sum{s,SRC, V3MAR(c,s,m) + V5MAR(c,s,m) +
      sum{i,IND, V1MAR(c,s,i,m) + V2MAR(c,s,i,m)}}};

Set DEST # Sale Categories #
(Interm, Invest, HouseH, Export, GovGE, Stocks, Margins);

Coefficient (all,c,COM)(all,s,SRC)(all,d,DEST) SALE(c,s,d) # Sales aggregates #;
Formula
(all,c,COM)(all,s,SRC) SALE(c,s,"Interm")   = sum{i,IND, V1BAS(c,s,i)};
(all,c,COM)(all,s,SRC) SALE(c,s,"Invest")    = sum{i,IND, V2BAS(c,s,i)};
(all,c,COM)(all,s,SRC) SALE(c,s,"HouseH")    = V3BAS(c,s);
(all,c,COM)             SALE(c,"dom","Export") = V4BAS(c);
(all,c,COM)             SALE(c,"imp","Export") = 0;
(all,c,COM)(all,s,SRC) SALE(c,s,"GovGE")     = V5BAS(c,s);
(all,c,COM)(all,s,SRC) SALE(c,s,"Stocks")    = V6BAS(c,s);
(all,c,COM)             SALE(c,"dom","Margins") = MARSALLES(c);
```

```
(all,c,COM)          SALE(c,"imp","Margins") = 0;
```

```
Write SALE to file SUMMARY header "SALE";
```

```
Coefficient
```

```
(all,c,COM) V0IMP(c) # Total basic-value imports of good c #;
```

```
(all,c,COM) SALES(c) # Total sales of domestic commodities #;
```

```
(all,c,COM) DOMSALES(c) # Total sales to local market #;
```

```
Formula
```

```
(all,c,COM) V0IMP(c) = sum{d,DEST, SALE(c,"imp",d)}+ TINY;
```

```
(all,c,COM) SALES(c) = sum{d,DEST, SALE(c,"dom",d)}+ TINY;
```

```
(all,c,COM) DOMSALES(c) = SALES(c) - V4BAS(c);
```

4.20. Market-clearing equations

Excerpt 21 includes market-clearing equations for locally-consumed commodities, both domestic and imported. It begins by defining equations for the delSale matrix variable, which corresponds to the SALE coefficient just described. Each element of delSale shows the ordinary change in a quantity, measured in current prices. This is seen most clearly in the inventory (stocks) terms. The other right hand side terms have the form:

$$V.x/100 = P.X.x/100 = P\Delta x$$

Having constructed delSale, the next two equations are rather simple. E_p0A just adds up all non-export demand for domestic commodities: the sum is equal to the previously defined supply variable, x0dom. Similarly, E_x0imp computes the total demand for imports. By exogenously setting the world price of imports, pf0cif(c), we assume infinite elasticity of the supply of imports.

```
! Excerpt 21 of TABLO input file: !
```

```
! Market clearing equations !
```

```
Variable (change)
```

```
(all,c,COM)(all,s,SRC)(all,d,DEST) delSale(c,s,d) # Sales aggregates #;
```

```
(all,c,COM) x0imp(c) # Total supplies of imported goods #;
```

```
Equation
```

```
E_delSaleA (all,c,COM)(all,s,SRC) delSale(c,s,"Interm") =
```

```
0.01*sum{i,IND,V1BAS(c,s,i)*x1(c,s,i)};
```

```
E_delSaleB (all,c,COM)(all,s,SRC) delSale(c,s,"Invest") =
```

```
0.01*sum{i,IND,V2BAS(c,s,i)*x2(c,s,i)};
```

```
E_delSaleC (all,c,COM)(all,s,SRC) delSale(c,s,"HouseH")=0.01*V3BAS(c,s)*x3(c,s);
```

```
E_delSaleD (all,c,COM) delSale(c,"dom","Export")=0.01*V4BAS(c)*x4(c);
```

```
E_delSaleE (all,c,COM) delSale(c,"imp","Export")= 0;
```

```
E_delSaleF (all,c,COM)(all,s,SRC) delSale(c,s,"GovGE") =0.01*V5BAS(c,s)*x5(c,s);
```

```
E_delSaleG (all,c,COM)(all,s,SRC) delSale(c,s,"Stocks") = LEVP0(c,s)*delx6(c,s);
```

```
E_delSaleH (all,m,MAR) delSale(m,"dom","Margins") = 0.01*
```

```
sum{c,COM, V4MAR(c,m)*x4mar(c,m) ! note nesting of sum parentheses !
```

```
+ sum{s,SRC, V3MAR(c,s,m)*x3mar(c,s,m) + V5MAR(c,s,m)*x5mar(c,s,m)
```

```
+ sum{i,IND, V1MAR(c,s,i,m)*x1mar(c,s,i,m) + V2MAR(c,s,i,m)*x2mar(c,s,i,m)};}
```

```
E_delSaleI (all,n,NONMAR) delSale(n,"dom","Margins") = 0;
```

```
E_delSaleJ (all,c,COM) delSale(c,"imp","Margins") = 0;
```

```
Set LOCUSER # Non-export users #(Interm, Invest, HouseH, GovGE, Stocks,Margins);
```

```
Subset LOCUSER is subset of DEST;
```

```
Equation
```

```
E_p0A # Supply = Demand for domestic commodities # (all,c,COM)
```

```
0.01*[TINY+DOMSALES(c)]*x0dom(c)=sum{u,LOCUSER,delSale(c,"dom",u)};
```

```
E_x0imp # Import volumes # (all,c,COM)
```

```
0.01*[TINY+V0IMP(c)]*x0imp(c) = sum{u,LOCUSER,delSale(c,"imp",u)};
```


4.21. Purchasers' prices

The equations in Excerpt 22 define purchasers' prices for each of the first five user groups: producers; investors; households; exports; and government. Purchasers' prices (in levels) are the sums of basic values, sales taxes and margins. Sales taxes are treated as *ad valorem* on basic values, with the sales-tax variables t in the linearised model being percentage changes in the powers of taxes (The power of a tax is one plus the *ad valorem* rate). For example, equation E_p3 is derived from the levels form:

$$X3(c,s)*P3(c,s) = X3(c,s)*P0(c,s)*T3(c,s) + \text{sum}\{m, \text{MAR}, X3\text{MAR}(c,s,m)*P0(m, \text{"dom"})\}.$$

In percentage-change form this is:

$$\begin{aligned} V3\text{PUR}(c,s)*\{x3(c,s) + p3(c,s)\} &= \{V3\text{TAX}(c,s)+V3\text{BAS}(c,s)\}*\{x3(c,s)+p0(c,s)+t3(c,s)\} \\ &+ \text{sum}\{m, \text{MAR}, V3\text{MAR}(c,s,m)*[x3\text{mar}(c,s,m)+p0(m, \text{"dom"})]\}. \end{aligned}$$

By using Equation E_x3mar from Excerpt 19 to eliminate $x3\text{mar}(c,s,m)$, we can cancel out the $x3(c,s)$ terms to obtain:

$$\begin{aligned} V3\text{PUR}(c,s)*p3(c,s) &= [V3\text{BAS}(c,s)+V3\text{TAX}(c,s)]*[p0(c,s)+ t3(c,s)] \\ &+ \text{sum}\{m, \text{MAR}, V3\text{MAR}(c,s,m)*[p0(m, \text{"dom"})+a3\text{mar}(c,s,m)]\}. \end{aligned}$$

For a commodity which is not used by households, $V3\text{PUR}(c,s)$ and its constituents would all be zero, leaving $p3(i,s)$ undefined. To finesse this problem, the TABLO file adds the coefficient TINY to $V3\text{PUR}(c,s)$ so that if it is zero, equation E_p3 becomes:

$$p3(c,s) = 0.$$

The same procedure is used for the purchasers'-price equations referring to intermediate, investment, export and government users.

The comment at the end shows how the final equation would appear if the $t5$ variable (% change in power) was replaced by $\text{delt}5$: an ordinary change in the *ad valorem* rate.

```
! Excerpt 22 of TABLO input file: !
! Purchasers prices !
Variable ! Powers of Commodity Taxes on Basic Flows !
(all,c,COM)(all,s,SRC)(all,i,IND) t1(c,s,i) # Power of tax on intermediate #;
(all,c,COM)(all,s,SRC)(all,i,IND) t2(c,s,i) # Power of tax on investment #;
(all,c,COM)(all,s,SRC)          t3(c,s)   # Power of tax on household #;
(all,c,COM)                      t4(c)    # Power of tax on export #;
(all,c,COM)(all,s,SRC)          t5(c,s)   # Power of tax on government #;

Equation
E_p1 # Purchasers prices - producers # (all,c,COM)(all,s,SRC)(all,i,IND)
[V1PUR(c,s,i)+TINY]*p1(c,s,i) = [V1BAS(c,s,i)+V1TAX(c,s,i)]*
[p0(c,s)+ t1(c,s,i)] +
sum{m,MAR, V1MAR(c,s,i,m)*
[p0dom(m)+a1mar(c,s,i,m)]};

E_p2 # Purchasers prices - capital creators # (all,c,COM)(all,s,SRC)(all,i,IND)
[V2PUR(c,s,i)+TINY]*p2(c,s,i) = [V2BAS(c,s,i)+V2TAX(c,s,i)]*
[p0(c,s)+ t2(c,s,i)] +
sum{m,MAR, V2MAR(c,s,i,m)*
[p0dom(m)+a2mar(c,s,i,m)]};

E_p3 # Purchasers prices - households # (all,c,COM)(all,s,SRC)
[V3PUR(c,s)+TINY]*p3(c,s) = [V3BAS(c,s)+V3TAX(c,s)]*[p0(c,s)+ t3(c,s)] +
sum{m,MAR, V3MAR(c,s,m)*[p0dom(m)+a3mar(c,s,m)]};

E_p4 # Zero pure profits in exporting # (all,c,COM)
[V4PUR(c)+TINY]*p4(c) = [V4BAS(c)+V4TAX(c)]*[pe(c)+ t4(c)] +
sum{m,MAR, V4MAR(c,m)*[p0dom(m)+a4mar(c,m)]};

! note that we refer to export taxes, not subsidies !
```

```

E_p5 # Zero pure profits in distribution to government # (all,c,COM)(all,s,SRC)
[V5PUR(c,s)+TINY]*p5(c,s) = [V5BAS(c,s)+V5TAX(c,s)]*[p0(c,s)+ t5(c,s)] +
sum{m,MAR, V5MAR(c,s,m)*[p0dom(m)+a5mar(c,s,m)]];

! alternate form
E_p5q # Zero pure profits in distribution of government# (all,c,COM)(all,s,SRC)
[V5PUR(c,s)+TINY]*p5(c,s) = [V5BAS(c,s)+V5TAX(c,s)]*p0(c,s) +
100*V5BAS(c,s)*delt5(c,s) +
sum{m,MAR, V5MAR(c,s,m)*[p0dom(m)+a5mar(c,s,m)]];!

```

4.22. Indirect taxes

Similar to ORANI-G, PHILGEM allows for great flexibility in the treatment of indirect taxes. However, it is cumbersome to shock 3-dimensional variables such as t_1 and t_2 directly, because they have so many elements. Besides, most projected changes to tax rates have a fairly simple structure. For example, an increase in the tax on petrol might raise the price to all users by 10%. To simulate a tax change like this, it helps to include equations which implement the tax regime to be simulated. The effect is to replace multi-dimensional exogenous tax variables with vectors that are easier to shock.

Excerpt 23 contains default rules for setting sales-tax rates for producers, investors, households, and government. Sales taxes are treated as *ad valorem* on basic values, with the sales-tax variables in the linearised model being percentage changes in the powers of the taxes. Each equation allows the changes in the relevant tax rates to be commodity-specific or user-specific. To simulate more complex patterns of tax changes, we would omit or modify these equations.¹⁸

```

! Excerpt 23 of TABLO input file: !
! Tax rate equations !
Variable
f1tax_csi    # Uniform % change in powers of taxes on intermediate usage #;
f2tax_csi    # Uniform % change in powers of taxes on investment #;
f3tax_cs     # Uniform % change in powers of taxes on household usage #;
f4tax_ntrad  # Uniform % change in powers of taxes on nontradtnl exports #;
f4tax_trad   # Uniform % change in powers of taxes on tradtnl exports #;
f5tax_cs     # Uniform % change in powers of taxes on government usage #;
(all,c,COM) f0tax_s(c)      # General sales tax shifter #;

Equation
E_t1 # Power of tax on sales to intermediate #
(all,c,COM)(all,s,SRC)(all,i,IND) t1(c,s,i) = f0tax_s(c) + f1tax_csi;
E_t2 # Power of tax on sales to investment #
(all,c,COM)(all,s,SRC)(all,i,IND) t2(c,s,i) = f0tax_s(c) + f2tax_csi;
E_t3 # Power of tax on sales to households #
(all,c,COM)(all,s,SRC) t3(c,s) = f0tax_s(c) + f3tax_cs;
E_t4A # Power of tax on sales to individual exports #
(all,c,TRADEXP) t4(c) = f0tax_s(c) + f4tax_trad;
E_t4B # Power of tax on sales to collective exports #
(all,c,NTRADEXP) t4(c) = f0tax_s(c) + f4tax_ntrad;
E_t5 # Power of tax on sales to government #
(all,c,COM)(all,s,SRC) t5(c,s) = f0tax_s(c) + f5tax_cs;

```

The equations of Excerpt 24 compute ordinary changes in the indirect tax flows. Their main use is in the Update statements for the various commodity tax matrices (see above). To understand the equations, note that the bases for the sales taxes are the basic values of the corresponding commodity flows, and the tax-rate variables appearing in the model are powers of the sales-tax rates. Hence, for any component of

¹⁸ The same problem, that exogenous variables have too many dimensions, sometimes arises when we wish to simulate technical change. Suppose we wished to shock the variable a_1 (which varies by commodity, source and industry). It might be simplest to write a new equation, like the equations of Excerpt 28, that related a_1 to one or more new vector variables. The modeller is expected to add these equations and variables as required, since their form is experiment-specific.

sales tax, we can express revenue (VTAX), in levels, as the product of the base (VBAS) and the power of the tax (T) minus one, i.e.,

$$VTAX = VBAS(T-1). \quad (31)$$

Hence: $\Delta VTAX = \Delta VBAS(T-1) + VBAS\Delta T$,

$$\begin{aligned} &= VBAS(T-1) \frac{\Delta VBAS}{VBAS} + VBAS * T \frac{\Delta T}{T}, \\ &= VBAS(T-1) * wbas / 100 + VBAS * T * t / 100, \\ &= VTAX * wbas / 100 + (VBAS + VTAX) t / 100, \end{aligned}$$

where wbas and t are percentage changes in VBAS and T. VBAS is in turn a product of a quantity (X) and a basic price (P), so its percentage change wbas can be written as (x + p). Hence:

$$\Delta VTAX = VTAX(x+p) / 100 + (VBAS + VTAX) t / 100 \quad (32)$$

The RHS shows two components of revenue change; due to changes in the base and rate respectively.

```
! Excerpt 24 of TABLO input file: !
! Update formulae for commodity taxes !
Equation
E_delV1TAX (all,c,COM)(all,s,SRC)(all,i,IND)
  delV1TAX(c,s,i) = 0.01*V1TAX(c,s,i)* [x1(c,s,i) + p0(c,s)] +
    0.01*[V1BAS(c,s,i)+V1TAX(c,s,i)]*t1(c,s,i);
E_delV2TAX (all,c,COM)(all,s,SRC)(all,i,IND)
  delV2TAX(c,s,i) = 0.01*V2TAX(c,s,i)* [x2(c,s,i) + p0(c,s)] +
    0.01*[V2BAS(c,s,i)+V2TAX(c,s,i)]*t2(c,s,i);
E_delV3TAX (all,c,COM)(all,s,SRC)
  delV3TAX(c,s) = 0.01*V3TAX(c,s)* [x3(c,s) + p0(c,s)] +
    0.01*[V3BAS(c,s)+V3TAX(c,s)]*t3(c,s);
E_delV4TAX (all,c,COM)
  delV4TAX(c) = 0.01*V4TAX(c)* [x4(c) + pe(c)] +
    0.01*[V4BAS(c)+V4TAX(c)]*t4(c);
E_delV5TAX (all,c,COM)(all,s,SRC)
  delV5TAX(c,s) = 0.01*V5TAX(c,s)*[x5(c,s) + p0(c,s)] +
    0.01*[V5BAS(c,s)+V5TAX(c,s)]*t5(c,s);

! alternate form
E_delV5TAX (all,c,COM)(all,s,SRC)
  delV5TAX(c,s) = 0.01*V5TAX(c,s)*[x5(c,s) + p0(c,s)] +
    V5BAS(c,s)*delt5(c,s); !
```

The comment at the end shows how the final equation would appear if the t5 variable (% change in power) was replaced by delt5: an ordinary change in the ad valorem rate.

In the next excerpt, equation E_p0B relates the domestic-currency prices of imports (c.i.f., duty-paid) to their foreign-currency prices. It is derived from the levels form:

$$P0(c,"imp") = PF0CIF(c)*PHI*TOIMP(c). \quad (33)$$

The other equation computes changes in tariff revenues, used to update V0TAR.

```
! Excerpt 25 of TABLO input file: !
! Import prices and tariff revenue !
Variable
(all,c,COM) pf0cif(c) # C.I.F. foreign currency import prices #;
(all,c,COM) t0imp(c) # Power of tariff #;

Equation
E_p0B # Zero pure profits in importing # (all,c,COM)
  p0(c,"imp") = pf0cif(c) + phi + t0imp(c);

E_delV0TAR # Tariff revenue# (all,c,COM)
  delV0TAR(c) = 0.01*V0TAR(c)*[x0imp(c)+pf0cif(c)+phi] +
    0.01*V0IMP(c)*t0imp(c);
```

Excerpt 26 computes various aggregates of indirect tax revenue—both in the levels and in ordinary change form. Both equations and formulae are simple add-ups.

```

! Excerpt 26 of TABLO input file: !
! Indirect tax revenue aggregates !
Coefficient
V1TAX_CSI # Total intermediate tax revenue #;
V2TAX_CSI # Total investment tax revenue #;
V3TAX_CS  # Total households tax revenue #;
V4TAX_C   # Total export tax revenue #;
V5TAX_CS  # Total government tax revenue #;
V0TAR_C   # Total tariff revenue #;

Formula
V1TAX_CSI = sum{c,COM, sum{s,SRC, sum{i,IND, V1TAX(c,s,i)}}};
V2TAX_CSI = sum{c,COM, sum{s,SRC, sum{i,IND, V2TAX(c,s,i)}}};
V3TAX_CS  = sum{c,COM, sum{s,SRC, V3TAX(c,s)}};
V4TAX_C   = sum{c,COM, V4TAX(c)};
V5TAX_CS  = sum{c,COM, sum{s,SRC, V5TAX(c,s)}};
V0TAR_C   = sum{c,COM, V0TAR(c)};

Variable
(change) delV1tax_csi # Aggregate revenue from indirect taxes on intermediate #;
(change) delV2tax_csi # Aggregate revenue from indirect taxes on investment #;
(change) delV3tax_cs  # Aggregate revenue from indirect taxes on households #;
(change) delV4tax_c   # Aggregate revenue from indirect taxes on export #;
(change) delV5tax_cs  # Aggregate revenue from indirect taxes on government #;
(change) delV0tar_c   # Aggregate tariff revenue #;

Equation
E_delV1tax_csi delV1tax_csi = sum{c,COM,sum{s,SRC,sum{i,IND, delV1TAX(c,s,i)}}};
E_delV2tax_csi delV2tax_csi = sum{c,COM,sum{s,SRC,sum{i,IND, delV2TAX(c,s,i)}}};
E_delV3tax_cs  delV3tax_cs  = sum{c,COM,sum{s,SRC, delV3TAX(c,s)}};
E_delV4tax_c   delV4tax_c   = sum{c,COM, delV4TAX(c)};
E_delV5tax_cs  delV5tax_cs  = sum{c,COM,sum{s,SRC,delV5TAX(c,s)}};
E_delV0tar_c   delV0tar_c   = sum{c,COM, delV0TAR(c)};

```

4.23. GDP from the income and expenditure sides

Excerpt 27 defines the nominal aggregates which make up GDP from the income side. These include totals of factor payments, the value of other costs, and the total yield from commodity taxes. Their derivation is straightforward. For example, the formula for total land revenue:

$$V1LND_I = \text{Sum}\{i, \text{IND}, V1LND(i)\} = \text{Sum}\{i, \text{IND}, X1LND(i) * P1LND(i)\},$$

becomes, in percentage-change form:

$$V1LND_I * w1lnd_i = \text{Sum}\{i, \text{IND}, V1LND(i) * [x1lnd(i) + p1lnd(i)]\}.$$

```

! Excerpt 27 of TABLO input file: !
! Factor incomes and GDP !
Coefficient
V1CAP_I # Total payments to capital #;
V1LAB_IO # Total payments to labour #;
V1LND_I # Total payments to land #;
V1PTX_I # Total production tax/subsidy #;
V1OCT_I # Total other cost ticket payments #;
V1PRIM_I # Total primary factor payments #;
V0GDPINC # Nominal GDP from income side #;
V0TAX_CSI # Total indirect tax revenue #;

Formula
V1CAP_I = sum{i,IND, V1CAP(i)};

```

```

V1LAB_IO = sum{i,IND, V1LAB_O(i)};
V1LND_I  = sum{i,IND, V1LND(i)};
V1PTX_I  = sum{i,IND, V1PTX(i)};
V1OCT_I  = sum{i,IND, V1OCT(i)};
V1PRIM_I = V1LAB_IO + V1CAP_I + V1LND_I;
V0TAX_CSI = V1TAX_CSI + V2TAX_CSI + V3TAX_CS + V4TAX_C + V5TAX_CS + V0TAR_C +
            V1OCT_I + V1PTX_I;
V0GDPINC = V1PRIM_I + V0TAX_CSI;

```

Variable

```

w1lab_io # Aggregate payments to labour #;
w1cap_i  # Aggregate payments to capital #;
w1lnd_i  # Aggregate payments to land #;
w1prim_i # Aggregate primary factor payments #;
w1oct_i  # Aggregate "other cost" ticket payments #;
w0tax_csi # Aggregate revenue from all indirect taxes #;
w0gdpinc # Nominal GDP from income side #;
(change) delV1PTX_i # Ordinary change in all-industry production tax revenue #;
(change) delV0tax_csi # Aggregate revenue from all indirect taxes #;

```

Equation

```

E_w1lab_io # Aggregate payments to labour #
V1LAB_IO*w1lab_io = sum{i,IND, sum{o,OCC, V1LAB(i,o)*[x1lab(i,o)+p1lab(i,o)]}};

E_w1cap_i # Aggregate payments to capital #
V1CAP_I*w1cap_i = sum{i,IND, V1CAP(i)*[x1cap(i)+p1cap(i)]};

E_w1lnd_i # Aggregate payments to land #
ID01[V1LND_I]*w1lnd_i = sum{i,IND, V1LND(i)*[x1lnd(i)+p1lnd(i)]};

E_w1prim_i # Aggregate primary factor payments #
V1PRIM_I*w1prim_i = V1LAB_IO*w1lab_io + V1CAP_I*w1cap_i + V1LND_I*w1lnd_i;

E_w1oct_i # Aggregate "other cost" ticket payments #
ID01[V1OCT_I]*w1oct_i = sum{i,IND, V1OCT(i)*[x1oct(i)+p1oct(i)]};

E_delV1PTX_i # Ordinary change - all-industry production tax revenue #
delV1PTX_i = sum{i,IND, delV1PTX(i)};

E_delV0tax_csi # Ordinary change - aggregate revenue from all indirect taxes #
delV0tax_csi = delV1tax_csi + delV2tax_csi + delV3tax_cs + delV4tax_c +
              delV5tax_cs + delV0tar_c + delV1PTX_i + 0.01*V1OCT_I*w1oct_i;

E_w0tax_csi # Aggregate revenue from all indirect taxes #
[TINY+V0TAX_CSI]*w0tax_csi = 100*delV0tax_csi;

E_w0gdpinc # Nominal GDP from income side #
V0GDPINC*w0gdpinc = V1PRIM_I*w1prim_i + 100*delV0tax_csi;

```

Because TABLO does not distinguish upper and lower case, we cannot use 'v1lnd_i' to refer to the change in V1LND_I—instead we use 'w1lnd_i'. This conflict arises only for aggregate flows, since only these flows appear simultaneously as variables and coefficients.

Excerpt 28 defines the aggregates which make up GDP from the expenditure side. We could have computed percentage changes in the nominal aggregates as in the previous section. For example, in equation E_w2tot_i, total nominal investment could have been written as:

$$V2TOT_I * w2tot_i = \text{Sum}\{i, \text{IND}, V2TOT(i) * [x2tot(i) + p2tot(i)]\}.$$

We choose to decompose this change into price and quantity components—see equations E_p2tot_i and E_x2tot_i. The nominal percentage change is the sum of percentage changes in these two Divisia indices.

For investment, and for each other expenditure component of GDP, we define a quantity index and a price index which sum to the (percentage change in the) nominal value of the aggregate. We weight

these together to form expenditure-side measures of percentage changes in real GDP, the GDP deflator and nominal GDP.

```
! Excerpt 28 of TABLO input file: !
! GDP expenditure aggregates !
Coefficient ! Expenditure Aggregates at Purchaser's Prices !
(all,c,COM) V0CIF(c) # Total ex-duty imports of good c #;
      V0CIF_C # Total local currency import costs, excluding tariffs #;
      V0IMP_C # Total basic-value imports (includes tariffs) #;
      V2TOT_I # Total investment usage #;
      V4TOT   # Total export earnings #;
      V5TOT   # Total value of government demands #;
      V6TOT   # Total value of inventories #;
      V0GNE   # GNE from expenditure side #;
      V0GDPEXP # GDP from expenditure side #;
```

Formula

```
(all,c,COM) V0CIF(c) = V0IMP(c) - V0TAR(c);
      V0CIF_C = sum{c,COM, V0CIF(c)};
      V0IMP_C = sum{c,COM, V0IMP(c)};
      V2TOT_I = sum{i,IND, V2TOT(i)};
      V4TOT   = sum{c,COM, V4PUR(c)};
      V5TOT   = sum{c,COM, sum{s,SRC, V5PUR(c,s)}};
      V6TOT   = sum{c,COM, sum{s,SRC, V6BAS(c,s)}};
      V0GNE   = V3TOT + V2TOT_I + V5TOT + V6TOT;
      V0GDPEXP = V0GNE + V4TOT - V0CIF_C;
```

Variable

```
x2tot_i # Aggregate real investment expenditure #;
p2tot_i # Aggregate investment price index #;
w2tot_i # Aggregate nominal investment #;
```

Equation

```
E_x2tot_i V2TOT_I*x2tot_i = sum{i,IND, V2TOT(i)*x2tot(i)};
E_p2tot_i V2TOT_I*p2tot_i = sum{i,IND, V2TOT(i)*p2tot(i)};
E_w2tot_i w2tot_i = x2tot_i + p2tot_i;
```

Variable

```
x4tot # Export volume index #;
p4tot # Exports price index, local currency #;
w4tot # Local currency border value of exports #;
```

Equation

```
E_x4tot V4TOT*x4tot = sum{c,COM, V4PUR(c)*x4(c)};
E_p4tot V4TOT*p4tot = sum{c,COM, V4PUR(c)*p4(c)};
E_w4tot w4tot = x4tot + p4tot;
```

Variable

```
x5tot # Aggregate real government demands #;
p5tot # Government price index #;
w5tot # Aggregate nominal value of government demands #;
```

Equation

```
E_x5tot V5TOT*x5tot = sum{c,COM, sum{s,SRC, V5PUR(c,s)*x5(c,s)}};
E_p5tot V5TOT*p5tot = sum{c,COM, sum{s,SRC, V5PUR(c,s)*p5(c,s)}};
E_w5tot w5tot = x5tot + p5tot;
```

Variable

```
x6tot # Aggregate real inventories #;
p6tot # Inventories price index #;
w6tot # Aggregate nominal value of inventories #;
```

Equation

```
E_x6tot [TINY+V6TOT]*x6tot = 100*sum{c,COM, sum{s,SRC, LEVP0(c,s)*delx6(c,s)}};
E_p6tot [TINY+V6TOT]*p6tot = sum{c,COM, sum{s,SRC, V6BAS(c,s)*p0(c,s)}};
E_w6tot w6tot = x6tot + p6tot;
```

Variable

```

x0cif_c    # Import volume index, C.I.F. weights #;
p0cif_c    # Imports price index, C.I.F., local currency #;
w0cif_c    # C.I.F. local currency value of imports #;
Equation
E_x0cif_c  V0CIF_C*x0cif_c = sum{c,COM, V0CIF(c)*x0imp(c)};
E_p0cif_c  V0CIF_C*p0cif_c = sum{c,COM, V0CIF(c)*[phi+pf0cif(c)]};
E_w0cif_c  w0cif_c = x0cif_c + p0cif_c;

Variable
x0gne      # Real GNE #;
p0gne      # GNE price index #;
w0gne      # Nominal GNE #;
Equation
E_x0gne    V0GNE*x0gne = V3TOT*x3tot + V2TOT_I*x2tot_i + V5TOT*x5tot + V6TOT*x6tot;
E_p0gne    V0GNE*p0gne = V3TOT*p3tot + V2TOT_I*p2tot_i + V5TOT*p5tot + V6TOT*p6tot;
E_w0gne    w0gne = x0gne + p0gne;

Variable
x0gdpepx  # Real GDP from expenditure side #;
p0gdpepx  # GDP price index, expenditure side #;
w0gdpepx  # Nominal GDP from expenditure side #;
Equation
E_x0gdpepx x0gdpepx = [1/V0GDPEXP]*[V3TOT*x3tot + V2TOT_I*x2tot_i + V5TOT*x5tot
+ V6TOT*x6tot + V4TOT*x4tot - V0CIF_C*x0cif_c];
E_p0gdpepx p0gdpepx = [1/V0GDPEXP]*[V3TOT*p3tot + V2TOT_I*p2tot_i + V5TOT*p5tot
+ V6TOT*p6tot + V4TOT*p4tot - V0CIF_C*p0cif_c];
E_w0gdpepx w0gdpepx = x0gdpepx + p0gdpepx;

```

Superficially, price and quantity components such as $p2tot_i$ and $x2tot_i$ resemble the price and quantity indices which arise from the nested production functions of agents. The latter Divisia indices arise from homothetic functional forms. However, the model contains no analogous function to aggregate investment quantities across industries. Similarly, our definition of real consumption is not derived from the household utility function. We use these price-quantity decompositions only as convenient summary measures¹⁹.

It is an accounting identity that GDP from the expenditure and income sides must be equal, both in the levels and in percentage changes. That is:

$$V0GDPEXP \equiv V0GDPINC, \text{ and } w0gdpepx \equiv w0gdpinc. \quad (34)$$

Nonetheless, we find it useful to compute and print these values separately as a check on the model's accounting relations.

4.24. The trade balance and other aggregates

Because zero is a plausible base-period value, the balance of trade is computed in the first equation in Excerpt 29 as an ordinary change (delB), not a percentage change. We avoid choosing units by expressing this change as a fraction of GDP. Note that delB is the [change in the] ratio of two nominal variables.

The next three equations measure percentage changes in imports at tariff-inclusive prices. Finally, the excerpt contains measures of percentage changes in the terms of trade and the real exchange rate.

```

! Excerpt 29 of TABLO input file: !
! Trade balance and other indices !
Variable
(change) delB # (Nominal balance of trade)/{nominal GDP} #;
x0imp_c # Import volume index, duty-paid weights #;
w0imp_c # Value of imports plus duty #;

```

¹⁹ There is indeed no levels equation corresponding to our change definition of, say, the investment price index. A levels formulation could use either initial or final weights; our formula uses weights that vary continuously between these two values. Because our formula for $p2tot_i$ can only be written in change form, results for that variable suffer from path-dependence: they depend slightly on details of the solution algorithm that should be irrelevant. The effect however is normally very small, and rarely propagates through to other equations.

```

    p0imp_c # Duty-paid imports price index, local currency #;
    p0realdev # Real devaluation #;
    p0toft # Terms of trade #;
Equation
E_de1B # (Nominal balance of trade)/{nominal GDP} #
    100*V0GDPEXP*de1B = V4TOT*w4tot -V0CIF_C*w0cif_c-[V4TOT-V0CIF_C]*w0gdpexp;

E_x0imp_c # Import volume index, duty-paid weights #
    x0imp_c = sum{c,COM, [V0IMP(c)/V0IMP_C]*x0imp(c)};

E_p0imp_c # Duty-paid imports price index, local currency #
    p0imp_c = sum{c,COM, [V0IMP(c)/V0IMP_C]*p0(c,"imp")};

E_w0imp_c # Value of imports plus duty #
    w0imp_c = x0imp_c + p0imp_c;

E_p0toft # Terms of trade #
    p0toft = p4tot - p0cif_c;

E_p0realdev # Real devaluation #
    p0realdev = p0cif_c - p0gdpexp;

```

4.25. Primary factor aggregates

Excerpt 30 begins by computing some quantity measures of employment, capital usage, and GDP at factor cost. These aggregates indicate the aggregate productive capacities of the relevant factors. For example, in computing the aggregate employment measures, we use wage-bill weights, reflecting the relative marginal products of different workers. Different aggregates could be defined, using as weights hours worked, or numbers employed²⁰.

Next computed are three indices of primary factor prices.

```

! Excerpt 30 of TABLO input file: !
! Primary factor aggregates !
Variable
(all,i,IND) employ(i) # Employment by industry: wage bill weights #;
    employ_i # Aggregate employment: wage bill weights #;
    x1cap_i # Aggregate capital stock, rental weights #;
    x1lnd_i # Aggregate land stock, rental weights #;
    x1prim_i # Aggregate primary factor use (excludes tech change) #;
    p1prim_i # Index of factor cost (excludes tech change) #;
    p1lab_io # Average nominal wage #;
    realwage # Average real wage #;
    p1cap_i # Average capital rental #;
    p1lnd_i # Average land rental #;

Equation
E_employ # Employment by industry: wage bill weights # (all,i,IND)
    ID01[V1LAB_O(i)]*employ(i) = sum{o,OCC, V1LAB(i,o)*x1lab(i,o)};

E_employ_i # Aggregate employment: wage bill weights #
    V1LAB_IO*employ_i = sum{i,IND, V1LAB_O(i)*employ(i)};

E_x1cap_i # Aggregate capital stock, rental weights #
    V1CAP_I*x1cap_i = sum{i,IND, V1CAP(i)*x1cap(i)};

E_x1lnd_i # Aggregate land stock, rental weights #
    ID01[V1LND_I]*x1lnd_i = sum{i,IND, V1LND(i)*x1lnd(i)};

```

²⁰ The TABLO input file contains optional code, not shown here, which could be used to compute such indices, data permitting.


```

E_x1prim_i # Aggregate primary factor use (excludes tech change) #
  V1PRIM_I*x1prim_i = [V1LAB_IO*employ_i + V1CAP_I*x1cap_i + V1LND_I*x1lnd_i];

E_p1lab_io # Average nominal wage #
  V1LAB_IO*p1lab_io = sum{i,IND, sum{o,OCC, V1LAB(i,o)*p1lab(i,o)}};

E_realwage # Average real wage #
  realwage = p1lab_io - p3tot;

E_p1cap_i # Average capital rental #
  V1CAP_I*p1cap_i = sum{i,IND, V1CAP(i)*p1cap(i)};

E_p1lnd_i # Average land rental #
  ID01[V1LND_I]*p1lnd_i = sum{i,IND, V1LND(i)*p1lnd(i)};

E_p1prim_i # Index of factor cost (excludes tech change) #
  V1PRIM_I*p1prim_i = V1LAB_IO*p1lab_io + V1CAP_I*p1cap_i + V1LND_I*p1lnd_i;

![[! Optional addition: employment aggregates weighted by numbers employed !
Coefficient
(all,i,IND)(all,o,OCC) WORKERS(i,o) # numbers employed #;
(all,i,IND)(all,o,OCC) UNITWAGE(i,o) # wagebill/numbers employed #;
Read WORKERS from file BASEDATA header "WORK";
Update (all,i,IND)(all,o,OCC) WORKERS(i,o) = x1lab(i,o);
Formula (all,i,IND)(all,o,OCC) UNITWAGE(i,o) = V1LAB(i,o)/WORKERS(i,o);
Write UNITWAGE to file SUMMARY header "UWAG";

Variable
(all,i,IND) workers_o(i) # employment by industry (persons weights average) #;
(all,o,OCC) workers_i(o) # employment by skill (persons weights average) #;
      workers_io # aggregate employment (persons weights average) #;

Equation
E_workers_o (all,i,IND) sum{o,OCC, WORKERS(i,o)*[x1lab(i,o)-workers_o(i)]}=0;
E_workers_i (all,o,OCC) sum{i,IND, WORKERS(i,o)*[x1lab(i,o)-workers_i(o)]}=0;
E_workers_io Sum{o,OCC, sum{i,IND, WORKERS(i,o)*[x1lab(i,o)-workers_io]}=0;
!]]!

```

4.26. Rates of return and investment

Excerpt 31 contains equations determining $x2_{tot}$, the amount of new capital stock created for each industry. Following ORANI-G, PHILGEM contains 3 alternate investment rules, one of which will be active for each industry. The active rule either determines industry investment, or (if aggregate investment is fixed) it determines that industry's share in aggregate investment.

Rule 1 relates the creation of new capital stock in each industry to profitability in that industry. It follows the original ORANI computer model. For a fuller explanation, the reader is referred to DPSV, Section 19. The effect of the equation is that industries which become more profitable (gret(i) rising) attract more investment (higher ggro(i)).

Equation E_{finv1} relates the investment/capital ratio to the rate of return (relative to the variable $invslack$, which is interpretable here as an economy-wide rate of return). It is to be interpreted as a risk-related relationship with relatively fast- (slow-) growing industries requiring premia (accepting discounts) on their rates of return. If exogenous, the variable $finv1(i)$ could be shocked to allow for additional shifts in investment. If, for some industry, $finv1(i)$ is endogenous (and $x2_{tot}(i)$ is determined in some other way), equation E_{finv1} serves only to determine $finv1(i)$, which appears in no other equation.

Rule 2, given by Equation E_{finv2} , may be used to determine investment in those industries for which Rule 1 is deemed inappropriate. These might be industries where investment is determined by government policy. For such an industry the corresponding element of $finv2$ should be exogenous, causing that industry's investment to follow the national trend. For the other industries, E_{finv2} just determines $finv2$.

Rule 3, given by Equation E_finv3, is intended for long run simulations. For an industry with fixed finv3, it fixes gross capital growth rates (ggro=I/K ratios). The effect is that investment follows the industry capital stock.

If the variable invslack is exogenous, the preceding equations will determine investment by industry, and therefore in aggregate. However, many short-run simulations have aggregate investment exogenous. This is made possible by leaving the variable invslack endogenous, with x2tot_i exogenous. Or, using equation E_f2tot, link aggregate investment to aggregate real consumption by holding f2tot exogenous with invslack endogenous.

```

! Excerpt 31 of TABLO input file: !
! Investment equations !
Variable
  (all,i,IND) ggro(i) # Gross growth rate of capital = Investment/capital #;
  (all,i,IND) gret(i) # Gross rate of return = Rental/[Price of new capital] #;
Equation
  E_ggro (all,i,IND) ggro(i) = x2tot(i) - x1cap(i);
  E_gret (all,i,IND) gret(i) = p1cap(i) - p2tot(i);

! Three alternative rules for investment:
  Choose which applies to each industry by setting JUST ONE of
  the corresponding elements of x2tot, finv1, finv2, or finv3 exogenous.
  Iff aggregate investment x2tot_i is exogenous, invslack must be endogenous. !
Variable
  (all,i,IND) finv1(i)# Shifter to enforce DPSV investment rule #;
  (all,i,IND) finv2(i)# Shifter for "exogenous" investment rule #;
  (all,i,IND) finv3(i)# Shifter for longrun investment rule #;
  invslack # Investment slack variable for exogenizing aggregate investment #;

! Rule 1: Follows Section 19 of DPSV. The ratios Q and G are treated as
  parameters, just as in the original ORANI implementation. Attempts to
  improve the theory by updating these parameters have been found to
  occasionally lead to perversely signed coefficients !
Equation E_finv1 # DPSV investment rule #
  (all,i,IND) ggro(i) = finv1(i) + 0.33*[2.0*gret(i) - invslack];
! Note: above equation comes from substituting together DPSV
  equations 19.7-9. The value 0.33 and 2.0 correspond to the DPSV ratios
  [1/G.Beta] and Q (= ratio, gross to net rate of return) and are typical
  values of this ratio. In DPSV invslack was called "omega" and was interpreted
  as the "economy-wide rate of return" !

! Rule 2: For industries where investment is not mainly driven by current
  profits (eg, Education) make investment follow aggregate investment. !
Equation E_finv2 # Alternative rule for "exogenous" investment industries #
  (all,i,IND) x2tot(i) = x2tot_i + finv2(i);
! NB: you must not set ALL of finv2 exogenous else above would conflict with
  Equation E_x2tot_i !

! Rule 3: longrun investment rule: investment/capital ratios are exogenous !
Equation E_finv3 # Alternative long-run investment rule #
  (all,i,IND) ggro(i) = finv3(i) + invslack;

Variable f2tot # Ratio, investment/consumption #;
Equation E_f2tot x2tot_i = x3tot + f2tot;
! set f2tot exogenous and invslack endogenous
  to link aggregate investment to real consumption !

! Mechanism to allow fixed total capital to flow between sectors !
Variable
  (all,i,IND) fgret(i) # Shifter to lock together industry rates of return #;
  capslack # Slack variable to allow fixing aggregate capital #;
Equation E_fgret # Equation to force rates of return to move together #
  (all,i,IND) gret(i) = fgret(i) + capslack;

```

```

! normally, caps Slack is exogenous and fgret endogenous, so above just
! determines fgret. To allow capital to be mobile between sectors, yet
! fixed in aggregate, swap [newly exogenous at left]:
  x1cap_i   with   caps Slack
  fgret     with   x1cap           !

```

The last equation of Excerpt 31 may be used to enforce the rule that capital is fixed in aggregate, yet freely mobile between sectors. In this case, real returns to capital, $gret(i)$, move together (with caps Slack).

4.27. The labour market

Like ORANI-G, PHILGEM contains no theory of labour supply. Users of the model have the option of setting employment exogenously, with market-clearing wage rates determined endogenously, or setting wage rates (real or nominal) exogenously, allowing employment to be demand determined.

In Excerpt 32, equation E_{x1lab_i} computes changes in the aggregate demand for labour of each skill. Equation E_{p1lab} is a wage setting equation. In the standard ORANI short-run closure, the 'f1lab' variables are exogenous, so that all wages are indexed to the CPI. Then, if shocked, the 'f1lab' variables allow for deviations in the growth of some or all wages relative to the growth of the CPI. This short-run labour market closure was originally devised to reflect the centralised wage-fixing mechanisms that used to operate in Australia. However, it has proven popular in quite different contexts. It assumes that labour is mobile between industries, and that supply of each skill type is elastic.

As a variant on the standard short-run closure, the average nominal wage, $p1lab_io$, may be fixed, while endogenizing the overall wage shifter, $f1lab_io$. The effect will be to fix all the $p1lab$, giving the model a Keynesian flavour.

The typical long-run closure sets $employ_i$ (aggregate employment: wage bill weights) exogenous, while endogenizing the overall wage shifter, $f1lab_io$. The other 'f1lab' variables remain exogenous, reflecting fixed wage relativities. This assumes that labour is mobile between industries *and* occupations.

These labour-market modelling decisions are usually made at an economy-wide level, but could be applied individually and perhaps differentially to different industries or types of labour. For example, we could exogenize the supply of skilled workers and the wages of unskilled workers. Or, we could exogenize employment in agriculture and wages in other sectors²¹.

In the standard long-run and short-run closures, wage relativities are fixed, so firms do not substitute between labour of different types. Thus the values of $SIGMA1LAB$, the CES substitution elasticity between skill types, do not affect simulation results. This is handy for the modeller who is agnostic about those values (econometric evidence is scant). The more exotic labour market closures, which do allow wage relativities to change, focus attention on the values assumed for $SIGMA1LAB$.

```

! Excerpt 32 of TABLO input file: !
! Labour market !
Variable
  (all,i,IND)(all,o,OCC) f1lab(i,o) # Wage shift variable #;
                        (all,o,OCC) f1lab_i(o) # Occupation-specific wage shifter #;
                        (all,o,OCC) x1lab_i(o) # Employment by occupation #;
  (all,i,IND)           f1lab_o(i) # Industry-specific wage shifter #;
                        f1lab_io   # Overall wage shifter #;

Coefficient (all,o,OCC) V1LAB_I(o) # Total wages, occupation o #;
Formula    (all,o,OCC) V1LAB_I(o) = sum{i,IND, V1LAB(i,o)};
Equation
  E_x1lab_i # Demand equals supply for labour of each skill #
  (all,o,OCC) V1LAB_I(o)*x1lab_i(o) = sum{i,IND, V1LAB(i,o)*x1lab(i,o)};

  E_p1lab   # Flexible setting of money wages #
  (all,i,IND)(all,o,OCC)
  p1lab(i,o)= p3tot + f1lab_io + f1lab_o(i) + f1lab_i(o) + f1lab(i,o);

```

²¹ The TABLO input file contains optional code, not shown here, to implement upward-sloping labour supply schedules.

```

Variable (all,o,OCC) p1lab_i(o) # Average wage of occupation #;
Equation E_p1lab_i # Average wage of occupation #
  (all,o,OCC) V1LAB_I(o)*p1lab_i(o) = sum{i,IND, V1LAB(i,o)*p1lab(i,o)};

![[! ! Optional addition: Labour supply function !
! to activate equation below, exogenize flabsup(o) and endogenize f1lab_i(o) !
Variable (all,o,OCC) flabsup(o) # Labour supply shift #;
Equation E_flabsup # Labour supply functions #
  (all,o,OCC) x1lab_i(o) = 1.0*[p1lab_i(o) - p3tot] + flabsup(o);
!]]!

```

4.28. Miscellaneous equations

Equation E_p1oict in Excerpt 33 allows for indexation of the unit price of 'other costs' to the CPI. The variable f1oict(i) can be interpreted as the percentage change in the real price of 'other costs' to industry i.

Equation E_f3tot implements a simple, optional, consumption function. If f3tot is exogenous, the equation links household spending to nominal GDP. Otherwise, there is no effect on the rest of the system, since the variable f3tot appears nowhere else.

The next two equations define two vector variables, p0dom and p0imp, which are simply equal to the corresponding columns of the basic price matrix, p0. They are included to facilitate the viewing of simulation results: in that situation, vectors are often more convenient to work with than are matrices.

```

! Excerpt 33 of TABLO input file: !
! Miscellaneous equations !
Variable
  (all,i,IND) f1oict(i) # Shift in price of "other cost" tickets #;
  f3tot # Ratio, consumption/ GDP #;
Equation
  E_p1oict # Indexing of prices of "other cost" tickets # (all,i,IND)
  p1oict(i) = p3tot + f1oict(i); ! assumes full indexation !

  E_f3tot # Consumption function #
  w3tot = w0gdpexp + f3tot;

! Map between vector and matrix forms of basic price variables !
Variable
  (all,c,COM) p0imp(c) # Basic price of imported goods = p0(c,"imp") #;

Equation
  E_p0dom # Basic price of domestic goods = p0(c,"dom") # (all,c,COM)
  p0dom(c) = p0(c,"dom");

  E_p0imp # Basic price of imported goods = p0(c,"imp") # (all,c,COM)
  p0imp(c) = p0(c,"imp");

```

4.29. Adding variables for explaining results

The equations described thus far define the main part of the original ORANI-G. The remaining excerpts of the TABLO input file contain supplementary material which does not directly affect simulation results.

Part of the ORANI tradition is that simulation results, although voluminous, must all be capable of verbal explanation based on model equations and data. It is customary to examine and present results in great detail. The aim is to dispel any tendency to treat the model as a black box. These detailed analyses sometimes yield theoretical insights; for example, we may find that some mechanism which we thought to be of minor significance exerts a dominant force in certain sectors. More often we discover errors—either in the data or in the model equations. Inappropriate theory may also lead to implausible results.

Results analysis, then, is an indispensable (but laborious) part of quality control for an economic model. To make it less painful, we often add equations and variables merely to help explain results. The next two excerpts are examples of this type of addition.

4.30. Sales decomposition

The sales decomposition, implemented in Excerpt 34, breaks down the percentage change in the total sales of each commodity into various intermediate and final demand categories. This would be useful, for example, if we wondered whether an increase in motor vehicle output was due to increases in either investment or household demand.

To decompose a percentage change in this way (i.e., find parts which add up to the whole), we have to perform some arithmetical manipulations. Suppose we have a variable X which is the sum of 2 parts:

$$X = A + B \quad \text{or} \quad PX = PA + PB \quad (\text{where } P \text{ is a common price}) \quad (35)$$

then, for small percentage changes, we can write:

$$x = \text{conta} + \text{contb} \quad \text{where } \text{conta} = (PA/PX)a \quad \text{and} \quad \text{contb} = (PB/PX)b \quad (36)$$

We call conta and contb the *contributions* of A and B to the percentage change in X .

For larger changes, which require a multistep computation, equation (36) would result in values for conta and contb which did not quite add up to the total percentage change in X ²². To avoid this, it is useful to specify both conta and contb as ordinary change variables and to define a new ordinary change variable, q , in such a way that the final result for q (after results for several computational steps have been accumulated) is identical to that for x . This leads to the small change equation:

$$X^0 q = Xx \quad \text{where } X^0 \text{ is the initial value of } X, \quad (37)$$

and to the revised decomposition:

$$q = \text{conta} + \text{contb} \quad (38)$$

$$\text{where } \text{conta} = (PA/PX^0)a \quad \text{and} \quad \text{contb} = (PB/PX^0)b \quad (39)$$

In Excerpt 34, `INITSALES` is the *initial quantity* of `SALES` measured in current prices, analogous to PX^0 above. The last, Total, column of the `SalesDecomp` variable is the sum of the preceding 7 and should be identical to `x0com(c)`, the percentage change in commodity output. Each of the first 7 columns shows how some demand category contributed to `x0com`. The `delSale` variable, calculated earlier, contains the ordinary changes in quantities, measured in current prices.

```
! Excerpt 34 of TABLO input file: !
! Decomposition of sales change by destination !
Coefficient
(all,c,COM) INITSALES(c) # Initial volume of SALES at current prices #;

Formula (initial) (all,c,COM) INITSALES(c) = SALES(c);

Update (all,c,COM) INITSALES(c) = p0com(c);

Set DESTPLUS # Sale Categories #
(Interm, Invest, HouseH, Export, GovGE, Stocks, Margins, Total);
Subset DEST is subset of DESTPLUS;

Variable (change)
(all,c,COM)(all,d,DESTPLUS) SalesDecomp(c,d) # Sales decomposition #;

Equation
E_SalesDecompA # Initial volume of SALES # (all,c,COM)(all,d,DEST)
INITSALES(c)*SalesDecomp(c,d) = 100*delSale(c,"dom",d);

E_SalesDecompB # Total SALES # (all,c,COM)
SalesDecomp(c,"Total")= sum{d,DEST, SalesDecomp(c,d)};
```

4.31. The Fan decomposition

Suppose our simulation predicts an increase in domestic production of Textiles. This could be due to three causes:

²² The reason is that during a multistep computation percentage changes are compounded, whilst ordinary changes are added.

- the local market effect: an increase in local usage of Textiles, whether domestically-produced or imported;
- the export effect: an increase in exports of Textiles; or
- the domestic share effect: a shift in local usage of Textiles, from imported to domestically-produced.

Very often these 3 effects will work in different directions; for example, a increase in foreign demand might pull local producers up the supply curve, so increasing the domestic price and facilitating import penetration. The decomposition of Fan²³ aims to show the relative magnitude of these 3 contributions to output change.

Excerpt 35 starts by defining $x0loc$, the percentage change in local sales from both sources. Equation E_fandecompA says that this percentage, weighted by the value of local domestic sales, is the local market component of the percentage change in domestic production. Similarly, equation E_fandecompB defines the export component. In these equations INITSALES corresponds to the term PX^0 in equation (39): it is the initial value of sales, updated only by the change in price. Equation E_fandecompC defines the domestic share component as a residual²⁴. Finally, equation E_fandecompD corresponds to equation (37).

```
! Excerpt 35 of TABLO input file: !
! Decomposition of Fan !
Set FANCAT # Parts of Fan decomposition #
  (LocalMarket, DomShare, Export, Total);
Variable
  (all,c,COM) x0loc(c) # Real percent change in LOCSALES (dom+imp) #;
(change)(all,c,COM)(all,f,FANCAT) fandecomp(c,f) # Fan decomposition #;

Coefficient (all,c,COM) LOCSALES(c) # Total local sales of dom + imp good c #;
Formula    (all,c,COM) LOCSALES(c) = DOMSALES(c) + V0IMP(c);

Equation
E_x0loc # %Growth in local market # (all,c,COM)
  LOCSALES(c)*x0loc(c) = DOMSALES(c)*x0dom(c) + V0IMP(c)*x0imp(c);

E_fandecompA # Growth in local market effect # (all,c,COM)
  INITSALES(c)*fandecomp(c,"LocalMarket") = DOMSALES(c)*x0loc(c);
! The local market effect is the % change in output that would have occurred
if local sales of the domestic product had followed dom+imp sales (x0loc) !

E_fandecompB # Export effect # (all,c,COM)
  INITSALES(c)*fandecomp(c,"Export") = V4BAS(c)*x4(c);

E_fandecompC # Import leakage effect - via residual # (all,c,COM)
  fandecomp(c,"Total") = fandecomp(c,"LocalMarket") + fandecomp(c,"DomShare")
    + fandecomp(c,"Export");

E_fandecompD # Fan total = x0com # (all,c,COM)
  INITSALES(c)*fandecomp(c,"Total") = SALES(c)*x0com(c);
```

4.32. The expenditure side GDP decomposition

Excerpt 36 breaks down changes in real GDP into the contributions of the main expenditure-side aggregates (contGDPexp). This enables us to quickly see how much of the change in real expenditure-

²³ Named after Fan Ming-Tai of the Academy of Social Sciences, Beijing; their PRCGEM is one of the most elaborate versions of ORANI-G.

²⁴ No interactive term is concealed in the residual. Because these decompositions are specified in small change terms, the changes due to each part add up to the change in the whole. To convince yourself, retrace the example starting at equation (35) with the multiplicative form $X = AB$, leading to $X^0q = Xa + Xb$, with contribution terms $(X/X^0)a$ and $(X/X^0)b$. However, the cumulative results of these contributions can be defined only as a path integral of the contribution terms computed at each solution step. Hence they are not (quite) invariant to the details of our solution procedure. See also footnote 17.

side GDP is due, say, to a change in aggregate investment. The mathematics is the same as that of the Sales decomposition in Excerpt 34. For convenience, the contributions of exports and imports are combined into the variable contBOT. Note that contBOT, like each component of contGDPexp, is a *real* contribution: it takes no account of changes in export or import prices. It is the same (apart from units of measurement) as the ΔB variable in the original ORANI model.

```

! Excerpt 36 of TABLO input file: !
! GDP decomposition !
Set EXPMAC # Expenditure Aggregates #
  (Consumption, Investment, Government, Stocks, Exports, Imports);

Variable
  (change)(all,e,EXPMAC) contGDPexp(e)
                                # Contributions to real expenditure-side GDP #;
  (change) contBOT # Contribution of BOT to real expenditure-side GDP #;

Coefficient
  INITGDP # Initial real GDP at current prices #;

Formula (initial) INITGDP = V0GDPEXP;

Update INITGDP = p0gdpepx;

Equation
  E_contGDPexpA # Contributions to real expenditure-side GDP: Consumption #
    INITGDP*contGDPexp("Consumption") = V3TOT*x3tot;

  E_contGDPexpB # Contributions to real expenditure-side GDP: Investment #
    INITGDP*contGDPexp("Investment") = V2TOT_I*x2tot_i;

  E_contGDPexpC # Contributions to real expenditure-side GDP: Government #
    INITGDP*contGDPexp("Government") = V5TOT*x5tot;

  E_contGDPexpD # Contributions to real expenditure-side GDP: Stocks #
    INITGDP*contGDPexp("Stocks") = V6TOT*x6tot;

  E_contGDPexpE # Contributions to real expenditure-side GDP: Exports #
    INITGDP*contGDPexp("Exports") = V4TOT*x4tot;

  E_contGDPexpF # Contributions to real expenditure-side GDP: Imports #
    INITGDP*contGDPexp("Imports") = -V0CIF_C*x0cif_c;

  E_contBOT # Contribution of BOT to real expenditure-side GDP #
    contBOT = contGDPexp("Exports") + contGDPexp("Imports");

! Exact GDP decomposition from income side !
Set CONTINC (Land, Labour, Capital, IndTax, TechChange);
Variable
  (change) x0gdpinc # Real GDP from the income side #;
  (change) (all,c,CONTINC) contGDPinc(c) # Income side real GDP decomposition #;

Equation
  E_contGDPincA # Real GDP from the income side: Land #
    contGDPinc("Land") = [V1LND_I/INITGDP]*x1lnd_i;

  E_contGDPincB # Real GDP from the income side: Labour #
    contGDPinc("Labour") = [V1LAB_IO/INITGDP]*employ_i;

  E_contGDPincC # Real GDP from the income side: Capital #
    contGDPinc("Capital") = [V1CAP_I/INITGDP]*x1cap_i;

  E_contGDPincD # Real GDP from the income side: Indirect tax #
    contGDPinc("IndTax") = sum{i,IND, [V1OCT(i)/INITGDP]*x1oct(i)}

```

```

+ sum{i,IND, [V1PTX(i)/INITGDP]*x1tot(i)}
+ sum{c,COM, [V0TAR(c)/INITGDP]*x0imp(c)}
+ sum{c,COM,sum{s,SRC, sum{i,IND, [V1TAX(c,s,i)/INITGDP]*x1(c,s,i)}}}
+ sum{c,COM,sum{s,SRC, sum{i,IND, [V2TAX(c,s,i)/INITGDP]*x2(c,s,i)}}}
+ sum{c,COM,sum{s,SRC, [V3TAX(c,s)/INITGDP]*x3(c,s)}}
+ sum{c,COM, [V4TAX(c)/INITGDP]*x4(c)}
+ sum{c,COM,sum{s,SRC, [V5TAX(c,s)/INITGDP]*x5(c,s)}};

```

```

E_contGDPincE # Real GDP from the income side: Technical change #
contGDPinc("TechChange") =
-sum{c,COM,sum{s,SRC,sum{i,IND,[V1PUR(c,s,i)/INITGDP]*[a1(c,s,i)+a1_s(c,i)]}}
-sum{c,COM,sum{s,SRC,sum{i,IND,[V2PUR(c,s,i)/INITGDP]*[a2(c,s,i)+a2_s(c,i)]}}
-sum{i,IND, [V1LAB_O(i)/INITGDP]*a1lab_o(i)}
-sum{i,IND, [V1CAP(i)/INITGDP]*a1cap(i)}
-sum{i,IND, [V1LND(i)/INITGDP]*a1lnd(i)}
-sum{i,IND, [V1OCT(i)/INITGDP]*a1oct(i)}
-sum{i,IND, [V1PRIM(i)/INITGDP]*a1prim(i)}
-sum{c,COM,sum{s,SRC,sum{i,IND,sum{m,MAR,
[V1MAR(c,s,i,m)/INITGDP]*a1mar(c,s,i,m)}}}
-sum{c,COM,sum{s,SRC,sum{i,IND,sum{m,MAR,
[V2MAR(c,s,i,m)/INITGDP]*a2mar(c,s,i,m)}}}
-sum{c,COM,sum{s,SRC,sum{m,MAR,[V3MAR(c,s,m)/INITGDP]*a3mar(c,s,m)}}}
-sum{c,COM,sum{m,MAR,[V4MAR(c,m)/INITGDP]*a4mar(c,m)}}
-sum{c,COM,sum{s,SRC,sum{m,MAR,[V5MAR(c,s,m)/INITGDP]*a5mar(c,s,m)}}}
-sum{i,IND, [V2TOT(i)/INITGDP]*a2tot(i)}
-sum{i,IND, [V1CST(i)/INITGDP]*a1tot(i)};
E_x0gdpinc # Decomposition of real GDP from income side #
x0gdpinc = sum{c,CONTINC, contGDPinc(c)};

```

```

! Variables for back-of-the-envelope explanations of results !
! Contributions to real GDP at factor cost: x1prim_i !
! = exact form of approximate equation: x1prim_i = SL.l + SK.k + TechChange !
Coefficient INITVGDPfac # Initial factor cost GDP at current prices #;

```

```

Formula (initial) INITVGDPfac = V1PRIM_I;

```

```

Update INITVGDPfac = p1prim_i;

```

```

Set

```

```

CONTFAC # Primary factors, tech change #(Land, Labour, Capital, TechChange);

```

```

Variable

```

```

(change) x0gdpfac # Real GDP at factor cost #;
(change)(all,c,CONTFAC) contGDPfac(c)
# Contributions to real GDP at factor cost:x0gdpfac #;

```

```

Equation

```

```

E_contGDPfacA contGDPfac("Land") = [V1LND_I /INITVGDPfac]*x1lnd_i;
E_contGDPfacB contGDPfac("Labour") = [V1LAB_IO/INITVGDPfac]*employ_i;
E_contGDPfacC contGDPfac("Capital") = [V1CAP_I /INITVGDPfac]*x1cap_i;
E_contGDPfacD contGDPfac("TechChange") = [INITGDP/INITVGDPfac]*
contGDPinc("TechChange");

```

```

E_x0gdpfac # Decomposition of real GDP at factor cost #
x0gdpfac = sum{c,CONTFAC, contGDPfac(c)};

```

```

! Effective price of labour -- including labour-saving tech change !
Variable pLabEff # Effective price of labour, inc. labour-saving tech change #;
Equation E_pLabEff

```

```

V1LAB_IO*pLabEff = sum{i,IND,V1LAB_O(i)*[p1lab_o(i) + a1lab_o(i)]};

```

```

! Useful price ratios !

```

```

Variable pLabEff_p1prim # PLabEff / p1prim_i #;
Equation E_pLabEff_p1prim pLabEff_p1prim = pLabEff - p1prim_i;

```



```

Variable pLabEff_p3tot # PLabEff / p3tot #; ! maybe constant in short run !
Equation E_pLabEff_p3tot pLabEff_p3tot = pLabEff - p3tot;

Variable p3tot_p0GNE # P3tot / p0GNE #; ! might not change much !
Equation E_p3tot_p0GNE p3tot_p0GNE = p3tot - p0gne;

Variable p0GNE_p0GDPExp # P0GNE / p0GDPExp #; ! = f(terms of trade) !
Equation E_p0GNE_p0GDPExp p0GNE_p0GDPExp = p0gne - p0gdpepx;

Variable p0GDPExp_p1prim # P0GDPExp / p1prim_i #; ! = f(indirect taxes) !
Equation E_p0GDPExp_p1prim p0GDPExp_p1prim = p0gdpepx - p1prim_i;

Variable pCap_p1prim # PCap / p1prim_i #;
Equation E_pCap_p1prim pCap_p1prim = p1cap_i - p1prim_i;

Variable pCap_p2tot # PCap_i / p2tot #; ! maybe constant in long run !
Equation E_pCap_p2tot pCap_p2tot = p1cap_i - p2tot_i;

Variable p2tot_p0GNE # P2tot_i / p0GNE #; ! might not change much !
Equation E_p2tot_p0GNE p2tot_p0GNE = p2tot_i - p0gne;

```

The second part of Excerpt 36 breaks down changes in real GDP from the income side: into the contributions due to primary factor usage, indirect taxes, and technological change. If primary factor endowments and technology are fixed, the indirect tax terms may be identified with changes in allocative efficiency.

4.33. Checking the data

A model rendered in the TABLO language is a type of computer program, and like other computer programs tends to contain errors. We employ a number of strategies to prevent errors and to make errors apparent. One strategy is to check all conditions which the initial data must satisfy. This is done in Excerpt 37. The conditions are:

- The row sums of the MAKE matrix must equal the row sums of the BAS and MAR rows of Figure 4. That is, the output of domestically produced commodities must equal the total of the demands for them.
- The column sums of the MAKE matrix must equal the sum of the first, producers', column of Figure 4. That is, the value of output by each industry must equal the total of production costs.
- The satellite matrix COM*HOU, which must add up (over HOU) to the "core" data matrix V3PUR (COM*SRC).
- The average value of the household expenditure elasticities, EPS, should be one. The average should be computed using the expenditure weights, V3PUR_S.

To check these conditions, the items PURE_PROFITS, LOST_GOODS, V3PURH_CHECK and EPSTOT are stored on the SUMMARY file. Their values should be near to zero (or one, for EPSTOT).

The Assertion statements which come next enforce the above rules. If the specified condition is not met during some simulation, GEMPACK will stop with an error message indicating which condition is not satisfied. The first three Assertions are checked before every stage of a multi-step simulation. The last one, for EPSTOT, should be true before and after (but not necessarily *during*) a simulation.

It should be emphasized that the validity of percentage change equations depends on the validity of the data from which the equation coefficients are calculated. The GEMPACK solution method must start from a database which is consistent, in the levels, with all the equations.

There are other formal tests which can reveal errors in model formulation. These are set out in Appendix I.

```

! Excerpt 37 of TABLO input file: !
! Check identities !
Coefficient                ! coefficients for checking !
(all,i,IND) DIFFIND(i)    # COSTS-MAKE_C : should be zero #;
(all,c,COM) DIFFCOM(c)    # SALES-MAKE_I : should be zero #;

```

Formula

```
(all,i,IND) DIFFIND(i) = V1TOT(i) - MAKE_C(i);
(all,c,COM) DIFFCOM(c) = SALES(c) - MAKE_I(c);
```

Write ! we file these numbers BEFORE the assertions below !

```
DIFFIND to file SUMMARY header "DIND";
DIFFCOM to file SUMMARY header "DCOM";
EPSTOTH to file SUMMARY header "ETOT";
```

Assertion ! if below not true, program will stop with message !

```
# DIFFIND = V1TOT-MAKE_C = tiny # (all,i,IND) ABS[DIFFIND(i)/V1TOT(i)] <0.001;
# DIFFCOM = SALES-MAKE_I = tiny # (all,c,COM) ABS[DIFFCOM(c)/SALES(c)] <0.001;
(initial) # Average Engel elasticity = 1 # (all,h,HOU) ABS[1-EPSTOTH(h)]<0.1;
```

4.34. Summarizing the data

The next few Excerpts collect together various summaries of the data and store these on file in a form that is convenient for later viewing. These summaries are useful for checking the plausibility of data and for explaining simulation results. Excerpt 38 groups into vectors the various components of, first, GDP from the expenditure side; second, GDP from the income side; and third, the components of total indirect taxes²⁵.

```
! Excerpt 38 of TABLO input file: !
! Summary: components of GDP from income and expenditure sides !
Coefficient (all,e,EXPMAC) EXPGDP(e) # Expenditure Aggregates #;
```

Formula

```
EXPGDP("Consumption") = V3TOT;
EXPGDP("Investment") = V2TOT_I;
EXPGDP("Government") = V5TOT;
EXPGDP("Stocks") = V6TOT;
EXPGDP("Exports") = V4TOT;
EXPGDP("Imports") = -V0CIF_C;
```

Write EXPGDP to file SUMMARY header "EMAC";

Set INCMAC # Income Aggregates #

```
(Land, Labour, Capital, IndirectTax);
```

Coefficient (all,i,INCMAC) INCGDP(i) # Income Aggregates #;

Formula

```
INCGDP("Land") = V1LND_I;
INCGDP("Labour") = V1LAB_IO;
INCGDP("Capital") = V1CAP_I;
INCGDP("IndirectTax") = V0TAX_CSI;
```

Write INCGDP to file SUMMARY header "IMAC";

Set TAXMAC # Tax Aggregates #

```
(Intermediate, Investment, Consumption, Exports, Government, OCT, ProdTax, Tariff);
```

Coefficient (all,t,TAXMAC) TAX(t) # Tax Aggregates #;

Formula

```
TAX("Intermediate") = V1TAX_CSI;
TAX("Investment") = V2TAX_CSI;
TAX("Consumption") = V3TAX_CS;
TAX("Exports") = V4TAX_C;
TAX("Government") = V5TAX_CS;
TAX("OCT") = V1OCT_I;
TAX("ProdTax") = V1PTX_I;
TAX("Tariff") = V0TAR_C;
```

Write TAX to file SUMMARY header "TMAC";

²⁵ GEMPACK stores data in its own, binary, format. A Windows program, ViewHAR, is normally used for viewing or modifying these so-called HAR files. The data matrices created here are designed to be convenient for examining with ViewHAR. ViewHAR automatically calculates and displays subtotals, so that total GDP, for example, does not need to be included in the summary vectors defined here.

```

Coefficient (parameter)(all,t,TAXMAC) TAX0(t) # Initial Tax Aggregates #;
Formula    (initial) (all,t,TAXMAC) TAX0(t) = TAX(t); ! used in postsim !

```

Excerpt 39 forms a matrix showing the main parts of production cost for each industry.

```

! Excerpt 39 of TABLO input file: !
! Summary: matrix of industry costs !
Set COSTCAT # Cost Categories #
(IntDom, IntImp, Margin, ComTax, Lab, Cap, Lnd, ProdTax, OCT);
Coefficient (all,i,IND)(all,co,COSTCAT) COSTMAT(i,co) # Cost Matrix #;
Formula
(all,i,IND) COSTMAT(i,"IntDom") = sum{c,COM, V1BAS(c,"dom",i)};
(all,i,IND) COSTMAT(i,"IntImp") = sum{c,COM, V1BAS(c,"imp",i)};
(all,i,IND) COSTMAT(i,"Margin") =
    sum{c,COM, sum{s,SRC, sum{m,MAR, V1MAR(c,s,i,m)}}};
(all,i,IND) COSTMAT(i,"ComTax") = sum{c,COM, sum{s,SRC, V1TAX(c,s,i)}};
(all,i,IND) COSTMAT(i,"Lab") = V1LAB_0(i);
(all,i,IND) COSTMAT(i,"Cap") = V1CAP(i);
(all,i,IND) COSTMAT(i,"Lnd") = V1LND(i);
(all,i,IND) COSTMAT(i,"ProdTax") = V1PTX(i);
(all,i,IND) COSTMAT(i,"OCT") = V1OCT(i);
Write COSTMAT to file SUMMARY header "CSTM";

```

Excerpt 40 calculates for each main destination, the basic, margin, and commodity tax components of purchases by commodity and source.

```

! Excerpt 40 of TABLO input file: !
! Summary: basic, margins and taxes !
Set
SALECAT2 # SALE Categories # (Interm, Invest, HouseH, Export, GovGE, Stocks);
FLOWTYPE # Type of flow # (Basic, Margin, TAX);
Coefficient
(all,c,COM)(all,f,FLOWTYPE)(all,s,SRC)(all,sa,SALECAT2) SALEMAT2(c,f,s,sa)
    # Basic, margin and tax components of purchasers' values #;
Formula
(all,c,COM)(all,f,FLOWTYPE)(all,s,SRC)(all,sa,SALECAT2) SALEMAT2(c,f,s,sa)=0;
(all,c,COM)(all,s,SRC) SALEMAT2(c,"Basic",s,"Interm") = sum{i,IND,V1BAS(c,s,i)};
(all,c,COM)(all,s,SRC) SALEMAT2(c,"Tax",s,"Interm") = sum{i,IND,V1TAX(c,s,i)};
(all,c,COM)(all,s,SRC) SALEMAT2(c,"Margin",s,"Interm") =
    sum{i,IND, sum{m,MAR, V1MAR(c,s,i,m)}};
(all,c,COM)(all,s,SRC) SALEMAT2(c,"Basic",s,"Invest") = sum{i,IND,V2BAS(c,s,i)};
(all,c,COM)(all,s,SRC) SALEMAT2(c,"Tax",s,"Invest") = sum{i,IND,V2TAX(c,s,i)};
(all,c,COM)(all,s,SRC) SALEMAT2(c,"Margin",s,"Invest") =
    sum{i,IND, sum{m,MAR, V2MAR(c,s,i,m)}};
(all,c,COM)(all,s,SRC) SALEMAT2(c,"Basic",s,"HouseH") = V3BAS(c,s);
(all,c,COM)(all,s,SRC) SALEMAT2(c,"Tax",s,"HouseH") = V3TAX(c,s);
(all,c,COM)(all,s,SRC) SALEMAT2(c,"Margin",s,"HouseH") = sum{m,MAR,V3MAR(c,s,m)};
(all,c,COM)(all,s,SRC) SALEMAT2(c,"Basic",s,"GovGE") = V5BAS(c,s);
(all,c,COM)(all,s,SRC) SALEMAT2(c,"Tax",s,"GovGE") = V5TAX(c,s);
(all,c,COM)(all,s,SRC) SALEMAT2(c,"Margin",s,"GovGE") = sum{m,MAR,V5MAR(c,s,m)};
(all,c,COM) SALEMAT2(c,"Basic","dom","Export") = V4BAS(c);
(all,c,COM) SALEMAT2(c,"Tax","dom","Export") = V4TAX(c);
(all,c,COM) SALEMAT2(c,"Margin","dom","Export") = sum{m,MAR,V4MAR(c,m)};
(all,c,COM)(all,s,SRC) SALEMAT2(c,"Basic",s,"Stocks") = V6BAS(c,s);
Write SALEMAT2 to file SUMMARY header "MKUP";

```

4.35. Import shares and short-run supply elasticities

Excerpt 41 computes and stores 2 more items which help explain results. The higher is the import share IMPSHR for some commodity, the more vulnerable is the associated local industry to import competition.

Short-run simulation results are often coloured by the effects of inelastic supply schedules for capital-intensive industries. We compute and store the short-run supply elasticity SUPPLYELAST to help identify such industries. The formula used is derived in Appendix J. The closely-related primary factor shares can be seen from the FACTOR matrix (using the *row shares* feature of the GEMPACK program ViewHAR).

```

! Excerpt 41 of TABLO input file: !
! Import shares and short-run supply elasticities !
Coefficient (all,c,COM) IMPSHR(c) # Share of imports in local market #;
Formula (all,c,COM) IMPSHR(c) = V0IMP(c)/[TINY+DOMSALES(c)+V0IMP(c)];
Write IMPSHR to file SUMMARY header "MSHR";

Coefficient (all,i,IND) SUPPLYELAST(i) # Short-run supply elasticity #;
Zerodivide default 999;
Zerodivide (nonzero_by_zero) default 999;
Formula (all,i,IND) SUPPLYELAST(i) =
    SIGMA1PRIM(i)*V1LAB_0(i)*V1TOT(i)/[V1PRIM(i)*{V1CAP(i)+V1LND(i)}];
Zerodivide off;
Zerodivide (nonzero_by_zero) off;
Write SUPPLYELAST to file SUMMARY header "SRSE";

Set FAC # Primary Factors # (Lab, Cap, Lnd);
Coefficient (all,i,IND)(all,f,FAC) FACTOR(i,f) # Primary Factor Costs #;
Formula
    (all,i,IND) FACTOR(i,"Lab") = V1LAB_0(i);
    (all,i,IND) FACTOR(i,"Cap") = V1CAP(i);
    (all,i,IND) FACTOR(i,"Lnd") = V1LND(i);
Write FACTOR to file SUMMARY header "FACT";

PostSim (begin); ! You need GEMPACK 9 or later for POSTSIM sections !
! report FanDecomp in order of x0com, losers first !
Set COMUP = COM ranked up by x0com;
Coefficient
    (all,c,COMUP)(all,f,FANCAT) fanrank(c,f) # Ranked Fan decomposition #;
Formula (all,c,COMUP)(all,f,FANCAT) fanrank(c,f) = fandecomp(c,f);
Write fanrank to file SUMMARY header "FRNK";
Set TRD (Import,Export);
Coefficient
    (all,c,COMUP)(all,t,TRD) TRADEPRESSRE(c,t) # Trade pressure, ranked by x0com #;
Formula ! TRADEPRESSRE is an approximation of demand elasticity !
    (all,c,COM) TRADEPRESSRE(c,"Import") = IMPSHR(c)*SIGMA1(c);
    (all,c,TRADEXP) TRADEPRESSRE(c,"Export") = EXPSHR(c)*ABS[EXP_ELAST(c)];
    (all,c,NTRADEXP) TRADEPRESSRE(c,"Export") = EXPSHR(c)*ABS[EXP_ELAST_NT];
Write TRADEPRESSRE to file SUMMARY header "TPRS";
! make tax revenue report !
Set CHNG (initial,Final,OrdChange,PercChange);
Coefficient
    (all,t,TAXMAC)(all,c,CHNG) TAXVAL(t,c) # Changes in nominal tax aggregates #;
Formula
    (all,t,TAXMAC) TAXVAL(t,"Initial") = TAX0(t);
    (all,t,TAXMAC) TAXVAL(t,"Final") = TAX(t);
    (all,t,TAXMAC) TAXVAL(t,"OrdChange") = TAX(t)-TAX0(t);
    (all,t,TAXMAC) TAXVAL(t,"PercChange") = 100*[TAX(t)-TAX0(t)]/ID01[TAX0(t)];
Write TAXVAL to file SUMMARY header "TAXV";
PostSim (end);

```

4.36. Storing Data for Other Computations

It is often useful to extract data from the model for other calculations. For example, we might wish to combine levels data with change results in our presentation of simulation results. Another important use

of such data is in aggregating the model database. It is wise to prepare the initial model database at the highest level of disaggregation supported by available Input-Output tables. This large database can be aggregated later, if desired, to a smaller number of sectors. A specialized program, AggHAR, may ease the aggregation task.

For flows data, each item in the aggregated database is simply the sum of corresponding sectors in the original database. Parameters, however, can not normally be added together. Instead, aggregated parameters are normally weighted averages of the original parameters. The purpose of Excerpt 42 is store such weights on a file. For example, the parameter SIGMA2 (Armington elasticity between domestic and imported commodities used for investment) could be aggregated using the weight vector V2PUR_SI. Information about which weight to use for each elasticity is conveyed to AggHAR by the WAGGSET vector of strings. For example, the first element of WAGGSET, "SCET@@@1TOT" indicates that the MAKE CET elasticities stored at header SCET on the ORANIG input data file should be aggregated using the industry output values stored at header 1TOT on the summary file²⁶.

```

! Excerpt 42 of TABLO input file: !
! Weight vectors for use in aggregation and other calculations !
Coefficient (all,c,COM) V1PUR_SI(c) # Dom+imp intermediate purch. value #;
              (all,c,COM) V2PUR_SI(c) # Dom+imp investment purch. value #;
              (all,c,COM) V5PUR_S(c)  # Dom+imp government purch. value #;
              (all,c,COM) V6BAS_S(c)  # Dom+imp inventories #;
Formula      (all,c,COM) V1PUR_SI(c) = sum{i,IND, V1PUR_S(c,i)};
              (all,c,COM) V2PUR_SI(c) = sum{i,IND, V2PUR_S(c,i)};
              (all,c,COM) V5PUR_S(c)  = sum{s,SRC, V5PUR(c,s)};
              (all,c,COM) V6BAS_S(c)  = sum{s,SRC, V6BAS(c,s)};

Write                                               ! weight to aggregate..... !
V1TOT      to file SUMMARY header "1TOT";!MAKE CET elasticities at SCET !
V4PUR      to file SUMMARY header "4PUR";!export elasticities at header P018!
V1LAB_0    to file SUMMARY header "LAB1";!CES inter-skill elasticities at SLAB!
V1PRIM     to file SUMMARY header "VLAD";!CES primary-factor elasts at P028!
V1PUR_SI   to file SUMMARY header "1PUR";!Armington elasticities at 1ARM !
V2PUR_SI   to file SUMMARY header "2PUR";!Armington elasticities at 2ARM !
V3PUR_S    to file SUMMARY header "3PUR";!Armington[3ARM] & Engel[XPEL] !

Set WAGGSET # Instructions to AGGHAR for weighted aggregation of parameters # (
  SCET@@@1TOT,
  P018@@@4PUR,
  ITEX@@@4PUR, ! members of individual exporter group !
  XPEL@@@3PUR,
  P028@@@VLAD,
  SLAB@@@LAB1,
  xx1ARM@@1PUR, ! leading "xx" is ignored by AGGHAR !
  xx2ARM@@2PUR,
  xx3ARM@@3PUR,
  R001@@@RW01, ! next for regional shares, see below !
  R002@@@RW02,
  R003@@@RW03,
  R004@@@RW04,
  R005@@@RW05,
  R006@@@RW06);
Write (Set) WAGGSET to file SUMMARY header "WAGG";

! Other useful values !
Coefficient (all,c,COM) TARFRATE(c) # Ad valorem tariff rate #;
Formula    (all,c,COM) TARFRATE(c) = V0TAR(c)/[TINY+V0CIF(c)];
Write
  TARFRATE to file SUMMARY header "TRAT";
  V0TAR    to file SUMMARY header "0TAR";

```

²⁶ The set WAGGSET is used to work around a GEMPACK limitation: the only way to write lists of strings (such as headers) to a file is to write them as set elements. Furthermore, set elements may not begin with a digit.

```
V0CIF    to file SUMMARY header "0CIF";
V0IMP    to file SUMMARY header "0IMP";
MAKE     to file SUMMARY header "MAKE";
V1CAP    to file SUMMARY header "1CAP";
V6BAS_S  to file SUMMARY header "6BSS";
V2TOT    to file SUMMARY header "2TOT";
V5PUR_S  to file SUMMARY header "5PUR";
```

5. Equations for SAM extension

As suggested at the outset, PHILGEM offers a good starting point for model development, especially for researchers who want to extend their ORANI-G models to draw on supplementary data coming from a social accounting matrix (SAM). We now explain the additional variables and behavioural or identity equations needed to implement this extension. The additional material is merely added on at the bottom of the original ORANI-G model code.

The coefficient and variable names related to the SAM extension, apart from being consistent with naming conventions described in section 4.2, conform to SAM conventions. That is, flows related to SAM data are named based on the row and column in which they are situated, with rows representing receipts and columns outlays. For example, the flow VHOUGOS refers to households income from capital or gross operating surplus (GOS).²⁷ Similarly, VTAXHOU specifies direct income tax revenue paid by households.

The broad strategy is to explain the modelling of each new data item (shown shaded in Figure 5) proceeding from top left to bottom right of the Figure. We explain new items in each row to determine income, which then drives expenditure items in the corresponding column.

5.1. Gross Operating Surplus

Excerpt 43 distributes income derived from capital or gross operating surplus (GOS) to the household, enterprise and government agents. The first equation E_wgos determines the percentage change in total gross operating surplus (GOS) while the next equations specify that the percentage change in GOS income for each agent follows the percentage change in total GOS.

The capital income shares of these agents are sourced from the national income accounts of the Philippines for the year 2000 (NSCB 2006) . In turn, each household's share in total capital income earned by all households in the economy is taken from the 2003 household survey, known locally as the family income and expenditure survey (NSO 2003).

```
! Excerpt 43 of TABLO input file: !
! Distribution of Gross Operating Surplus (GOS)!
Variable ! GOS payments!
    wgos      # Total GOS #;
(all,h,HOU) whougos(h) # Household receipts/income from GOS #;
    whougos_h # Total Household receipts/income from GOS #;
    wentgos   # Enterprises' receipts/income from GOS #;
    wgovgos   # Government receipts/income from GOS #;

Coefficient
    VGOS      # Total GOS #;
(all,h,HOU) VHOUGOS(h) # Household receipts/income from GOS #;
    VHOUGOS_H # Total Household/income receipts from GOS #;
    VENTGOS   # Enterprises' receipts/income from GOS #;
    VGOVGOS   # Government receipts/income from GOS #;

Read
    VHOUGOS from file BASEDATA header "VHGS";
```

²⁷ The related PEKGEM model of South Africa followed the *opposite* ordering convention.

```

VENTGOS from file BASEDATA header "VEGS";
VGOVGOS from file BASEDATA header "VGGS";

Update
(all,h,HOU) VHOUGOS(h) = whougos(h);
                VENTGOS = wentgos;
                VGOVGOS = wgovgos;

Formula
    VGOS = V1CAP_I + V1LND_I;
    VHOUGOS_H = sum{h,HOU, VHOUGOS(h)};

Equation
E_wgos # GOS from income side #
    VGOS*wgos = V1CAP_I*w1cap_i + V1LND_I*w1lnd_i;

E_whougos # Household receipts/income from GOS # (all,h,HOU)
    whougos(h) = wgos;

E_whougos_h # Total Household/income receipts from GOS#
    VHOUGOS_H*whougos_h = sum{h,HOU, VHOUGOS(h)*whougos(h)};

E_wentgos # Enterprises' receipts/income from GOS #
    wentgos = wgos;

E_wgovgos # Government receipts/income from GOS #
    wgovgos = wgos;

Write
VGOS    to file SUMMARY header "VGOS";
VHOUGOS to file SUMMARY header "VHGS";
VENTGOS to file SUMMARY header "VEGS";
VGOVGOS to file SUMMARY header "VGGS";

```

5.2. Enterprises Account

The coefficients and variables associated with Excerpt 44 relate to the enterprises account which represents all public and private corporations in the economy. As shown in the first equation of Excerpt 44, enterprises derive their income from factors (gross operating surplus) and transfers from households and the rest of the world (ROW). The succeeding equations postulate that the percentage change in each household's property payments to enterprises is directly related to the percentage change in their income from GOS; and ROW transfers to enterprises are influenced by the movement of the economy's gross domestic product (GDP).

The final set of equations found in Excerpt 43 presents the outlays incurred by enterprises. After paying income taxes to the government, enterprises then transfer part of their income, in the form of dividends to households government and the ROW—with the percentage change in transfers received by these agents determined by the percentage change in the post-tax income of enterprises. Enterprise savings or retained earnings is computed as a residual. The last equation in Excerpt 43 determines the income tax rate imposed on enterprises, with the variable *ftaxent* acting as a shift variable to facilitate possible changes in tax structure. This will be explained in section 5.11.

```

! Excerpt 44 of TABLO input file: !
! Enterprises account: Receipts, expenditures and savings !
Variable !Income!
    went          # Total income of enterprises #;
(all,h,HOU) wenthou(h) # Enterprise receipts from household #;
    wenthou_h     # Total enterprise receipts from households #;
    wentrow       # Enterprise receipts from the rest of the world #;

Coefficient
    VENT          # Total income of enterprises #;

```

```
(all,h,HOU) VENTHOU(h) # Enterprise receipts from household #;
           VENTHOU_H # Total enterprise receipts from households #;
           VENTROW   # Enterprise receipts from the rest of the world #;
```

Read

```
VENTHOU from file BASEDATA header "VETH";
VENTROW from file BASEDATA header "VERW";
```

Update

```
(all,h,HOU) VENTHOU(h) = wenthou(h);
           VENTROW   = wentrow;
```

Formula

```
VENT = VENTGOS + sum{h,HOU, VENTHOU(h)} + VENTROW;
VENTHOU_H = sum{h,HOU, VENTHOU(h)};
```

Equation

```
E_went # Total income of enterprises #
VENT*went = VENTGOS*wentgos + sum{h,HOU, VENTHOU(h)*wenthou(h)}
           + VENTROW*wentrow;
```

```
E_wenthou # Enterprise receipts from household # (all,h,HOU)
wenthou(h) = whougos(h);      ! default assumption !
```

```
E_wenthou_h # Total enterprise receipts from households #
VENTHOU_H*wenthou_h = sum{h,HOU, VENTHOU(h)*wenthou(h)};
```

```
E_wentrow # Row transfers to enterprises #
wentrow = w0gdpxp;      ! default assumption !
```

Write

```
VENT      to file SUMMARY header "VENT";
VENTHOU   to file SUMMARY header "VETH";
VENTROW   to file SUMMARY header "VERW";
```

Variable !enterprise payments!

```
           went_posttax # Enterprise tax payments #;
(all,h,HOU) whouent(h) # Household receipts from enterprises #;
           wgovent     # Government transfers to enterprises #;
           wtaxent     # Enterprise direct income tax payments #;
           wrowent     # Row income from enterprises: repatriated profits #;
           wsavent     # Enterprise savings #;
           ftaxent     # Ad valorem rate of corporation tax #;
           whouent_h   # Total household income/dividends from enterprises #;
```

Coefficient

```
VENT_POSTTAX # VGOS less VGOSTAX #;
(all,h,HOU) VHOUEENT(h) # Household receipts from enterprises #;
           VGOVENT     # Government transfers to enterprises #;
           VTAXENT     # Enterprise direct income tax payments #;
           VROWENT     # Row income from enterprises: repatriated profits #;
           VSAVENT     # Enterprise savings #;
           VHOUEENT_H  # Total household income/dividends from enterprises #;
```

Read

```
VHOUEENT from file BASEDATA header "VHET";
VGOVENT  from file BASEDATA header "VGVE";
VTAXENT  from file BASEDATA header "VTXE";
VROWENT  from file BASEDATA header "VRWE";
```

Update

```
(all,h,HOU) VHOUEENT(h) = whouent(h);
           VGOVENT     = wgovent;
```



```
VTAXENT    = wtaxent;
VROWENT    = wrowent;
```

Formula

```
VSAVENT    = VENT - [sum{h,HOU, VHOUEENT(h)} + VGOVENT + VTAXENT + VROWENT];
VENT_POSTTAX = VENT - VTAXENT;
VHOUEENT_H  = sum{h,HOU, VHOUEENT(h)};
```

Equation

```
E_went_posttax # Enterprise post-tax income #
  VENT_POSTTAX*went_posttax = VENT*went - VTAXENT*wtaxent;
```

```
E_wgovent # Enterprise transfers to gov #
  wgovent = went_posttax;
```

```
E_wrowent # Enterprise transfers to ROW #
  wrowent = went_posttax;
```

```
E_whouent # Enterprise transfers to households # (all,h,HOU)
  whouent(h) = went_posttax;
```

```
E_whouent_h # Total enterprise transfers to households #
  VHOUEENT_H*whouent_h = sum{h,HOU, VHOUEENT(h)*whouent(h)};
```

```
E_wsavent # Find savings as residual #
  VSAVENT*wsavent = VENT*went - [!sum{h,HOU, VHOUEENT(h)*whouent(h)}!
    VHOUEENT_H*whouent_h + VGOVENT*wgovgos +
    VTAXENT*wtaxent + VROWENT*wrowent];
```

```
E_wtaxent # Corporation tax #
  wtaxent = ftaxent + went;
```

Write

```
VHOUEENT to file SUMMARY header "VHTE";
VGOVENT to file SUMMARY header "VGVE";
VTAXENT to file SUMMARY header "VTXE";
VROWENT to file SUMMARY header "VRWE";
VSAVENT to file SUMMARY header "VSVE";
VENT_POSTTAX to file SUMMARY header "VEPT";
```

5.3. Labour income of households

A new feature introduced in PHILGEM is the possibility of distinguishing each household's labour income earnings by industry and occupation. As shown in Excerpt 45, this is achieved by introducing a satellite matrix `V1LABINC(IND*OCC*HOU)`, which must add up (over HOU) to the "core" data matrix `V1LAB(IND*OCC)`. The Assertion statements imposed upon coefficient `LABCHECK` enforce the above rule.

The first equation in Excerpt 45 determines the percentage change in the value of wages paid by industry *i* to occupation *o*. The `Backsolve` command substitutes out `wllab`, so reducing the number of variables and equations, and speeding solution. The next equation specifies that each household's wage income follows population and the percentage change in the value of wages paid by industry *i* to occupation *o*. A constraint is then imposed, through a complementarity condition, to ensure that the value of industry wage bill by occupation type is equal to value of wage income received by households arising from selling occupation *o* to industry *i*. The next three equations compute various add-ups of the labour wage bill, for use in succeeding excerpts.

```
! Excerpt 45 of TABLO input file: !
```

```
! Labour income of households !
```

```
Variable
```

```
(all,i,IND)(all,o,OCC) wllab(i,o) # Labour income #;
(all,i,IND)(all,o,OCC)(all,h,HOU) wllabinc(i,o,h) # HH Labour income #;
```

```
(all,i,IND)(all,o,OCC) labslack(i,o) # Employment rate #;
      (all,i,IND) w1labh_oh(i) # Labour bills #;
(all,h,HOU)(all,o,OCC) w1labh_i(h,o) # Labour income of hh #;
      (all,h,HOU) w1labinc_io(h) # Total hh Labour income #;
```

Coefficient

```
(all,i,IND)(all,o,OCC)(all,h,HOU) V1LABINC(i,o,h) # HH Labour income #;
      (all,i,IND) V1LABH_OH(i) # Labour bills #;
      (all,h,HOU)(all,o,OCC) V1LABOH_I(h,o) # Labour income of hh#;
      (all,h,HOU) V1LABINC_IO(h) # Total hh Labour income #;
```

Read V1LABINC from file BASEDATA header "LINC";

Update

```
(all,i,IND)(all,o,OCC)(all,h,HOU) V1LABINC(i,o,h) = w1labinc(i,o,h);
```

Formula

```
(all,i,IND) V1LABH_OH(i) = sum{o,OCC, sum{h,HOU, V1LABINC(i,o,h)}};
(all,h,HOU)(all,o,OCC) V1LABOH_I(h,o)= sum{i,IND, V1LABINC(i,o,h)};
      (all,h,HOU) V1LABINC_IO(h)= sum{o,OCC, sum{i,IND, V1LABINC(i,o,h)}};
```

Coefficient (all,i,IND)(all,o,OCC) LABCHECK(i,o) # Check V1LAB = V1LABINC #;

```
Formula (all,i,IND)(all,o,OCC) LABCHECK(i,o) = V1LAB(i,o) -
      sum{h,HOU, V1LABINC(i,o,h)};
```

Assertion (all,i,IND)(all,o,OCC) ABS[LABCHECK(i,o)] < 0.1;

Equation

```
E_w1lab # Labour income # (all,i,IND)(all,o,OCC)
      w1lab(i,o) = p1lab(i,o) + x1lab(i,o);
```

Backsolve w1lab using E_w1lab;

Equation

```
E_w1labinc # HH Labour income # (all,i,IND)(all,o,OCC)(all,h,HOU)
      w1labinc(i,o,h) = qh(h) + labslack(i,o) + w1lab(i,o);
```

```
E_labslack # Constraint # (all,i,IND)(all,o,OCC)
      V1LAB(i,o)*w1lab(i,o) = sum{h,HOU, V1LABINC(i,o,h)*w1labinc(i,o,h)}
      + IF[ V1LAB(i,o)=0, labslack(i,o)];
```

```
E_w1labh_i # HH Labour income # (all,h,HOU)(all,o,OCC)
      ID01[V1LABOH_I(h,o)]*w1labh_i(h,o) = sum{i,IND,V1LABINC(i,o,h)
      *w1labinc(i,o,h)};
```

```
E_w1labinc_io # Total labour income of hh # (all,h,HOU)
      ID01[V1LABINC_IO(h)]*w1labinc_io(h) = sum{o,OCC, V1LABOH_I(h,o)
      *w1labh_i(h,o)};
```

```
!E_w1labinc_io # Total labour income of hh # (all,h,HOU)
      sum{o,OCC, V1LABOH_I(h,o)*[w1labh_i(h,o)-w1labinc_io(h)]} = 0;!
```

```
E_w1labh_oh # Labour bills # (all,i,IND)
      [TINY+V1LABH_OH(i)]*w1labh_oh(i) = sum{o,OCC, sum{h,HOU,
      V1LABINC(i,o,h)*w1labinc(i,o,h)}};
```

Write

```
V1LABINC to file SUMMARY header "LINC";
V1LABH_OH to file SUMMARY header "L_OH";
V1LABOH_I to file SUMMARY header "LOHI";
V1LABINC_IO to file SUMMARY header "LIIO";
```

5.4. Household income

Excerpt 46 contains equations that calculate pre- and post- tax household income. It begins by determining each component of household income, with the first two equations representing government transfers to households and household receipts from the rest of the world. By default, both of these variables follow movements in GDP — we have no real theory²⁸ to explain them. In contrast, the third equation postulates that transfers from one household to another are proportional to the post-tax income of the donor. With all household income components accounted for, the fifth equation then determines each household's pre-tax income.

The final block of equations in Excerpt 46 starts by calculating each household's disposable income, which is derived after deducting direct income tax and other non-tax related transfers to the government. The income tax rate equation $E_wtaxhou$ includes two shift variables (shifters) in order to allow for possible changes in tax structure. The first shifter $f_inctaxrate(h)$ permits household specific tax rate changes whereas $f_inctaxrate_h$ is an overall shifter which enforces identical changes in tax rates across all households. On the other hand, the next, $E_wgovhou$, equation determining each household's non-tax related transfers to government could be operated in two variants. The first option specifies that non-tax payments are proportional to that household's pre-tax income, while the second, commented-out, option introduces the two household-specific tax shift variables just described. Activating this option implies that changes in household non-tax transfer payments to the government follow exogenous changes in income tax structure.

Finally, the last five equations in Excerpt 46 respectively compute the economy-wide pre- and post tax household income, the average income tax rate imposed on all households, as well as real income and real disposable income for each household.

```
! Excerpt 46 of TABLO input file: !
! Household Income, disposable income, savings and tax payments !
Variable
    (all,h,HOU) winchou(h) # Pre-tax h'hold income #;
    (all,h,HOU) whougov(h) # Gov transfers to households #;
    (all,hto,HOU)(all,hfrm,HOU) whouhou(hto,hfrm) # Intra-hh transfers#;
    (all,h,HOU) whourow(h) # Gov transfers to households #;
    (all,h,HOU) wgovhou(h) # Hh trasnfers to govt#;
    (all,h,HOU) wtaxhou(h) # Household tax payments #;
    (all,h,HOU) wdispinc(h) # Post-tax h'hold income #;
Coefficient
    (all,h,HOU) VINCHOU(h) # Pre-tax h'hold income #;
    (all,h,HOU) VHOUGOV(h) # Gov transfers to households #;
    (all,hto,HOU)(all,hfrm,HOU) VHOUHOU(hto,hfrm) # Intra-hh transfers#;
    (all,h,HOU) VHOUROW(h) # Row transfers to households #;

    (all,h,HOU) VDISPINC(h) # Post-tax h'hold income #;
    (all,h,HOU) VGOVHOU(h) # Hh trasnfers to govt#;
    (all,h,HOU) VTAXHOU(h) # Hh income tax to govt#;
Read
    VHOUGOV from file BASEDATA header "VHGV";
    VHOUHOU from file BASEDATA header "VHHO";
    VHOUROW from file BASEDATA header "VHRW";
    VGOVHOU from file BASEDATA header "VGVH";
    VTAXHOU from file BASEDATA header "VTXH";
Update
    (all,h,HOU) VHOUGOV(h) = whougov(h);
    (all,hto,HOU)(all,hfrm,HOU) VHOUHOU(hto,hfrm) = whouhou(hto,hfrm);
    (all,h,HOU) VHOUROW(h) = whourow(h);
    (all,h,HOU) VGOVHOU(h) = wgovhou(h);
    (all,h,HOU) VTAXHOU(h) = wtaxhou(h);
Formula
    (all,h,HOU) VINCHOU(h) = VHOUGOS(h) + VHOUEENT(h)
```

²⁸ A more interesting theory could easily be added, for applications where these flows were pivotal.

```

+ V1LABINC_IO(h) + VHOUGOV(h)
+ sum{hfrom,HOU, VHOUHOU(h,hfrom)} + VHOUROW(h);
(all,h,HOU) VDISPINC(h) = VINCHOU(h) - VTAXHOU(h) - VGOVHOU(h);
Equation
E_whougov # Gov transfers to households # (all,h,HOU)
whougov(h) = w0gdpepx; ! default assumption !

E_whouhou # Intra-hh transfers# (all,hto,HOU)(all,hfrm,HOU)
whouhou(hto,hfrm) = wdispinc(hfrm);
! transfer proportional to post-tax donor income !

E_whourow # Row transfers to households # (all,h,HOU)
whourow(h) = w0gdpepx; ! default assumption !

E_winchou # Pre-tax household income # (all,h,HOU)
VINCHOU(h)*winchou(h) = VHOUGOS(h)*whougos(h) + VHOUEHT(h)*whouent(h) +
VHOUGOV(h)*whougov(h) +
V1LABINC_IO(h)*w1labinc_io(h) +
sum{hfrom,HOU, VHOUHOU(h,hfrom)*whouhou(h,hfrom)}
+ VHOUROW(h)*whourow(h);

Variable !Post tax household income!
winchou_h # Total pre-tax h'hold income #;
(all,h,HOU) f_inctaxrate(h) # Income tax shifter: by income #;
f_inctaxrate_h # Income tax shifter: overall #;
wdispinc_h # Total post-tax h'hold income #;
avetax_h # Average Tax Factor: avediswager - avewager #;
(all,h,HOU) xinchou(h) # Real household income #;
(all,h,HOU) xdispinc(h) # Real household income #;

Equation
E_wdispinc # Post-tax household income # (all,h,HOU)
VDISPINC(h)*wdispinc(h) = VINCHOU(h)*winchou(h) - VGOVHOU(h)*wgovhou(h)
- VTAXHOU(h)*wtaxhou(h);

E_wtaxhou # Households to gov: income taxes # (all,h,HOU)
wtaxhou(h) = winchou(h) + f_inctaxrate(h) + f_inctaxrate_h;
! note: f_inctaxrate(h), and f_inctaxrate_h are % changes in ad valorem rates !

E_wgovhou # Households to gov: transfers # (all,h,HOU)
wgovhou(h) = winchou(h);
! wgovhou(h) = winchou(h) + f_inctaxrate(h) + f_inctaxrate_h;!

E_winchou_h # Total pre-tax household income #
sum{h,HOU, VINCHOU(h)*[winchou(h) - winchou_h]} = 0;

E_wdispinc_h # Total post-tax h'hold income #
sum{h,HOU, VDISPINC(h)*[wdispinc(h) - wdispinc_h]} = 0;

E_avetax_h # Average tax factor #
wdispinc_h = winchou_h + avetax_h;

E_xinchou # Real household income# (all,h,HOU)
xinchou(h) = winchou(h) - p3toth(h);

E_xdispinc # Real post-tax household income # (all,h,HOU)
xdispinc(h) = wdispinc(h) - p3toth(h);

Write
VHOUGOV to file SUMMARY header "VHGV";
VHOUHOU to file SUMMARY header "VHHO";
VHOUROW to file SUMMARY header "VHRW";
VGOVHOU to file SUMMARY header "VGVH";

```

```
VTAXHOU to file SUMMARY header "VTXH";
VINCHOU to file SUMMARY header "VSIH";
```

5.5. Household consumption function, savings and other transfers

The first equation in Excerpt 47, E_f3toth, links household consumption to disposable income and to two exogenous shift variables: a household-specific shifter f3toth(h) which allows specific consumption changes for each household; and an overall shifter f3tot_h which imposes identical movements in consumption for all households. The equation, E_wrowhou pertains to household transfers to the rest of the world, which move with disposable income. Finally, the identity equation E_delSAVHOU determines household savings as a residual after deducting expenses from gross income.

```
! Excerpt 47 of TABLO input file: !
! Household consumption function, savings and other transfer payments!
Variable
(all,h,HOU) f3toth(h) # Consumption function ratio [consumption/GDP]#;
              f3tot_h # Over-all shifter for consumption #;
(all,h,HOU) wrowhou(h) # Household transfers to row #;
(change)(all,h,HOU) delSAVHOU(h) # Household saving #;
Coefficient
  (all,h,HOU) VROWHOU(h) # Household transfers to row #;
  (all,h,HOU) VSAVHOU(h) # Household saving #;
Read
  VROWHOU from file BASEDATA header "VRWH";

Update
  (all,h,HOU) VROWHOU(h) = wrowhou(h);

Formula
  (all,h,HOU) VSAVHOU(h) = VINCHOU(h) - V3TOT(h) -
                        sum{hto,HOU, VHOUHOU(hto,h)}
                        - VENTHOU(h) - VGOVHOU(h) - VTAXHOU(h) - VROWHOU(h);

Equation
  E_f3toth # Consumption function # (all,h,HOU)
    w3toth(h) = f3toth(h) + wdispinc(h) + f3tot_h;

  E_wrowhou # Household transfers to row # (all,h,HOU)
    wrowhou(h) = wdispinc(h); ! default rule !

  E_delSAVHOU # Household saving # (all,h,HOU)
    100*VSAVHOU(h)*delSAVHOU(h) = VINCHOU(h)*winchou(h) - V3TOT(h)*w3toth(h)
      - sum{hto,HOU, VHOUHOU(hto,h)*whouhou(hto,h)}
      - VGOVHOU(h)*wgovhou(h) - VTAXHOU(h)*wtaxhou(h)
      - VENTHOU(h)*wenthou(h) - VROWHOU(h)*wrowhou(h);

Write
  VROWHOU to file SUMMARY header "VRWH";
  VSAVHOU to file SUMMARY header "VSVH";
  V3TOT to file SUMMARY header "3TOT";
```

5.6. Government Income

Excerpt 48 determines government income. The first equation, E_wgovrow specifies that government receipts from the rest of the world (ROW) follow GDP, while the final equation E_wincgov provides an add up of components of government income.

```
! Excerpt 48 of TABLO input file: !
! Fill in government row (income) !
```

```

! Apart from VROWGOV, all entries are already determined !
Variable
  wgovrow # Transfers from ROW to gov #;
  wincgov # Government income #;

Coefficient
  VGOVROW # Transfers from ROW to gov #;
  VINGGOV # Government income #;

Read
  VGOVROW from file BASEDATA header "VGVR";

Update  VGOVROW = wgovrow;

Formula VINGGOV = V0TAX_CSI + VGOVGOS + VGOVENT + VTAXENT + VGOVROW
          + sum{h,HOU, VGOVHOU(h) + VTAXHOU(h)};

Equation
  E_wgovrow # Transfers from ROW to gov #
  wgovrow = w0gdpexp; ! default rule !

  E_wincgov # Government income #
  VINGGOV*wincgov = V0TAX_CSI*w0tax_csi + VGOVGOS*wgovgos
                  + VTAXENT*wtaxent + VGOVENT*wgovent
                  !- V1SUB_I*w1sub_i!
                  + sum{h,HOU, VGOVHOU(h)*wgovhou(h)
                      + VTAXHOU(h)*wtaxhou(h)} + VGOVROW*wgovrow;

Write
  VINGGOV to file SUMMARY header "VING";

```

5.7. Government expenditure

The coefficients and variables associated with Excerpt 49 relate to government expenditures and savings. The first equation $E_wrowgov$ links government transfers to the world with GDP, while the next equation, $E_wcurgov$ adds up current government expenditure. The next equation, $E_wcapgov$, determines government investment expenditures by industry using initial investment shares.

The final three equations refer to government savings. Initially, identity $E_wsavgov$ derives government savings as a residual between income and expenditure. Because government budget deficit (negative savings) is possible, the next equation $E_delGOVSAV$ determines $delgovsav$, which derives the ordinary change in government savings as a ratio of GDP. Note that $delgovsav$ is the [change in the] ratio of two nominal variables. The final equation, $E_realsavgov$ calculates real government savings, which is nominal savings deflated by the GDP price.

```

! Excerpt 49 of TABLO input file: !
! Find current gov expenditure and capital gov expenditure !
! Hence find gov saving - might be negative in levels !
! Government expenditure !
Variable
  wcurgov # Current gov expenditure #;
  wrowgov # GOV transfers to ROW #;
  wcapgov # Investment gov expenditure #;
(all,i,IND) s2gov(i) # Gov share of investment by industry #;
  wexpgov # Total gov expenditure #;
  wsavgov # Gov (income - expenditure) #;
!
  (change) wgosgov # Gov payments to GOS #;!
  (change) delsavgov # Gov (income - expenditure)/GDP#;
  realsavgov # Real gov (income - expenditure)#;

Coefficient
  VROWGOV # GOV transfers to ROW #;
  VCURGOV # Current gov expenditure #;

```

```

                VCAPGOV # Investment gov expenditure #;
(all,i,IND) GOVSHRINV(i) # Gov share of investment by industry #;
                VEXPGOV # Total gov expenditure #;
!                VGOSGOV # GOV GOS payments#!;
                VSAVGOV # Gov (income - expenditure) #;

Read
VROWGOV from file BASEDATA header "VRGV";
!VGOSGOV from file BASEDATA header "VSGV";!
GOVSHRINV from file BASEDATA header "GVSH";

Update
                VROWGOV = wrowgov;
!                VGOSGOV = wgosgov;!
(all,i,IND) GOVSHRINV(i) = s2gov(i);

Formula
VCURGOV = V5TOT !+ VGOSGOV! + VROWGOV + sum{h,HOU, VHOUGOV(h)};
VCAPGOV = sum{i,IND, GOVSHRINV(i)*V2TOT(i)};
VEXPGOV = VCURGOV + VCAPGOV;
VSAVGOV = VINCGOV - VEXPGOV;

Equation
E_wrowgov # GOV transfers to ROW #
wrowgov = w0gdpepx; ! default rule !

E_wcurgov # Current gov expenditure #
VCURGOV*wcurgov = V5TOT*w5tot + VROWGOV*wrowgov
+ sum{h,HOU, VHOUGOV(h)*whougov(h)};

E_wcapgov # Investment gov expenditure #
VCAPGOV*wcapgov = sum{i,IND, GOVSHRINV(i)*V2TOT(i)*
[s2gov(i) + p2tot(i) + x2tot(i)]};
! assume exogenous s2gov(i) # gov share of investment by industry # !

E_wexpgov # Total gov expenditure #
VEXPGOV*wexpgov = VCURGOV*wcurgov + VCAPGOV*wcapgov;

E_wsavgov # Gov (income - expenditure) #
VSAVGOV*wsavgov = VINCGOV*wincgov - VEXPGOV*wexpgov;

E_delSAVGOV # Gov (income - expenditure)/GDP #
100*V0GDPEXP*delSAVGOV = VSAVGOV*wsavgov - V0GDPEXP*w0gdpepx;

E_realsavgov # Real gov (income - expenditure)#
realsavgov = wsavgov - p0gdpepx;

Write VSAVGOV to file SUMMARY header "GSAV";
VCAPGOV to file SUMMARY header "GCAP";

```

5.8. Private investment expenditure

The sole equation in Excerpt 50, E_wcappriv, determines aggregate private investment expenditure as a residual between total investment (inclusive of inventories) and government investment expenditure.

```

! Excerpt 50 of TABLO input file: !
! Find investment private expenditure !
! Private investment finance requirement is just the negative of this !
Coefficient
VCAPPRIV # Investment private expenditure #;

Formula
VCAPPRIV = V2TOT_I - VCAPGOV + V6TOT;

```

Variable

wcappriv # Investment private expenditure #;

Equation

E_wcappriv # Investment private expenditure #

$$VCAPPRIV*wcappriv = V2TOT_I*w2tot_i - VCAPGOV*wcapgov + V6TOT*w6tot;$$

5.9. Rest of the world

Excerpt 51 summarises the link between the domestic economy and the rest of the world. The first two equations E_wexprow and E_wrowinc respectively identify receipts from and payments to the rest of the world. The penultimate identity E_wsavrow derives the current account balance (or foreign savings) as a residual between total receipts and payments.

! Excerpt 51 of TABLO input file: !

! Find ROW row and column sums !

Variable

wexprow # Total ROW expenditure #;
wrowinc # Total ROW income #;
wsavrow # ROW (income - expenditure) #;

Coefficient

VEXPROW # Total ROW expenditure #;
VROWINC # Total ROW income #;
VSAVROW # ROW (income - expenditure)-Current Account Balance #;

Formula

$$VEXPROW = V4TOT + VGOVROW + VENTROW + \text{sum}\{h, HOU, VHOURROW(h)\};$$

$$VROWINC = \text{sum}\{h, HOU, VROWHOU(h)\} + VROWGOV + V\emptyset CIF_C + VROWENT;$$

$$VSAVROW = VROWINC - VEXPROW;$$

Equation

E_wexprow # Total ROW expenditure #

$$VEXPROW*wexprow = V4TOT*w4tot + VGOVROW*wgovrow + VENTROW*wentrow + \text{sum}\{h, HOU, VHOURROW(h)*whourrow(h)\};$$

E_wrowinc # Total ROW income #

$$VROWINC*wrowinc = \text{sum}\{h, HOU, VROWHOU(h)*wrowhou(h)\} + VROWENT*wrowent + V\emptyset CIF_C*w\emptyset cif_c + VROWGOV*wrowgov;$$

E_wsavrow # ROW (income - expenditure) - Current Account Balance #

$$VSAVROW*wsavrow = VROWINC*wrowinc - VEXPROW*wexprow;$$

Write

VROWINC to file SUMMARY header "VRWI";
VEXPROW to file SUMMARY header "VEXR";
VGOVROW to file SUMMARY header "VGVR";
VSAVROW to file SUMMARY header "VSRW";
V4TOT to file SUMMARY header "4TOT";

5.10. SAM consistency

At this point, we have specified equations that ensure SAM balance (row total = column total) for all SAM accounts in Figure 5 except the savings/investment accounts. Necessarily, if all but one SAM accounts balance, so too does the last. Thus we cannot *enforce* this last identity - we can merely check that it does indeed hold. That is the task of Excerpt 52: it provides a consistency check that total income is equal to total expenditure in both levels and in percent changes. The former is verified through an

Assertion statement on the coefficient VSAMCHECK, while the latter is checked via equation E_wsamcheck. If accounting equations have been correctly coded, the variable wsamcheck (=100*delVSAMCHECK/GDP) should be close to zero.

```

! Excerpt 52 of TABLO input file: !
! Check Accounting !

! Check if this is so, both in levels and in changes !
Variable
wsamcheck # Global (income - expenditure) #;

Coefficient
VSAMCHECK      # Global (income - expenditure) #;
(all,i,IND) PURE_PROFITS(i) # PURE_PROFITS, should be zero #;

Formula
VSAMCHECK = sum{h,HOU, VSAVHOU(h)} + VSAVENT + VSAVGOV
           - VCAPPRIV + VSAVROW;
(all,i,IND) PURE_PROFITS(i) = V1TOT(i) - sum{co,COSTCAT,COSTMAT(i,co)};

Assertion ABS[VSAMCHECK]<0.9;

Equation
E_wsamcheck # Global (income - expenditure) #
V0GDPEXP*wsamcheck = sum{h,HOU, 100*VSAVHOU(h)*delSAVHOU(h)}
                   + VSAVENT*wsavent + VSAVGOV*wsavgov
                   - VCAPPRIV*wcappriv + VSAVROW*wsavrow;
! note wsamcheck is expressed as a % of GDP: it should be tiny !

Write
PURE_PROFITS to file SUMMARY header "PURE";
VSAMCHECK to file SUMMARY header "SCHK";

```

5.11. Revenue Neutrality

Excerpt 53 presents additional equations for undertaking government revenue-neutral simulations. The initial equation E_wtaxtot sums up government tax revenue (commodity and income tax revenue), with the succeeding equation E_rtaxtot deriving total tax revenue in real terms. The final equation E_f_entinctax is determined by two exogenous shift variables: tax on enterprises, ftaxent and an over-all household tax shifter, f_inctaxrate_h.

By default, PHILGEM does not impose revenue neutrality. Nonetheless, the user can specify 3 alternate rules for undertaking revenue-neutral simulations. First, swapping f_inctaxrate_h with rtaxtot enforces changes in government income to be compensated by identical changes in tax rates across all households. Second, endogenizing ftaxent while holding rtaxtot exogenous results in enterprise tax rates compensating for changes in government income. Finally, with f_entinctax exogenous and either ftaxent or f_inctaxrate_h endogenous, we force enterprises and households taxes to move together.

```

! Excerpt 53 of TABLO input file: !
!Add conditions for revenue neutrality!

Coefficient
VTAXTOT      # Sum of income and commodity tax revenue #;

Formula
VTAXTOT = sum{h, HOU, VTAXHOU(h)} + VTAXENT + V0TAX_CSI;

Variable
wtaxtot      # Sum of income and commodity tax revenue #;
rtaxtot      # Real tax collection: for nom homogeniety test #;
f_entinctax  # Shifter: tax on enterprises, Fgostax follows f_inctaxrate_h #;

```

Equation

E_wtaxtot # Sum of income and commodity tax revenue #

$$VTAXTOT*wtaxtot = \sum\{h, HOU, VTAXHOU(h)*wtaxhou(h)\} + VTAXENT*wtaxent + V0TAX_CSI*w0tax_csi;$$

E_rtaxtot # Total real tax: for nom homogeniety test #

$$rtaxtot = wtaxtot - p5tot;$$

!Tax on enterprises !

E_f_entinctax # Shifter: tax on enterprises, Fgostax follows f_inctaxrate_h #

$$f_entinctax = ftaxent - f_inctaxrate_h;$$

![[! Normally f_inctaxrate_h exogenous; whouinc(h) given from model,

so this determines whougov.

Rule 1: Swap f_inctaxrate_h with rtaxtot enforces changes in government income to be compensated by identical changes in tax rates across all hhss.

Rule 2: Swap f_ftaxent with rtaxtot enforces changes in government income to be compensated by changes in tax rates on enterprises.

Rule 3: By exogenizing f_entinctax and endogenizing ONE of f_inctaxrate_h and ftaxent we can force ftaxent and f_inctaxrate_h to move together !]]!

!-----!

5.12. Deriving a Macro SAM

Excerpt 55 contains additional calculations needed to create an aggregate SAM for reporting purposes. Initially, individual row and column accounts in the SAM are defined. We then derive the value flows generated and assign them to a specific cell in the aggregated SAM. Excerpt 55 ends by checking that each column sum equals its corresponding row sum.

! Excerpt 54 of TABLO input file: !

! Construct macro SAM !

Set

MSAM # Macro SAM accounts # (Firms,DomCom,ImpCom,Labour,Capital,ProdTax,ComTax,Tariff,DirTax, Households,Enterprises,GovCurrent, GovInvest,PrvInvest,Stocks,ROW);

MTAXES # All tax accounts # (ProdTax,ComTax,Tariff,DirTax);

Set MainUser # Main users # (Firms, Households, Enterprises, GovCurrent, GovInvest,PrvInvest, Stocks, ROW);

Subset MainUser is subset of MSAM;

Subset MTAXES is subset of MSAM;

Coefficient

(all,i,IND) PRVSHRINV(i) # Private share of investment by industry #;

(all,i,IND) DOMINV(i) # Investment use of dom goods #;

(all,i,IND) IMPINV(i) # Investment use of imp goods #;

Formula

(all,i,IND) PRVSHRINV(i) = 1.0 - GOVSHRINV(i);

(all,i,IND) DOMINV(i) = sum{c,COM, V2BAS(c,"dom",i) + sum{s,SRC, sum{m,MAR, V2MAR(c,s,i,m) }}};

(all,i,IND) IMPINV(i) = sum{c,COM, V2BAS(c,"imp",i)};

Coefficient (all,u,MainUser) DOMUSE(u) # all use dom, basic + margins #;

Formula

DOMUSE("Firms") = sum{i,IND, sum{c,COM, V1BAS(c,"dom",i) + sum{s,SRC, sum{m,MAR, V1MAR(c,s,i,m) }}}};

DOMUSE("Households") = sum{c,COM, V3BAS(c,"dom") + sum{s,SRC, sum{m,MAR, V3MAR(c,s,m) }}};

DOMUSE("GovCurrent") = sum{c,COM, V5BAS(c,"dom")};

DOMUSE("GovInvest") = sum{i,IND, GOVSHRINV(i)*DOMINV(i)};

DOMUSE("PrvInvest") = sum{i,IND, PRVSHRINV(i)*DOMINV(i)};

DOMUSE("ROW") = sum{c,COM, V4BAS(c) + sum{m,MAR, V4MAR(c,m)}};

```
DOMUSE("Stocks") = sum{c,COM, V6BAS(c,"dom") };
```

Coefficient (all,u,MainUser) IMPUSE(u) # Imports at basic prices #;

Formula

```
IMPUSE("Firms") = sum{c,COM,sum{i,IND, V1BAS(c,"imp",i)}};
IMPUSE("Households")= sum{c,COM,V3BAS(c,"imp")};
IMPUSE("GovCurrent")= sum{c,COM,V5BAS(c,"imp")};
IMPUSE("GovInvest") = sum{i,IND, GOVSHRINV(i)*IMPINV(i)};
IMPUSE("PrvInvest") = sum{i,IND, PRVSHRINV(i)*IMPINV(i)};
IMPUSE("ROW") = 0;
IMPUSE("Stocks") = sum{c,COM,V6BAS(c,"imp")};
```

Coefficient (all,i,IND) V2TAX_CS(i) # Taxes on investment #;

Formula (all,i,IND) V2TAX_CS(i) = sum{c,COM, sum{s,SRC, V2TAX(c,s,i)}};

Coefficient

```
(All,r,MSAM)(All,c,MSAM) SAM(r,c) # Macro SAM #;
(All,t,MTAXES) TAXREV(t) # Tax revenue#;
```

Formula

```
(All,r,MSAM)(All,c,MSAM) SAM(r,c) = 0.0;
```

! Use of labour and capital !

```
SAM("Labour","Firms") = V1LAB_IO;
```

```
SAM("Capital","Firms") = V1CAP_I + V1LND_I;
```

! Use of Domestic Commodities !

```
(all,u,MainUser) SAM("DomCom",u) = DOMUSE(u);
```

! Use of Imported Commodities !

```
(all,u,MainUser) SAM("ImpCom",u) = IMPUSE(u);
```

Formula

! Commodity tax payments!

```
SAM("ComTax","Firms") = V1TAX_CSI;
```

```
SAM("ProdTax","Firms") = V1PTX_I + V1OCT_I;
```

```
SAM("ComTax","Households") = V3TAX_CS;
```

```
SAM("ComTax","GovCurrent") = V5TAX_CS;
```

```
SAM("ComTax","GovInvest") = sum{i,IND, GOVSHRINV(i) * V2TAX_CS(i)};
```

```
SAM("ComTax","PrvInvest") = sum{i,IND, PRVSHRINV(i) * V2TAX_CS(i)};
```

```
SAM("ComTax","ROW") = sum{c,COM, V4TAX(c)};
```

! Payment of Import duties !

```
SAM("Tariff","ImpCom") = V0TAR_C;
```

! Income tax !

```
SAM("DirTax","Households") = sum{h,HOU, VTAXHOU(h)};
```

```
SAM("DirTax","Enterprises")= VTAXENT;
```

! Transfers to government !

```
SAM("GovCurrent","Households") = sum{h,HOU, VGOVHOU(h)};
```

```
SAM("GovCurrent","Enterprises") = VGOVENT;
```

! Household receipts !

```
SAM("Households","Labour") = sum{h,HOU,V1LABINC_IO(h)};
```

```
SAM("Households","Capital") = VHOUGOS_H;
```

```
SAM("Households","Enterprises") = VHOUEVENT_H;
```

```
SAM("Households","Households") = sum{a,HOU,sum{b,HOU, VHOUHOU(a,b)}};
```

```
SAM("Households","GovCurrent") = sum{h,HOU, VHOUGOV(h)};
```

```
SAM("Households","ROW") = sum{h,HOU, VHOUEVENT_H};
```

! Other Household payments !

```
SAM("Enterprises","Households") = sum{h,HOU, VENTHOU(h)};
```

```
SAM("PrvInvest","Households") = sum{h,HOU, VSAVHOU(h)};
```

! Enterprises Receipts !

```
SAM("Enterprises","Capital") = VENTGOS;
```

```
SAM("Enterprises","ROW") = VENTROW;
```

! Other Enterprises payments !

```
SAM("ROW","Enterprises") = VROWENT;
```

```
SAM("PrvInvest","Enterprises") = VSAVENT;
```

! Government receipts!

```
SAM("GovCurrent","Capital") = VGOVGOS;
```

```
SAM("GovCurrent","ROW") = VGOVROW;
```

```
(All,t,MTAXES) TAXREV(t) = sum{c,MSAM, SAM(t,c)};
```

```
(all,t,MTAXES) SAM("GovCurrent",t) = TAXREV(t);
```

! Government payments!

```
SAM("ROW","GovCurrent") = VROWGOV;
```

```
SAM("PrvInvest","GovCurrent") = VSAVGOV;
```

```

SAM("GovInvest","GovCurrent") = sum{r,MSAM, SAM(r,"GovInvest")};
! Others !
SAM("Firms","DomCom")          = sum{c,MSAM, SAM("DomCom",c)};!MAKE=SALES!
SAM("ROW","ImpCom")            = sum{c,MSAM, SAM("ImpCom",c)}-V0TAR_C;!CIF!
SAM("Stocks","PrvInvest")      = sum{r,MSAM, SAM(r,"Stocks")};
!Foreign Savings!
  SAM("PrvInvest","ROW")       = VSAVROW;

```

Coefficient

```
(all,rc,MSAM) VMSAMCHECK(rc) # Global check on macro SAM #;
```

Formula

```
(all,rc,MSAM) VMSAMCHECK(rc) = sum{c,MSAM, SAM(rc,c)} -
                               sum{r,MSAM, SAM(r,rc)};
```

Write

```

VMSAMCHECK to FILE SUMMARY header "SAMC";
SAM         to FILE SUMMARY header "MSAM";
(postsim) VMSAMCHECK to FILE SUMMARY header "USMC"
           longname "Post-update Global check on macro SAM";
(postsim) SAM to FILE SUMMARY header "UMSM"
           longname "Post-update Macro SAM";

```

```
Assertion (all,rc,MSAM) ABS[VMSAMCHECK(rc)]<0.9;
```

5.13. Condensation

The model as described so far has too many equations and variables for efficient solution. Their numbers are reduced in Excerpt 56 by instructing the TABLO program to:

- omit specified variables from the system. This option is useful for variables which will be exogenous and unshocked (zero percentage change). Normally it allows us to dispense with the bulk of the technical change terms. Of course, the particular selection of omitted variables will alter in accordance with the model simulations to be undertaken.
- substitute out specified variables using specified equations. This results in fewer but more complex equations. Typically we use this method to eliminate multi-dimensional matrix variables which are defined by simple equations. For example, the equation:

```

Equation E_x1_s # Demands for Commodity Composites #
(All,c,COM)(All,i,IND) x1_s(c,i) - {a1_s(c,i) + a1tot(i)} = x1tot(i);

```

which appears in Excerpt 10 of the TABLO Input file, can be used to substitute out variable $x1_s$. In fact the names of the PHILGEM equations are chosen to suggest which variable each equation could eliminate.²⁹

- backsolve allows values of a variable that was substituted out to be computed after a simulation. For example, the substituted variable $x1_s$ in the equation above can be calculated by using the known value of $x1tot$ found in the right hand side of equation E_x1_s . Note that values for $a1_s$ and $a1tot$ are not used since they are part of the omitted list.

```
! Excerpt 55 of TABLO input file: !
! Condensation !
```

```

Omit      a1 a1oct a1mar a1_s a2 a2mar a2_s a3 a3mar a4mar a5mar;
Backsolve p1      using E_p1;
Backsolve p1lab  using E_p1lab;
Substitute pq1   using E_pq1;
Backsolve p1_s   using E_p1_s;
Substitute p2    using E_p2;
Substitute p2_s  using E_p2_s;

```

²⁹ Substituted variables are necessarily endogenous and are not stored in the results file. GEMPACK also allows *back-substituted* variables; these do appear in the results file but use slightly more computing resources.

GEMPACK only allows a variable to be substituted using a single equation of just the same dimensions. For a larger model, it is very important that the model equations and variables are arranged in a way that allows large matrix variables to be substituted or back-substituted.

```

Backsolve p3      using E_p3;
Backsolve p5      using E_p5;
Backsolve q1      using E_q1;
Substitute t1     using E_t1;
Substitute t2     using E_t2;
Backsolve t3      using E_t3;
Backsolve t5      using E_t5;
Backsolve x1      using E_x1;
Backsolve x1lab   using E_x1lab;
Substitute x1mar  using E_x1mar;
Backsolve x1_s    using E_x1_s;
Backsolve x2      using E_x2;
Substitute x2mar  using E_x2mar;
Substitute x2_s   using E_x2_s;
Backsolve x3      using E_x3;
Backsolve x3mar   using E_x3mar;
Backsolve x4mar   using E_x4mar;
Backsolve x5      using E_x5;
Backsolve x5mar   using E_x5mar;
Backsolve x1oct   using E_x1oct;
Backsolve p1oct   using E_p1oct;
Backsolve p1var   using E_p1var;
Backsolve p1cst   using E_p1cst;
Backsolve p1mat   using E_p1mat;
Backsolve x3sub   using E_x3sub;
Backsolve x3lux   using E_x3lux;
Backsolve a3sub   using E_a3sub;
Backsolve a3lux   using E_a3lux;
Substitute delV1TAX using E_delV1TAX;
Substitute delV2TAX using E_delV2TAX;
Backsolve delV3TAX using E_delV3TAX;
Backsolve delV5TAX using E_delV5TAX;
Backsolve delV6   using E_delV6;
Substitute x3_sh  using E_x3_sh;

```

5.14. Household results for reporting purposes

Excerpt 56 is an optional component of PHILGEM. It contains variables to report household expenditure and income into different household groups. In particular, results for the 20 (=10 deciles X gender of household head) households of set HOU are translated into results reported by decile (DEC) and gender (GEN). The mappings, from HOU to DEC and from HOU to GEN, are read from BASEDATA input file.

The first three equations in Excerpt 54 report results for real household consumption. The first equation simply sets the new matrix variable $x3totq$, dimensioned DEC x GEN, to corresponding elements of the HOU-length vector $x3toth$. The next two equations, E_x3totq_d and E_x3totq_g then define real household consumption by gender and decile respectively as share-weighted averages. From the following equivalent equation patterns:

$$x = \sum[V_i/V]x_i \quad V \cdot x = \sum V_i x_i \quad \sum V_i x = \sum V_i x_i \quad 0 = \sum V_i [x_i - x]$$

we chose the last, terser, form (it avoids defining V).

The remaining equations of Excerpt 54, reporting income and disposable income, follow a similar pattern.

```

! Excerpt 56 of TABLO input file: !
! Optional: Could be commented out !
!Equations reporting changes in household consumption!
Set
DEC # Income types # (Dec01-Dec10);
GEN # Gender      # (Fem,Mal);

Mapping HOU2DEC from HOU to DEC;
Mapping HOU2GEN from HOU to GEN;

Read (by_elements) HOU2DEC from file BASEDATA header "H2DC";
Read (by_elements) HOU2GEN from file BASEDATA header "H2GN";

```

Variable

```
(all,d,DEC)(all,g,GEN) x3totq(d,g) # Household consumption #;
      (all,g,GEN) x3totq_d(g) # Hh cons by gender #;
      (all,d,DEC) x3totq_g(d) # Hh cons by decile #;
```

Coefficient

```
(all,d,DEC)(all,g,GEN) V3TOTQ(d,g) # Nominal hh consumption #;
```

Formula

```
(all,d,DEC)(all,g,GEN)
      V3TOTQ(d,g) = sum{h,HOU: [HOU2DEC(h)=d] and [HOU2GEN(h)=g], V3TOTQ(h)};
```

Equation

```
E_x3totq # Household consumption # (all,h,HOU)
      x3totq(HOU2DEC(h),HOU2GEN(h)) = x3totq(h);
```

```
E_x3totq_d # Household consumption by gender # (all,g,GEN)
      0 = sum{d,DEC, V3TOTQ(d,g)*[x3totq(d,g) - x3totq_d(g)]};
```

```
E_x3totq_g # Hh consumption by decile # (all,d,DEC)
      0 = sum{g,GEN, V3TOTQ(d,g)*[x3totq(d,g) - x3totq_g(d)]};
```

! Equations reporting changes in household income!

Coefficient

```
(all,d,DEC)(all,g,GEN) VINCHOUQ(d,g) # Household income #;
      (all,d,DEC)(all,g,GEN) VDISPINCQ(d,g) # Hh post-tax income #;
```

Variable

```
(all,d,DEC)(all,g,GEN) xinchouq(d,g) # Real Household income #;
      (all,g,GEN) xinchouq_d(g) # Real Hh income by gender #;
      (all,d,DEC) xinchouq_g(d) # Real Hh income by decile #;
(all,d,DEC)(all,g,GEN) xdispincq(d,g) # Real Household income #;
      (all,g,GEN) xdispincq_d(g) # Real Hh disp income by gender #;
      (all,d,DEC) xdispincq_g(d) # Real Hh disp income by decile #;
```

Formula

```
(all,d,DEC)(all,g,GEN)
      VINCHOUQ(d,g) = sum{h,HOU: [HOU2DEC(h)=d] and [HOU2GEN(h)=g], VINCHOU(h)};
```

```
(all,d,DEC)(all,g,GEN)
      VDISPINCQ(d,g) = sum{h,HOU: [HOU2DEC(h)=d] and [HOU2GEN(h)=g], VDISPINC(h)};
```

Equation

```
E_xinchouq # Real household income # (all,h,HOU)
      xinchouq(HOU2DEC(h),HOU2GEN(h)) = xinchouq(h);
```

```
E_xinchouq_d # Real income household by gender # (all,g,GEN)
      0 = sum{d,DEC, VINCHOUQ(d,g)*[xinchouq(d,g) - xinchouq_d(g)]};
```

```
E_xinchouq_g # Real hh income by decile # (all,d,DEC)
      0 = sum{g,GEN, VINCHOUQ(d,g)*[xinchouq(d,g) - xinchouq_g(d)]};
```

! Equations reporting changes in real disposable household income!

```
E_xdispincq # Real post-tax household income # (all,h,HOU)
      xdispincq(HOU2DEC(h),HOU2GEN(h)) = xdispincq(h);
```

```
E_xdispincq_dr # Real post-tax income household by gender # (all,g,GEN)
      0 = sum{d,DEC,VDISPINCQ(d,g)*[xdispincq(d,g) - xdispincq_d(g)]};
```

```
E_xdispincq_gr # Real post-tax hh income by decile # (all,d,DEC)
```

6. Closing the Model

PHILGEM's model specification has more variables than equations. To *close* the model, we choose which variables are to be exogenous and which endogenous³⁰. The number of endogenous variables must equal the number of equations. For a complex AGE model, it may be surprisingly difficult to find a sensible closure that satisfies this accounting restriction.

Table 2 allows us to pursue this task systematically. It arranges the model's 224 equations and 277 variables according to their dimensions. Equations broken into parts, such as E_x4A (covering individual export commodities) and E_x4B (covering collective exports) are treated as one equation block for this purpose. The first column lists the various combinations of set indices that occur in the model. The second column shows how many variables have these combinations. For example, 10 variables are dimensioned by COM, SRC and IND. The third column counts equations in the same way. For example, there are 56 macro, i.e., scalar, equations.

In most straightforward closures of the model, the correspondence between equations and endogenous variables applies for each row of the table, as well as in total. The fourth column shows the difference between the preceding two, i.e, it shows how many variables of that size would normally be exogenous.

In constructing the TABLO Input file, we chose to name each equation after the variable it seemed to explain or determine³¹. Some variables had no equation named after them—they appear in the fifth column. Those variables are promising candidates for exogeneity. They include:

- technical change variables, mostly beginning with the letter 'a';
- tax rate variables, mostly beginning with 't';
- shift variables, mostly beginning with 'f';
- land endowments x1lnd, and the number of households q;
- industry capital stocks, x1cap;
- foreign prices, pf0cif, and the investment slack variable, invslack;
- inventory to sales ratios, fx6;
- the exchange rate phi, which could serve as numeraire; and
- w3lux (household above-subsistence expenditure).

Although Column 5 contains a perfectly valid exogenous set for the model, we choose, for typical shortrun simulations, to adopt a slightly different closure. The macro variables italicised in Column 5 are *swapped* as follows.

- We exogenize x5tot instead of f5tot2, so disconnecting government from household consumption.
- We exogenize x2tot_i (aggregate investment), rather than invslack.
- We exogenize x3tot (household consumption), rather than w3lux.
- We exogenize delx6 (inventory changes), rather than fx6.
- We exogenize realwage (average real wage), rather than f1lab_io (overall wage shift).
- For each industry, we exogenize the corresponding element of either finv1 or finv2 in place of the x2tot vector.

These changes give rise to the closure shown in Table 3 (swapped variables are italicized).

³⁰ In GEMPACK parlance, a *closure* is a list of exogenous variables.

³¹ If equations are named after variables, the GEMPACK program TABmate will automatically perform the analysis summarized in Table 2. The equation naming system is slightly arbitrary. For example, where a supply equation and a demand equation jointly determine price and quantity, we could say that the supply equation explained the price variable, and that the demand equation explained the quantity variable. Or we could pair supply with quantity and demand with price. It would not matter which way we chose: the two equations between them endogenize both price and quantity.

To name some other equations, we must have a closure in mind. For example, the factor demand equations of Excerpt 8 are named E_x1lab_o, E_p1cap, and E_p1lnd. Thus the equation names suggest that the labour demand equation determines employment while the other two equations determine the unit rentals of capital and land. That naming choice reflects the standard ORANI short-run closure, which fixes industry use of capital and land.

Table 2 Tally of Variables and Equations

1	2	3	4	5
Dimension	Variable Count	Equation Count	Exogenous Count	List of unexplained variables (Mechanical closure)
Macro	120	104	16	capslack f1lab_io f1tax_csi f2tax_csi f3tax_cs f3tot_h f4p_ntrad f4q_ntrad f4tax_ntrad f4tax_trad f5tax_cs f5tot2 ftaxent f_inctaxrate_h invslack phi
IND	40	26	14	a1cap a1lab_o a1lnd a1oct a1prim a1tot a2tot delPTXRATE f1lab_o f1oct s2gov x1cap x1lnd x2tot
EXPMAC	1	1	0	
CONFAC	1	1	0	
CONTINC	1	1	0	
HOU	22	19	3	f_inctaxrate qh w3luxh
COM	21	16	5	f0tax_s f4p f4q pf0cif t0imp
OCC	3	2	1	f1lab_j
GEN	3	3	0	
DEC	3	3	0	
COM*IND	8	6	2	a1_s a2_s
COM*SRC	14	11	3	a3 f5 fx6
COM*HOU	6	5	1	a3_s
COM*MAR	2	1	1	a4mar
IND*OCC	5	4	1	f1lab
COM*FANCAT	1	1	0	
COM*DESTPLUS	1	1	0	
HOU*OCC	1	1	0	
HOU*HOU	1	1	0	
DEC*GEN	3	3	0	
COM*SRC*IND	10	8	2	a1 a2
COM*SRC*MAR	4	2	2	a3mar a5mar
COM*SRC*DEST	1	1	0	
IND*OCC*HOU	1	1	0	
COM*SRC*IND*MAR	4	2	2	a1mar a2mar
TOTAL	277	224	53	

A choice of closure reflects two different types of consideration. First, the closure is associated with our idea of the simulation *timescale*, that is, the period of time which would be needed for economic variables to adjust to a new equilibrium. The timescale assumption affects the way we model factor markets. For example, in a shortrun simulation we normally hold capital stocks fixed, as in Table 3. The idea is that capital stocks take some time to install—too long for them to be affected, in the short run, by the shocks. Shortrun closures often also allow for rigidities in the labour market: in this case by holding real wages fixed. The length of the 'short' run is not explicit, but is usually thought to be between 1 and 3 years.

Second, the choice of closure is affected by the needs of a particular simulation and by our view of the most appropriate assumption for those variables that the model does not explain. For example, PHILGEM provides little theory to explain the size and composition of absorption. In Table 3 the major expenditure-side aggregates are simply held fixed. Therefore, if some shock reduced GDP, the balance of trade (which is endogenous) might move towards deficit, reflecting national dis-saving. We might wish to prevent this by fixing the balance of trade, *delB*, instead of, say, *x3tot*. This would allow us to use the endogenous *x3tot* as a simple index of welfare.

Table 3 The PHILGEM short-run closure

Exogenous variables constraining real GDP from the supply side	
x1cap, x1lnd	industry-specific endowments of capital and land
capslack	allows industry-specific rates of return
a1cap, a1lab_o, a1lnd, a1prim, a1tot, a2tot	all technological change
<i>realwage</i>	average real wage
Exogenous settings of real GDP from the expenditure side	
<i>x3tot</i>	aggregate real private consumption expenditure
<i>x2tot_i</i>	aggregate real investment expenditure
<i>x5tot</i>	aggregate real government expenditure
f5	distribution of government demands
<i>delx6</i>	real demands for inventories by commodity
Foreign conditions: import prices fixed; export demand curves fixed in quantity and price axes	
pf0cif	foreign prices of imports
f4p, f4q	individual exports
f4p_ntrad, f4q_ntrad	collective exports
All tax rates are exogenous	
delPTXRATE, f0tax_s, f1tax_csi, f2tax_csi, f3tax_cs, f5tax_cs, t0imp, f4tax_trad, f4tax_ntrad, f1oct	
Distribution of investment between industries	
finv1(selected industries)	investment related to profits
finv2(the rest)	investment follows aggregate investment
Number of households and their consumption preferences are exogenous	
qh	number of households
a3_s	household tastes
Numeraire assumption	
phi	nominal exchange rate
Additional exogenous variables (SAM extension)	
Tax rates (shifter) on household and enterprises	
fenttax, f_inctaxrate, f_inctaxrate_h	income tax
Government Investment	
S2gov	government investment share
Household preference	
f3toth	Individual household consumption preferences

Table 4 shows a possible longrun closure. Differences from the shortrun closure of Table 3 are shown in italics. The main differences are:

- Capital stocks are free to adjust in such a way that fixed rates of return (gret) are maintained. An open capital market is implicitly assumed, since there is no link between capital formation and domestic saving.
- Aggregate employment is fixed and the real wage adjusts. This would be consistent with the idea that both the labour force and the rate of unemployment (NAIRU) are, in the long run, determined by mechanisms outside of the model.
- Household and government expenditure move together, to accomodate a balance of trade constraint: DelB (the balance of trade as a fraction of GDP) is fixed. The idea here is that, in the long run the rest of the world might be reluctant to fund an increased trade deficit. Aggregate investment follows the aggregate capital stock.

Table 4 A possible long-run closure

Exogenous variables constraining real GDP from the supply side	
<i>gret</i>	<i>gross sectoral rates of return</i>
<i>capslack</i>	allows industry-specific rates of return
<i>x1lnd</i>	industry-specific endowments of land
<i>a1cap a1lab_o a1lnd a1prim a1tot a2tot</i>	all technological change
<i>employ_i</i>	<i>total employment - wage weights</i>
Exogenous settings of real GDP from the expenditure side	
<i>delB</i>	<i>balance of trade/GDP</i>
<i>invslack</i>	<i>aggregate investment determined by industry specific rules</i>
<i>f5tot2</i>	<i>link government demands to total household</i>
<i>f5</i>	distribution of government demands
<i>delx6</i>	real demands for inventories by commodity
Foreign conditions: import prices fixed; export demand curves fixed in quantity and price axes	
<i>pf0cif</i>	foreign prices of imports
<i>f4p, f4q</i>	individual exports
<i>f4p_ntrad, f4q_ntrad</i>	collective exports
All tax rates are exogenous	
<i>delPTXRATE, f0tax_s, f1tax_csi, f2tax_csi, f3tax_cs, f5tax_cs, t0imp, f4tax_trad, f4tax_ntrad, f1oct</i>	
Distribution of investment between industries	
<i>finv3(selected industries)</i>	<i>fixed investment/capital ratios</i>
<i>finv2(the rest)</i>	investment follows aggregate investment
Number of households and their consumption preferences are exogenous	
<i>qh</i>	number of households
<i>a3_s</i>	household tastes
Nominaire assumption	
<i>phi</i>	nominal exchange rate
Additional exogenous variables (SAM extension)	
Tax rates (shifter) on household and enterprises	
<i>fentax, f_inctaxrate, f_inctaxrate_h</i>	income tax
Government Investment	
<i>S2gov</i>	government investment share

Another possible closure, often used for small theoretical models in textbooks of international trade, is the *trade-theoretic* closure. This is almost the same as the long-run closure just described except that aggregate capital *x1cap_i* is fixed instead of the variable *capslack*. Rates of return then move in unison to accommodate the *x1cap_i* constraint. In this closure, capital and labour are both fixed in aggregate but mobile between industries.

It should be emphasized that, particularly in a more complex model, many closures might be used for different purposes. There is not a unique natural or correct closure. On the other hand, every sensible closure must satisfy certain requirements. For example, price variables in the model's equations always appear as price ratios; it is a feature of most neo-classical models that only relative prices matter. Thus, if we have a solution of the model, we can always create another solution just by increasing all local prices by, say, 10%. Therefore, for the overall price level to be determined, there has to be at least one exogenous variable measured in local currency units. This variable—there is normally just one—is called the *numeraire*. In Tables 3 and 4 the exchange rate, *phi*, is the numeraire. Another popular choice is *p3tot*, the CPI.

For similar reasons, there is no mechanism in PHILGEM which would determine the total size of the economy. Some quantity variables must be exogenous: these often include primary factor endowments and final demand aggregates.

In summary, PHILGEM lacks theories which explain:

- *Changes in the absolute price level.* We have to exogenously set either the exchange rate or some domestic price index. This setting determines, in an arbitrary way, whether changes in the real exchange

rate are manifested either as changes in domestic prices or as changes in the exchange rate. However, results for real variables are unaffected by this decision.

- *Labour supply.* We have to exogenously set either the average wage level (real or nominal) or employment. This setting determines whether a more buoyant labour market is manifested either as higher wages or as greater employment.
- *Changes in the size and composition of absorption.* Either we must set these exogenously or else the trade balance must be fixed while one or more components of absorption adjust. This setting determines whether increased national income appears either as increased absorption, or alternatively as an improved balance of trade.

As well as suggesting a closure, Table 2 allows us to estimate the size of our model. With its 25 commodities, 24 industries, 2 sources, 2 margin goods, 2 occupations 20 households, the standard version of PHILGEM has more than 25,000 scalar variable elements and 17,000 scalar equations. A typical larger version of PHILGEM used for policy analysis might have at least 100 commodities and industries, 2 sources, 4 margin goods, 8 occupations and at least 340 households leading to more than 1 million variables and 0.75 million equations. In its raw form, such a model would be far too big to solve. The next section explains how GEMPACK can condense a model to manageable size.

7. Using GEMPACK to Solve the Model

Figure 14 and Figure 15 summarize the main stages in the GEMPACK process. The TABLO program converts the TAB file into a FORTRAN source file, PHILGEM.FOR, which contains the model-specific code needed for a solution program.

The Fortran compiler combines PHILGEM.FOR with other, general-purpose, code to produce the executable program PHILGEM.EXE, which can be used to solve the model specified by the user in the TAB file.

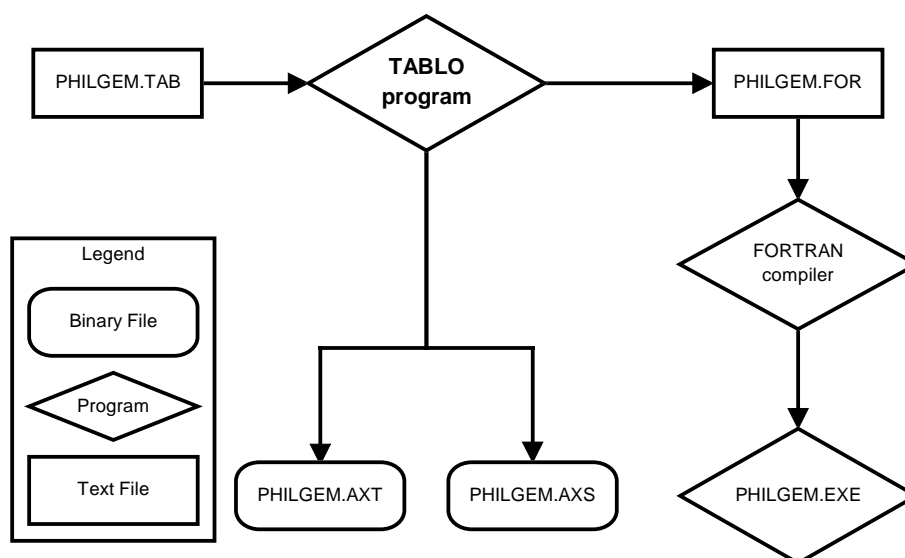


Figure 14. Building a model-specific EXE file

Simulations are conducted using PHILGEM.EXE. Its inputs are:

- a data file, containing input-output data and behavioural parameters. This data file contains all necessary information about the initial equilibrium.
- user input from a text (CMF) file, which specifies:
 - (a) which variables are to be exogenous;
 - (b) shocks to some exogenous variables;
 - (c) into how many steps the computation is divided; and
 - (d) the names of input and output files, and other details of the solution process.

Each simulation produces an SL4 (Solution) file. The SL4 file has a binary format: it may be viewed with the non-model-specific Windows program ViewSOL.

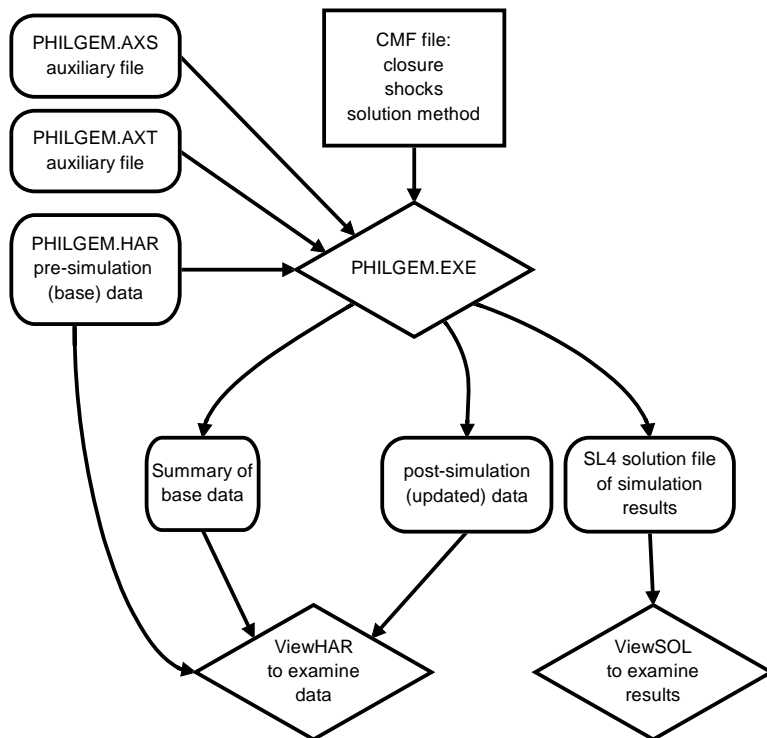


Figure 15. Using the model-specific EXE to run a simulation

For small databases, the processes depicted in Figure 15 do not require a GEMPACK licence. The computer files necessary to run PHILGEM with its standard 25-sector database are freely downloadable from the Internet. Modellers who wish to use larger databases with the standard model, or who wish to use PHILGEM as a starting point to build their own general equilibrium model, would require a GEMPACK licence.

Figure 16 shows a variation on the processes depicted in Figure 14 and Figure 15. This time, TABLO has produced a GSS file. Unlike the FOR file of Figure 8, the GSS file need not be compiled: it is interpreted directly by the standard program GEMSIM. The advantages of this approach is that no FORTRAN compiler is needed and that the required GEMPACK licence is cheaper. The disadvantage is that larger models may solve only slowly, or may altogether exceed size limits built in to GEMSIM. Both methods give the same numerical results.

The programs and steps depicted Figure 14 to Figure 16 can be (and still are) run in the traditional manner by typing into a command prompt. In recent years various Windows "shell" programs have become available, which give model users "point-and-click" control of these processes. The oldest, WinGEM, manages the whole process: from creating or modifying a model, to running simulations. Other programs are more specialized: TABmate is used to create and edit models; while RunGEM is concerned only with running simulations. Another program, AnalyseGE, assists with analysing simulation results.

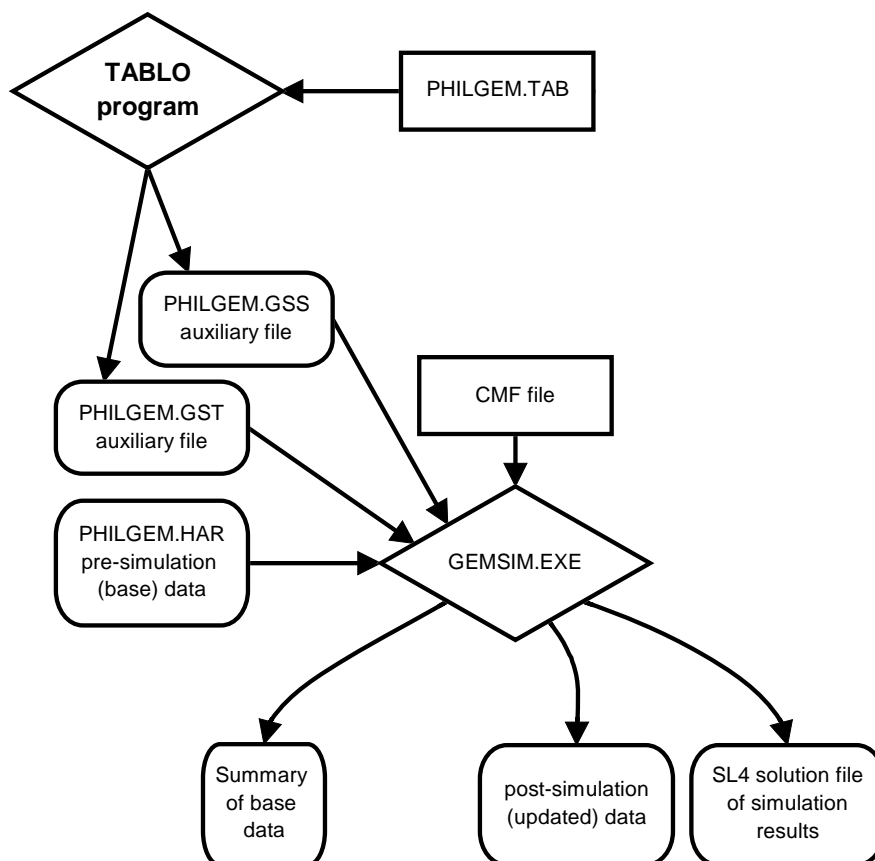


Figure 16. The GEMSIM alternative

8. Conclusion

This document is designed to extend the accessibility of ORANI-style models within the international community. It provides comprehensive documentation of a SAM-based CGE model of the Philippines (PHILGEM). PHILGEM offers a good starting point for model development especially for researchers wanting to extend their ORANI-G inspired models to accommodate: (a) multiple households, and (b) additional equations to facilitate use of data coming from a social accounting matrix (SAM). The extension is straightforward as it only entails adding a few equations at the bottom of the original ORANI-G model specification. PHILGEM can be used for comparative-static policy analysis of the Philippine economy as well as a starting point for developing similar models for other countries.

A distinctive feature of the document is that its account of the theoretical structure and data of the model is framed around the precise representation which is required as input to the GEMPACK computer system. This tight integration of economic and computational aspects of the modelling is intended to allow readers to acquire a hands-on familiarity with the model. The companion files (see Appendix B) contain enough of the GEMPACK system to allow readers to conduct their own simulations with PHILGEM.

References

- Black Book - see Dixon, Parmenter, Powell & Wilcoxon (1992).
- Green Book (DPSV) - see Dixon, Parmenter, Sutton & Vincent (1982).
- Yellow Book - see Dixon, Bowles & Kendrick (1980).
- Orange Book - see Dixon, Powell & Parmenter (1979).
- Armington PS (1969) The Geographic Pattern of Trade and the Effects of Price Changes, *IMF Staff Papers*, XVI, July, pp 176-199.
- Armington PS (1970) Adjustment of Trade Balances: Some Experiments with a Model of Trade Among Many Countries, *IMF Staff Papers*, XVII, November, pp 488-523.
- Cooper R, McLaren K and Powell A (1985) Macroeconomic Closure in Applied General Equilibrium Modelling: Experience from ORANI and Agenda for Further Research, in J. Piggot and J. Whalley (eds), *New Developments in Applied General Equilibrium Analysis*, New York: Cambridge University Press.
- Dixon PB, Bowles S and Kendrick D (1980) *Notes and Problems in Microeconomic Theory*, Amsterdam: North-Holland.
- Dixon PB, Parmenter BR, Powell AA and Wilcoxon PJ (1992) *Notes and Problems in Applied General Equilibrium Economics*, Amsterdam: North-Holland.
- Dixon PB, Parmenter BR, Ryland GJ and Sutton JM (1977) ORANI, A General Equilibrium Model of the Australian Economy: Current Specification and Illustrations of Use for Policy Analysis—First Progress Report of the IMPACT Project, Vol 2, Canberra: Australian Government Publishing Service.
- Dixon PB, Powell A and Parmenter BR (1979) *Structural Adaptation in an Ailing Macro-economy*, Melbourne University Press.
- Dixon PB, Parmenter BR, Sutton JM and Vincent DP (1982) *ORANI: A Multisectoral Model of the Australian Economy*, Amsterdam: North-Holland, hereafter *DPSV*.
- Dixon, P.B., B.R. Parmenter and R.J. Rimmer, (1986), "ORANI Projections of the Short-run Effects of a 50 Per Cent Across-the-Board Cut in Protection Using Alternative Data Bases", pp. 33-60 in J. Whalley and T.N. Srinivasan (eds), *General Equilibrium Trade Policy Modelling*, MIT Press, Cambridge, Mass.
- Harrison, WJ. and Pearson KR (1994) Computing Solutions for Large General Equilibrium Models Using GEMPACK, *IMPACT Preliminary Working Paper No IP-64*, June, 55pp (also in *Computational Economics*, 1996).
- Horridge, J.M (2000) ORANI-G: A General Equilibrium Model of the Australian Economy. Working Paper OP-93, Centre of Policy Studies (CoPS), Monash University. <http://www.monash.edu.au/policy/ftp/workpapr/op-93.pdf>
- Horridge, J.M (2003) ORANI-G: A Generic Single-Country Computable General Equilibrium Model, Centre of Policy Studies (CoPS), Monash University. <http://www.monash.edu.au/policy/oranig.htm>
- Hertel TW, Horridge JM and Pearson KR (1992) Mending the Family Tree, *Economic Modelling*, October, pp 385-407
- Johansen L (1960) *A Multisectoral Model of Economic Growth*, Amsterdam: North-Holland, (2nd edition 1974).
- NSCB (2006) Linked National Accounts series. National Statistical Coordination Board, Makati, Philippines.
- NSO (2005) 2003 Family Income and Expenditure Survey. National Statistical Office, Manila, Philippines.
- Parmenter BR and Meagher GA (1985) Policy Analysis Using a Computable General Equilibrium Model: a Review of Experience at the IMPACT Project, *Australian Economic Review*, No. 1'85, pp.3-15.
- Powell A (1977) *The IMPACT Project: an Overview—First Progress Report of the IMPACT Project*, Vol 1, Canberra: Australian Government Publishing Service.
- Powell A (1991) A Brief Account of Activities 1988 to 1990, *IMPACT Project Report No. R-08*, February, 50pp.
- Powell A and Lawson A (1989) A decade of applied general equilibrium modelling for policy work, pp. 241-290 in Bergman L, Jorgenson D and Zalai E (eds), *General Equilibrium Modeling and Economic Policy Analysis*, New York: Blackwell
- Round J I (2003) Constructing SAMs for Development Policy Analysis: Lessons Learned and Challenges Ahead, *Economic Systems Research*, Vol. 15, No.2, pp. 161-183.
- Vincent DP (1989) Applied General Equilibrium Modelling in the Australian Industries Assistance Commission: Perspectives of a Policy Analyst, pp. 291-350 in Bergman L, Jorgenson D and Zalai E (eds), *General Equilibrium Modeling and Economic Policy Analysis*, New York: Blackwell

Appendix A: Percentage-Change Equations of a CES Nest

Problem: Choose inputs X_i ($i = 1$ to N), to minimise the cost $\sum_i P_i X_i$ of producing given output Z , subject to the CES production function:

$$Z = \left(\sum_i \delta_i X_i^{-\rho} \right)^{-1/\rho}. \quad (\text{A1})$$

The associated first order conditions are:

$$P_k = \Lambda \frac{\partial Z}{\partial X_k} = \Lambda \delta_k X_k^{-(1+\rho)} \left(\sum_i \delta_i X_i^{-\rho} \right)^{-(1+\rho)/\rho}. \quad (\text{A2})$$

$$\text{Hence } \frac{P_k}{P_i} = \frac{\delta_k}{\delta_i} \left(\frac{X_i}{X_k} \right)^{1+\rho}, \quad (\text{A3})$$

$$\text{or } X_i^{-\rho} = \left(\frac{\delta_i P_k}{\delta_k P_i} \right)^{-\rho/(\rho+1)} X_k^{-\rho}. \quad (\text{A4})$$

Substituting the above expression back into the production function we obtain:

$$Z = X_k \left(\sum_i \delta_i \left[\frac{\delta_k P_i}{\delta_i P_k} \right]^{\rho/(\rho+1)} \right)^{-1/\rho}. \quad (\text{A5})$$

This gives the input demand functions:

$$X_k = Z \left(\sum_i \delta_i \left[\frac{\delta_k P_i}{\delta_i P_k} \right]^{\rho/(\rho+1)} \right)^{1/\rho}, \quad (\text{A6})$$

$$\text{or } X_k = Z \delta_k^{1/(\rho+1)} \left[\frac{P_k}{P_{\text{ave}}} \right]^{-1/(\rho+1)}, \quad (\text{A7})$$

$$\text{where } P_{\text{ave}} = \left(\sum_i \delta_i^{1/(\rho+1)} P_i^{\rho/(\rho+1)} \right)^{(\rho+1)/\rho}. \quad (\text{A8})$$

Transforming to percentage changes (see Appendix E) we get:

$$x_k = z - \sigma (p_k - p_{\text{ave}}), \quad (\text{A9})$$

$$\text{and } p_{\text{ave}} = \sum_i S_i p_i, \quad (\text{A10})$$

$$\text{where } \sigma = \frac{1}{\rho+1} \text{ and } S_i = \delta_i^{1/(\rho+1)} P_i^{\rho/(\rho+1)} / \sum_k \delta_k^{1/(\rho+1)} P_k^{\rho/(\rho+1)}. \quad (\text{A11})$$

Multiplying both sides of (A7) by P_k we get:

$$P_k X_k = Z \delta_k^{1/(\rho+1)} P_k^{\rho/(\rho+1)} P_{\text{ave}}^{1/(\rho+1)}. \quad (\text{A12})$$

$$\text{Hence } \frac{P_k X_k}{\sum_i P_i X_i} = \delta_k^{1/(\rho+1)} P_k^{\rho/(\rho+1)} / \sum_i \delta_i^{1/(\rho+1)} P_i^{\rho/(\rho+1)} = S_k, \quad (\text{A13})$$

i.e., the S_i of (A11) turn out to be cost shares.

Technical Change Terms

With technical change terms, we must choose inputs X_i so as to:

$$\text{minimise } \sum_i P_i X_i \text{ subject to: } Z = \left(\sum_i \delta_i \left[\frac{X_i}{A_i} \right]^{-\rho} \right)^{-1/\rho}. \quad (\text{A14})$$

$$\text{Setting } \tilde{X}_i = \frac{X_i}{A_i} \text{ and } \tilde{P}_i = P_i A_i \text{ we get:} \quad (\text{A15})$$

$$\text{minimise } \sum_i \tilde{P}_i \tilde{X}_i \text{ subject to: } Z = \left(\sum_i \delta_i \tilde{X}_i^{-\rho} \right)^{-1/\rho}, \quad (\text{A16})$$

which has the same form as problem (A1). Hence the percentage-change form of the demand equations is:

$$\tilde{x}_k = z - \sigma(\tilde{p}_k - \tilde{p}_{ave}), \quad (\text{A17})$$

$$\text{and } \tilde{p}_{ave} = \sum_i S_i \tilde{p}_i. \quad (\text{A18})$$

But from (A15), $\tilde{x}_k = x_k - a_k$, and $\tilde{p}_i = p_i + a_i$, giving:

$$x_k - a_k = z - \sigma(p_k + a_k - \tilde{p}_{ave}). \quad (\text{A19})$$

$$\text{and } \tilde{p}_{ave} = \sum_i S_i (p_i + a_i). \quad (\text{A20})$$

When technical change terms are included, we call \tilde{x}_k , \tilde{p}_k and \tilde{p}_{ave} *effective* indices of input quantities and prices.

The following two sub-topics are more advanced, and could be omitted at the first reading.

Two Input CES: Reverse Shares

Where a CES nest has only two inputs we can write (A19) and (A20) in a way which speeds up computation. Suppose we have domestic and imported inputs, with suffixes d and m. (A19) becomes:

$$x_d - a_d = z - \sigma(p_d + a_d - S_d(p_d + a_d) - S_m(p_m + a_m)),$$

$$\text{and } x_m - a_m = z - \sigma(p_m + a_m - S_d(p_d + a_d) - S_m(p_m + a_m)). \quad (\text{A21})$$

Simplifying, we get:

$$x_d - a_d = z - \sigma S_m((p_d + a_d) - (p_m + a_m)),$$

$$\text{and } x_m - a_m = z + \sigma S_d((p_d + a_d) - (p_m + a_m)). \quad (\text{A22})$$

In order for TABLO to substitute out x, we must express (A22) as a single vector equation:

$$x_k - a_k = z - \sigma R_k((p_d + a_d) - (p_m + a_m)), \quad k = d, m \quad (\text{A23})$$

The R_k are *reverse shares*, defined by:

$$R_d = S_m \quad \text{and} \quad R_m = R_d - 1 = S_m - 1 = -S_d \quad \text{note that } R_d - R_m = 1 \quad (\text{A24})$$

(A20) becomes:

$$\tilde{p}_{ave} = \sum_i S_i (p_i + a_i) = R_d(p_m + a_m) - R_m(p_d + a_d). \quad (\text{A25})$$

Twist for Two Input CES

A twist is a combination of small technical changes which, taken together, are locally cost neutral. For example, we might ask, what values for a_d and a_m would, in the absence of price changes, cause the ratio $(x_d - x_m)$ to increase by $t\%$ without affecting \tilde{p}_{ave} ? That is, find a_d and a_m such that:

$$S_d a_d + S_m a_m = 0, \quad \text{using (A20), and} \tag{A26}$$

$$x_d - x_m = (1-\sigma)(a_d - a_m) = t, \quad \text{using (A21);} \tag{A27}$$

giving

$$a_d = S_m t / (1-\sigma) \quad \text{and} \quad a_m = -S_d t / (1-\sigma). \tag{A28}$$

Adopting *reverse share* notation:

$$a_k = R_k t / (1-\sigma) \quad k = d, m \tag{A29}$$

Substituting (A29) back into (A23) we get:

$$x_k = z + R_k t / (1-\sigma) - \sigma R_k (p_d - p_m + R_d t / (1-\sigma) - R_m t / (1-\sigma)) \quad k = d, m$$

so $x_k = z + R_k t - \sigma R_k (p_d - p_m) \quad k = d, m$

allowing us to rewrite (A19) and (A20) as:

$$x_k = z + R_k (t - \sigma(p_d - p_m)) \quad k = d, m \tag{A30}$$

and $\tilde{p}_{ave} = R_d p_m - R_m p_d. \tag{A31}$

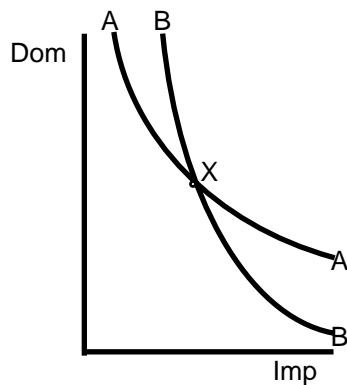


Figure A1

Twist variables, such as t , are heavily used in MONASH where they are used to simulate secular (i.e., not price-induced) trends in import shares. Figure A1 shows how 'twist' variables get their name. AA is an isoquant showing what quantities of domestic and imported goods can be combined to give the same utility. The chosen combination is at X. Technical changes a_d and a_m translate AA both down and to the right, in such a way that BB still passes through X. It is as if AA had been twisted or pivoted around X.

A small change concept of cost neutrality is used to develop the notion of twist variables. Where budget shares change by a large amount, the same technical change cannot be cost-neutral at both initial and final input proportions, although it will usually be cost-neutral at some intermediate proportion. Thus, there are no levels formulae corresponding to (A26).

Appendix B: PHILGEM on the Web

All the computer files mentioned in this document can be downloaded from a PHILGEM section on the web-page::

<http://www.monash.edu.au/policy/oranig.htm>

That page also contains supplementary material used for preparing or aggregating data, plus information about some of the adaptations of ORANI-G that have been made for different countries.

Appendix C: Hardware and Software Requirements for Using GEMPACK

The essential parts of GEMPACK can be used on most desktop and mainframe computers: wherever an ANSI-standard FORTRAN is available. A customized installation kit is available for Windows PCs.

Additional visual programs (eg, ViewHAR) are available that require Microsoft Windows XP or later. Details of GEMPACK products and pricing can be seen at:

<http://www.monash.edu.au/policy/gempack.htm>

Appendix D: Main differences between full-size PHILGEM and the version described here

The commodity, industry, occupational, and household classifications of PHILGEM described here are aggregates of the classifications used in the original version of PHILGEM, which had 240 industries and commodities, 10 labour occupations, and 42,092 households

Appendix E: Deriving Percentage-Change Forms

Using first principles, a levels equation, for example,

$$Y = X^2 + Z,$$

is turned into percentage-change form by first taking total differentials:

$$dY = 2XdX + dZ.$$

Percentage changes x , y , and z are defined *via*:

$$y = 100 \frac{dY}{Y} \quad \text{or} \quad dY = \frac{Yy}{100}, \quad \text{similarly} \quad dX = \frac{Xx}{100} \quad \text{and} \quad dZ = \frac{Zz}{100}.$$

Thus our sample equation becomes:

$$\frac{Yy}{100} = 2X \frac{Xx}{100} + \frac{Zz}{100}, \quad \text{or} \quad Yy = 2X^2x + Zz.$$

In practice such formal derivations are often unnecessary. Most percentage-change equations follow standard patterns which the modeller soon recognizes. Some of these are shown in Table E1.

Column (2) in Table E1 corresponds closely to the total differential form of column (1), and may be thought of as a step on the way to column (3). Alternatively, we may use column (2) directly, either because it is simpler, or to avoid computing shares.

The 2nd alternate form in column (3) for example 10 shows how ordinary and percentage changes may be mixed. It is based on the identity $Yy \equiv 100\Delta Y$. See also example 5.

Variables can only be added or subtracted (as in examples 9 and 12) where they share the same units. In adding quantities, we can normally identify a common price (often the basic price). By multiplying through additive expressions by a common price, we can express the coefficients of percentage-change equations as functions of flows, rather than quantities, so obviating the need to define physical units (compare examples 9 and 11).

Table E1 Examples of Percentage-Change Forms

Example	(1) Original or Levels Form	(2) Intermediate Form	(3) Percentage-Change Form
1	$Y = 4$	$Yy = 4*0$	$y = 0$
2	$Y = X$	$Yy = Xx$	$y = x$
3	$Y = 3X$	$Yy = 3Xx$	$y = x$
4	$Y = XZ$	$Yy = XZx + XZz$	$y = x + z$
5	$Y = X/Z$	$Yy = (X/Z)x - (X/Z)z$	$y = x - z$ or $100(Z)\Delta Y = Xx - Xz$
6	$X_1 = M/4P_1$	$X_1x_1 = (M/4P_1)m - (M/4P_1)p_1$	$x_1 = m - p_1$
7	$Y = X^3$	$Yy = X^3 3x$	$y = 3x$
8	$Y = X^\alpha$	$Yy = X^\alpha \alpha x$	$y = \alpha x$ (α assumed constant)
9	$Y = X + Z$	$Yy = Xx + Zz$	$y = S_x x + S_z z$ where $S_x = X/Y$, etc
10	$Y = X - Z$	$Yy = Xx - Zz$	$y = S_x x - S_z z$ or $100(\Delta Y) = Xx - Zz$
11	$PY = PX + PZ$	$PY(y+p) = PX(x+p) + PZ(z+p)$ or $PYy = PXx + PZz$	$y = S_x x + S_z z$ where $S_x = PX/PY$, etc
12	$Z = \sum X_i$	$Zz = \sum X_i x_i$ or $0 = \sum X_i (x_i - z)$	$z = \sum S_i x_i$ where $S_i = X_i/Z$
13	$XP = \sum X_i P_i$	$XP(x+p) = \sum X_i P_i (x_i + p_i)$	$x+p = \sum S_i (x_i + p_i)$ where $S_i = X_i P_i / XP$

Appendix F: Algebra for the Linear Expenditure System

The purpose of this appendix is to expand on some of the algebra underlying Excerpt 16 of the TABLO input file.

First note that, the utility function (22) in the text can be written:

$$\text{Utility per household} = \prod_c \left\{ \frac{X3_S(c)}{Q} - A3SUB(c) \right\}^{S3LUX(c)}, \tag{F1}$$

using equation (27). Here, $X3_S(c)/Q$ is the average consumption of each composite good c , and $A3SUB(c)$ is a parameter. The household's problem is to choose $X3_S(c)/Q$ to maximise utility subject to the constraint:

$$\sum_c X3_S(c)/Q * P3_S(c) = V3TOT/Q, \tag{F2}$$

The associated additional first order conditions are:

$$\begin{aligned} \Lambda P3_S(c) &= \frac{\partial U}{\partial (X3_S(c)/Q)} \\ &= S3LUX(c) \cdot U \cdot \left\{ \frac{X3_S(c)}{Q} - A3SUB(c) \right\}^{-1} \end{aligned} \tag{F3}$$

Manipulation (and use of equation (27)) yields:

$$P3_S(c)\{X3_S(c) - X3SUB(c)\} = S3LUX(c).Q.U/\Lambda. \quad (F4)$$

or
$$P3_S(c).X3_S(c) = P3_S(c).X3SUB(c) + S3LUX(c).Q.U/\Lambda. \quad (F5)$$

(F5) is really the same as equations (25) to (27) in the text. The key to the simplicity of the equations there is that no attempt is made to eliminate the Lagrange multiplier term, $Q.U/\Lambda$. Instead, the constraint (F2) is written down as part of the equation system: see (28). This implicit approach often yields dividends. Here in the appendix we press forward more conventionally to express demands directly as a function of prices and income. Summing over c and using (F2), we see that

or
$$Q.U/\Lambda = V3TOT - \sum_c P3_S(c).X3SUB(c) \quad (F6)$$

so that $Q.U/\Lambda$ is identified as the $V3LUX_C$ of equation (24) in the text. Equation (23) then follows from (F4) above. Combining (F4) with (F6) we get the linear expenditure system (LES):

$$P3_S(c).X3_S(c) = P3_S(c).X3SUB(c) + S3LUX(c).\{V3TOT - \sum_k X3SUB(k)*P3_S(k)\}. \quad (F7)$$

To find the expenditure elasticities, we convert to percentage change form, ignoring all changes in prices and tastes:

$$x3_s(c) = - FRISCH.B3LUX(c). w3tot \quad (F8)$$

where FRISCH is defined, by tradition, as $-\frac{V3TOT}{V3LUX_C}$,

and the $B3LUX(c)$ are the shares of 'luxury' in total expenditure on good c ,

i.e.
$$B3LUX(c) = \frac{X3_S(c) - X3SUB(c)}{X3_S(c)}.$$

Thus the expenditure elasticities are given by:

$$EPS(c) = - FRISCH.B3LUX(c) \quad (F9)$$

In the TABLO program, (F9) is reversed to derive $B3LUX(c)$ from $EPS(c)$ and FRISCH.

Taste Change Terms

Often we wish to simulate the effect of a switch in consumer spending, induced by a taste change. This could be brought about by a shock *either* to the $a3lux(c)$ (marginal budget shares) *or* to the $a3sub(c)$ (subsistence quantities). Two problems arise. First, what combination of $a3sub$ and $a3lux$ shocks is best? Second, the $a3lux$ shocks must obey the rule that marginal budget shares add to 1. To tie down the relation between the $a3lux$ and the $a3sub$, we will assume that they move in proportion:

$$a3lux(c) = a3sub(c) - \lambda, \quad (F12)$$

and that the constant of proportionality λ is given by the adding-up requirement:

$$\sum_k S3LUX(k).a3lux(k) = 0 \quad (F13)$$

implying that:

$$a3lux(c) = a3sub(c) - \sum_k S3LUX(k).a3sub(k), \quad E_a3lux$$

We also suppose that

$$\sum_k S3_S(k).a3sub(k) = 0 \quad (F14)$$

This is guaranteed by equation E_a3sub :

$$a3sub(c) = a3_s(c) - \sum_k S3_S(k).a3_s(k) \quad E_a3sub$$

The effect of these assumptions is to allow budget shares to be shocked whilst altering expenditure elasticities and the Frisch parameter as little as possible.

Appendix G: Making your own AGE model from PHILGEM

In conjunction with the full GEMPACK system, the PHILGEM.TAB file (which appears in Excerpts listed previously) forms an excellent starting-point for constructing your own AGE model. This file is available from the World-Wide Web (see Appendix B). The best plan is to start from something that works (PHILGEM) and modify it in small steps until it suits your needs.

The most minimal change is to attach a different data file, appropriate to another country. Use either of the GEMPACK programs ViewHAR or MODHAR to turn data in text files into GEMPACK's binary format. Table G1 lists the data matrices that will be needed.

Table G1 Contents of PHILGEM Data File

Header	Dimension	Symbol	Name
COM	25 length 12	COM	Set COM Goods
IND	24 length 12	IND	Set IND Industries
GEND	2 length 12	GEND	Set GEND Gender classification
OCC	2 length 8	OCC	Set OCC Occupations
MAR	3 length 12	MAR	Set MAR Margin Commodities
HOU	20 length 12	HOU	Set HOU Households
1BAS	COM*SRC*IND	V1BAS	Intermediate Basic
2BAS	COM*SRC*IND	V2BAS	Investment Basic
3BAS	COM*SRC	V3BAS	Households Basic
4BAS	COM	V4BAS	Exports Basic
5BAS	COM*SRC	V5BAS	Government Basic
6BAS	COM*SRC	V6BAS	Stocks
1MAR	COM*SRC*IND*MAR	V1MAR	Intermediate Margins
2MAR	COM*SRC*IND*MAR	V2MAR	Investment Margins
3MAR	COM*SRC*MAR	V3MAR	Households Margins
4MAR	COM*MAR	V4MAR	Exports Margins
5MAR	COM*SRC*MAR	V5MAR	Government Margins
1TAX	COM*SRC*IND	V1TAX	Intermediate Tax
2TAX	COM*SRC*IND	V2TAX	Investment Tax
3TAX	COM*SRC	V3TAX	Households Tax
4TAX	COM	V4TAX	Exports Tax
5TAX	COM*SRC	V5TAX	Government Tax
1CAP	IND	V1CAP	Capital
PERS	IND*OCC*HOU	PERSON	Persons by iNDxOCC
1LAB	IND*OCC	V1LAB	Labour
1LND	IND	V1LND	Land
1OCT	IND	V1OCT	Other Costs
1PTX	IND	V1PTX	Prod tax or subsidies
0TAR	COM	V0TAR	Tariff Revenue
SLAB	IND	SIGMA1LAB	Labour Sigma
P028	IND	SIGMA1PRIM	Primary Factor Sigma
1ARM	COM	SIGMA1	Intermediate Armington
2ARM	COM	SIGMA2	Investment Armington
3ARM	COM	SIGMA3	Households Armington
P018	COM	EXP_ELAST	Traditnl Export Elast.
LINC	IND*OCC*HOU	V1LABINC	Labour income of household h

.../continued

Table G1 Contents of PHILGEM Data File (continued)

Header	Dimension	Symbol	Name
MAKE	COM*IND	MAKE	Multiproduct Matrix
SCET	IND	SIGMA1OUT	Output Sigma
P21H	HOU	FRISCH	Hh Frisch parameters
XPEL	COM*HOU	EPS	Expenditure elasticities
3PUR	COM*HOU	PUR	Household purchases
ITEX	COM	IsIndivExp	Individual export elast
EXNT	1	EXP_ELAST_NT	Non-Trad. Export Elast.
P027	IND	QCOEF	Gross/Net Rate of Return
CPTR	COM	TARCHANGE	Change in the power of tariff: For Simulation
VHGS	HOU	VHOUGOS	HHs income from GOS
VHET	HOU	VHOUEENT	Hhs income/dividend from Enterprises
VHGV	HOU	VHOUGOV	Transfers by gov to hhs
VHRW	HOU	VHOUROW	ROW payments to HHs
VGVB	HOU	VGOVHOU	Transfers by hhs to gov
VTXH	HOU	VTAXHOU	Income tax payments by hhs to gov
VETH	HOU	VENTHOU	Household transfers to enterprises
VRWH	HOU	VROWHOU	HH payments to ROW
GVSH	IND	GOVSHRINV	Government shares of investment
VGGS	1	VGOVGOS	GOS inc to gov
VTXE	1	VTAXENT	Corporations tax
VGVE	1	VGOVENT	Enterprise transfers to gov
VGVR	1	VGOVROW	ROW transfers to gov
VRGV	1	VROWGOV	Gov transfers to ROW
VGSG	1	VGOSGOV	Interest on public debt
VEGV	1	VENTGOV	Interest on public debt
VEGS	1	VENTGOS	GOS payments to enterprises
VRGS	1	VROWGOS	GOS payments to ROW
VRWE	1	VROWENT	ROW payments to enterprises
VGSR	1	VGOSROW	GOS from ROW
VERW	1	VENTROW	ROW receipts from enterprises
NVNT	COM	STOKRAT	Stock/Sale ratios
VHHO	HOU*HOU	VHOUHOU	Intra-hh transfers

The dimensions of the model (e.g., the number of sectors) are controlled by the 'Set' data at the beginning of the list above.

Your own national input-output tables may not support the ORANI-G level of detail. For example, you may be unable to distinguish land as a third primary factor. Rather than deleting all mention of land from the equations, achieve the same effect by supplying a data vector of land rentals (header 'ILND' in Table G1) which is filled with tiny non-zero numbers. Again, if you cannot gather margins data, you may restrict the set of margins commodities (in Excerpt 1) to a single commodity, and fill all margins matrices with tiny numbers or zeroes. Define one occupational labour category only, if need be. A diagonal MAKE matrix is equivalent to removing multi-production. All these techniques are less error-prone than altering the equations, and leave scope for later collection of more data.

As your confidence grows, you may wish to change or extend the equation system. For example, you could easily alter the consumption function. More ambitiously, you could alter the production nesting system of Figure 5 to allow intermediate usage of energy to be substitutable with primary factors.

Appendix H: Formal Checks on Model Validity; Debugging Strategies

A number of tests should be performed each time a model's equations or data are changed. We set out here the proper procedure to follow.

1. Price homogeneity test

It is a property of neoclassical models that agents respond to changes in relative prices, but not to changes in the absolute level of prices. That is, a uniform increase in *all* prices does not affect any quantity variables. Nearly always there is only one exogenous variable measured in domestic currency units—it is called the numeraire. Typical choices of numeraire include the exchange rate (ϕ) or the consumer price index ($p3tot$). If we shock the numeraire by 1% we would expect to see that all domestic prices and flows increase by 1% while real variables remain unchanged.

The supplied file PHILGEM-NHOM.CMF can be used to perform this test. Check that it refers to the right model and data and that the only shock is a 1% shock to ϕ . Run the simulation and examine the solution file. You should see that all prices move by 1% whilst quantities are unchanged.

2. Check initial data

The price homogeneity simulation should have produced a HAR file that summarizes the initial data. Examine this file (using ViewHAR) and check that the database is balanced, using the headers written by Excerpt 37 of the TAB file (see main text).

3. Real homogeneity test

Neoclassical models normally display constant returns to scale. This means that if all real exogenous variables (not ratios or prices) are shocked by 1%, all endogenous real variables should also move by 1%, leaving prices unchanged.

To test this properly, you have to ensure that any real ordinary change variables are shocked by the right amount and that export demand curves move outwards by an amount corresponding to a 1% increase in the size of the rest of the world. The supplied file PHILGEM-RHOM.CMF simplifies this task. Comment out the numeraire shock used for the price homogeneity test and reinstate the series of real shocks which were originally commented out. Run the simulation and examine the results.

4. Change in GDP should be the same from both sides

Now run a simulation where relative prices change, ie, not a homogeneity test. For example, you could use the closure in PHILGEM-NHOM.CMF and shock real household consumption by 10%. It's best to administer a shock that is large enough to make at least a 1% difference to the majority of variables. Stick with a one-step or Johansen simulation for now. Check that the results for the two nominal GDP variables, $w0gdpxp$ and $w0gdpinc$ are the same up to 5 significant figures. An unbalanced data-base or errors in equations could disturb this equality.

5. Updated data-base should also be balanced.

An updated data file will have been produced by the previous (10% real consumption increase) simulation. Use this updated data as the starting point for a second simulation with the same shock. Again, a HAR file of summary data will be produced, incorporating the effects of the first but not the second simulation. As in Step 2, use this to check whether the updated data is balanced. If not, and if all the previous tests were passed, there is probably something wrong with some Update statement.

6. Repeat above tests using a multistep solution method.

Go through the above steps this time using a 2-4-8 Euler extrapolation solution method. If there is a problem, but there was no problem with the Johansen tests, you have a subtle problem. Possibly a percentage change variable is passing through zero; or maybe you are using formulae to alter the data after it has been read (always a bad idea).

7. Be sure you can explain the results.

It is important to realize that there are many errors that Tests 1 to 6 above will not detect. For example, if the export demand elasticities in the database had the wrong sign, the model would still pass these tests. Only a careful 'eye-balling' of results will uncover this type of error. After modifying the model, it

is a good idea to run a standard experiment for which the results have already been analysed. You should be able to understand how and why the new results differ.

Other tips

Change little, check often

Although the above tests can reveal the existence of a problem, they give little direct indication of its cause. The wise modeller performs the various tests both before and after making each small alteration to the model. Then, if there is a problem, the cause must lie with the minor changes just made.

Duplication of previous results.

Extensions and modifications to model equations should be designed so that you can still duplicate simulation results computed with the previous version of the model. For example, when adding new equations it is often a good idea to include shift variables which can be left endogenous. This prevents the new equations from affecting the rest of the system, and allows a standard closure to be used. To 'switch on' your new equations, exogenize these shift variables while at the same time endogenizing an equal number of other model variables. Sometimes modifications can be switched on and off by suitable parameter settings (an example might be the CET between goods for export and domestic markets in Excerpt 13 of the PHILGEM TAB file).

If you designed your extension in this flexible way, you could test that you are able to duplicate results that you computed before adding new equations. If the new results are different, you may have inadvertently made some other change that was overlooked.

Always check that GDP results are the same from both sides

It should become a reflex action, every time you look at results, to check that w0gdpexp and w0gdpinc are the same. It takes little time, and gives early warning of many possible errors.

Beware of rarely used exogenous variables

Shocks to rarely used exogenous variables often bring errors to light. Since in nearly all simulations these variables have zero change, the values of their coefficients are usually irrelevant. For example, when you shock normally-omitted technical change variables you are using parts of the model which have undergone relatively infrequent testing. So be alert.

To identify the problem, try different closures

Suppose you noticed that some problem occurred in long-run closures where capital stocks are mobile, but did not occur in short-run closures where capital stocks are fixed. That might suggest that some coefficient of x1cap (percentage change in capital stocks) was wrongly computed, or that x1cap had been inadvertently omitted from an equation or update statement.

Appendix I: Short-Run Supply Elasticity

Where capital stocks are fixed, we can derive an approximate expression for a short-run supply schedule as follows. Imagine that output, z, is a CES function of capital and labour, and that other, material, inputs are demanded in proportion to output. Using percentage change form, we may write:

$$p = H_K p_K + H_L p_L + H_M p_M \quad \text{zero pure profits} \quad (J1)$$

$$x_L - x_K = \sigma(p_K - p_L) \quad \text{factor proportions} \quad (J2)$$

$$z = S_L x_L + S_K x_K \quad \text{production function} \quad (J3)$$

where H_K , H_L and H_M are the shares in total costs of capital, labour and materials, and where S_K and S_L are the shares in primary factor costs of capital and labour. p is output price, and p_K , p_L and p_M are the prices of capital, labour and materials respectively. z is output, and x_K and x_L are the input quantities of capital and labour. "Capital" should be interpreted as covering all fixed factors, including land.

In the short run closure we set x_K to zero, so that the last 2 equations become:

$$x_L = \sigma(p_K - p_L) \quad \text{and} \quad z = S_L x_L \quad \text{giving:} \quad (J4)$$

$$z = S_L \sigma(p_K - p_L) \quad \text{or} \quad p_K = p_L + z/(S_L \sigma) \quad (J5)$$

Substituting (J5) into (J1) we get:

$$p = H_K(p_L + z/(S_L\sigma)) + H_L p_L + H_M p_M \quad (J6)$$

$$p = zH_K/(S_L\sigma) + (H_K + H_L)p_L + H_M p_M \quad (J7)$$

$$zH_K/(S_L\sigma) = p - (H_K + H_L)p_L - H_M p_M \quad (J8)$$

$$z = (S_L\sigma/H_K)[p - (H_K + H_L)p_L - H_M p_M] \quad (J9)$$

Call H_F the share of primary factor in total costs ($= H_K + H_L$). Then $H_K = S_K H_F$

$$\text{so } z = (\sigma S_L/S_K)[p/H_F - p_L - (H_M/H_F)p_M] \dots \text{compare DPSV eq. 45.19} \quad (J10)$$

The shortrun supply elasticity is the coefficient on p , namely:

$$\sigma S_L/(S_K H_F) \quad (J11)$$

In other words, supply is more elastic as either the labour/capital ratio is higher, or the share of materials in total cost is higher. (J11) is only a partial equilibrium estimate; it assumes that all inputs except capital are in elastic supply.

For industries which have permanently fixed factors (eg, land or natural resources) we could compute longrun supply elasticities in a similar way. In (J11) S_L should be interpreted as the share of *mobile* factors in primary factor cost, with $S_K = 1 - S_L$.

Appendix J: List of Variables

Symbol	Dimensions	Description
x1(c,s,i)	c∈COM s∈SRC i∈IND	Intermediate basic demands
x2(c,s,i)	c∈COM s∈SRC i∈IND	Investment basic demands
x3(c,s)	c∈COM s∈SRC	Household basic demands
x4(c)	c∈COM	Export basic demands
x5(c,s)	c∈COM s∈SRC	Government basic demands
delx6(c,s)	c∈COM s∈SRC	Inventories demands
p0(c,s)	c∈COM s∈SRC	Basic prices for local users
pe(c)	c∈COM	Basic price of exportables
delV6(c,s)	c∈COM s∈SRC	Value of inventories
x1mar(c,s,i,m)	c∈COM s∈SRC i∈IND m∈MAR	Intermediate margin demand
x2mar(c,s,i,m)	c∈COM s∈SRC i∈IND m∈MAR	Investment margin demands
x3mar(c,s,m)	c∈COM s∈SRC m∈MAR	Household margin demands
x4mar(c,m)	c∈COM m∈MAR	Export margin demands
x5mar(c,s,m)	c∈COM s∈SRC m∈MAR	Government margin demands
p0dom(c)	c∈COM	Basic price of domestic goods = p0(c,"dom")
delV1TAX(c,s,i)	c∈COM s∈SRC i∈IND	Intern tax rev
delV2TAX(c,s,i)	c∈COM s∈SRC i∈IND	Invest tax rev
delV3TAX(c,s)	c∈COM s∈SRC	H'hold tax rev
delV4TAX(c)	c∈COM	Export tax rev
delV5TAX(c,s)	c∈COM s∈SRC	Govmnt tax rev
x1lab(i,o)	i∈IND o∈OCC	Employment by industry and occupation
p1lab(i,o)	i∈IND o∈OCC	Wages by industry and occupation
x1cap(i)	i∈IND	Current capital stock
p1cap(i)	i∈IND	Rental price of capital
x1lnd(i)	i∈IND	Use of land
p1lnd(i)	i∈IND	Rental price of land
delV1PTX(i)	i∈IND	Ordinary change in production tax revenue
x1oct(i)	i∈IND	Demand for "other cost" tickets
p1oct(i)	i∈IND	Price of "other cost" tickets
delV0TAR(c)	c∈COM	Ordinary change in tariff revenue
p1(c,s,i)	c∈COM s∈SRC i∈IND	Purchaser's price, intermediate
p2(c,s,i)	c∈COM s∈SRC i∈IND	Purchaser's price, investment
p3(c,s)	c∈COM s∈SRC	Purchaser's price, household
p4(c)	c∈COM	Purchaser's price, exports,loc\$
p5(c,s)	c∈COM s∈SRC	Purchaser's price, government
p1lab_o(i)	i∈IND	Price to each industry of labour composite
x1lab_o(i)	i∈IND	Effective labour input
p1prim(i)	i∈IND	Effective price of primary factor composite
x1prim(i)	i∈IND	Primary factor composite
a1lab_o(i)	i∈IND	Labor-augmenting technical change
a1cap(i)	i∈IND	Capital-augmenting technical change
a1lnd(i)	i∈IND	Land-augmenting technical change
delV1PRIM(i)	i∈IND	Ordinary change in cost of primary factors

.../continued

Appendix J: List of Variables (continued)

Symbol	Dimensions	Description
a1(c,s,i)	c∈COM s∈SRC i∈IND	Intermediate basic tech change
x1_s(c,i)	c∈COM i∈IND	Intermediate use of imp/dom composite
p1_s(c,i)	c∈COM i∈IND	Price, intermediate imp/dom composite
p1mat(i)	i∈IND	Intermediate cost price index
p1var(i)	i∈IND	Short-run variable cost price index
x1tot(i)	i∈IND	Activity level or value-added
a1prim(i)	i∈IND	All factor augmenting technical change
a1tot(i)	i∈IND	All input augmenting technical change
p1tot(i)	i∈IND	Average input/output price
a1oct(i)	i∈IND	"Other cost" ticket augmenting technical change
a1_s(c,i)	c∈COM i∈IND	Tech change, intermediate imp/dom composite
delV1CST(i)	i∈IND	Change in ex-tax cost of production
delV1TOT(i)	i∈IND	Change in tax-inc cost of production
delPTXRATE(i)	i∈IND	Change in rate of production tax
p1cst(i)	i∈IND	Index of production costs (for AnalyseGE)
q1(c,i)	c∈COM i∈IND	Output by commodity and industry
pq1(c,i)	c∈COM i∈IND	Price of com c produced by ind i
p0com(c)	c∈COM	General output price of locally-produced commodity
x0com(c)	c∈COM	Output of commodities
x0dom(c)	c∈COM	Output of commodities for local market
x2_s(c,i)	c∈COM i∈IND	Investment use of imp/dom composite
p2_s(c,i)	c∈COM i∈IND	Price, investment imp/dom composite
a2(c,s,i)	c∈COM s∈SRC i∈IND	Investment basic tech change
a2tot(i)	i∈IND	Neutral technical change - investment
p2tot(i)	i∈IND	Cost of unit of capital
x2tot(i)	i∈IND	Investment by using industry
a2_s(c,i)	c∈COM i∈IND	Tech change, investment imp/dom composite
a3(c,s)	c∈COM s∈SRC	Household basic taste change
x3_s(c)	c∈COM	Household use of imp/dom composite
p3_s(c)	c∈COM	Price, household imp/dom composite
p3toth(h)	h∈HOU	Consumer price index
x3toth(h)	h∈HOU	Real household consumption
w3toth(h)	h∈HOU	Nominal total household consumption
w3luxh(h)	h∈HOU	Nominal luxury consumption
qh(h)	h∈HOU	Number of households
utilityh(h)	h∈HOU	Utility per household
x3lux(c,h)	c∈COM h∈HOU	Household - supernumerary demands
x3sub(c,h)	c∈COM h∈HOU	Household - subsistence demands
a3lux(c,h)	c∈COM h∈HOU	Taste change, supernumerary demands
a3sub(c,h)	c∈COM h∈HOU	Taste change, subsistence demands
a3_s(c,h)	c∈COM h∈HOU	Taste change, household imp/dom composite
x3_sh(c,h)	c∈COM h∈HOU	Consumption, household imp/dom composite
p3tot		Consumer price index
x3tot		Real household consumption
w3tot		Nominal total household consumption
pf4(c)	c∈COM	Foreign currency export prices
f4p(c)	c∈COM	Price (upward) shift in export demand schedule
f4q(c)	c∈COM	Quantity (right) shift in export demands

.../continued

Appendix J: List of Variables (continued)

Symbol	Dimensions	Description
phi		Exchange rate, local currency/\$world
x4_ntrad		Quantity, collective export composite
f4p_ntrad		Uniform upward (price) demand shift for collective expts
f4q_ntrad		Uniform right (quantity) demand shift for collective exports
p4_ntrad		Average price of collective exports
f5tot		Overall shift term for government demands
f5tot2		Ratio between f5tot and x3tot
f5(c,s)	c∈COM s∈SRC	Government demand shift
fx6(c,s)	c∈COM s∈SRC	Shifter on rule for stocks
a1mar(c,s,i,m)	c∈COM s∈SRC i∈IND m∈MAR	Intermediate margin tech change
a2mar(c,s,i,m)	c∈COM s∈SRC i∈IND m∈MAR	Investment margin tech change
a3mar(c,s,m)	c∈COM s∈SRC m∈MAR	Household margin tech change
a4mar(c,m)	c∈COM m∈MAR	Export margin tech change
a5mar(c,s,m)	c∈COM s∈SRC m∈MAR	Government margin tech change
delSale(c,s,d)	c∈COM s∈SRC d∈DEST	Sales aggregates
x0imp(c)	c∈COM	Total supplies of imported goods
t1(c,s,i)	c∈COM s∈SRC i∈IND	Power of tax on intermediate
t2(c,s,i)	c∈COM s∈SRC i∈IND	Power of tax on investment
t3(c,s)	c∈COM s∈SRC	Power of tax on household
t4(c)	c∈COM	Power of tax on export
t5(c,s)	c∈COM s∈SRC	Power of tax on government
f1tax_csi		Uniform % change in powers of taxes on intermediate usage
f2tax_csi		Uniform % change in powers of taxes on investment
f3tax_cs		Uniform % change in powers of taxes on household usage
f4tax_ntrad		Uniform % change in powers of taxes on nontradtnl exports
f4tax_trad		Uniform % change in powers of taxes on tradtnl exports
f5tax_cs		Uniform % change in powers of taxes on government usage
f0tax_s(c)	c∈COM	General sales tax shifter
pf0cif(c)	c∈COM	C.I.F. foreign currency import prices
t0imp(c)	c∈COM	Power of tariff
delV1tax_csi		Aggregate revenue from indirect taxes on intermediate
delV2tax_csi		Aggregate revenue from indirect taxes on investment
delV3tax_cs		Aggregate revenue from indirect taxes on households
delV4tax_c		Aggregate revenue from indirect taxes on export
delV5tax_cs		Aggregate revenue from indirect taxes on government
delV0tar_c		Aggregate tariff revenue
w1lab_io		Aggregate payments to labour
w1cap_i		Aggregate payments to capital
w1lnd_i		Aggregate payments to land
w1prim_i		Aggregate primary factor payments
w1oct_i		Aggregate "other cost" ticket payments
w0tax_csi		Aggregate revenue from all indirect taxes
w0gdpinc		Nominal GDP from income side
delV1PTX_i		Ordinary change in all-industry production tax revenue
delV0tax_csi		Aggregate revenue from all indirect taxes
x2tot_i		Aggregate real investment expenditure
p2tot_i		Aggregate investment price index
w2tot_i		Aggregate nominal investment
x4tot		Export volume index

.../continued

Appendix J: List of Variables (continued)

Symbol	Dimensions	Description
p4tot		Exports price index, local currency
w4tot		Local currency border value of exports
x5tot		Aggregate real government demands
p5tot		Government price index
w5tot		Aggregate nominal value of government demands
x6tot		Aggregate real inventories
p6tot		Inventories price index
w6tot		Aggregate nominal value of inventories
x0cif_c		Import volume index, C.I.F. weights
p0cif_c		Imports price index, C.I.F., local currency
w0cif_c		C.I.F. local currency value of imports
x0gne		Real GNE
p0gne		GNE price index
w0gne		Nominal GNE
x0gdpepx		Real GDP from expenditure side
p0gdpepx		GDP price index, expenditure side
w0gdpepx		Nominal GDP from expenditure side
delB		(Nominal balance of trade)/(nominal GDP)
x0imp_c		Import volume index, duty-paid weights
w0imp_c		Value of imports plus duty
p0imp_c		Duty-paid imports price index, local currency
p0realdev		Real devaluation
p0toft		Terms of trade
employ(i)	$i \in \text{IND}$	Employment by industry: wage bill weights
employ_i		Aggregate employment: wage bill weights
x1cap_i		Aggregate capital stock, rental weights
x1lnd_i		Aggregate land stock, rental weights
x1prim_i		Aggregate primary factor use (excludes tech change)
p1prim_i		Index of factor cost (excludes tech change)
p1lab_io		Average nominal wage
realwage		Average real wage
p1cap_i		Average capital rental
p1lnd_i		Average land rental
ggro(i)	$i \in \text{IND}$	Gross growth rate of capital = Investment/capital
gret(i)	$i \in \text{IND}$	Gross rate of return = Rental/[Price of new capital]
finv1(i)	$i \in \text{IND}$	Shifter to enforce DPSV investment rule
finv2(i)	$i \in \text{IND}$	Shifter for "exogenous" investment rule
finv3(i)	$i \in \text{IND}$	Shifter for longrun investment rule
invslack		Investment slack variable for exogenizing aggregate investment
f2tot		Ratio, investment/consumption
fgret(i)	$i \in \text{IND}$	Shifter to lock together industry rates of return
capslack		Slack variable to allow fixing aggregate capital
f1lab(i,o)	$i \in \text{IND} \quad o \in \text{OCC}$	Wage shift variable
f1lab_i(o)	$o \in \text{OCC}$	Occupation-specific wage shifter
x1lab_i(o)	$o \in \text{OCC}$	Employment by occupation
f1lab_o(i)	$i \in \text{IND}$	Industry-specific wage shifter
f1lab_io		Overall wage shifter
p1lab_i(o)	$o \in \text{OCC}$	Average wage of occupation

.../continued

Appendix J: List of Variables (continued)

Symbol	Dimensions	Description
f1oct(i)	$i \in \text{IND}$	Shift in price of "other cost" tickets
f3tot		Ratio, consumption/ GDP
p0imp(c)	$c \in \text{COM}$	Basic price of imported goods = $p0(c, \text{"imp"})$
SalesDecomp(c,d)	$c \in \text{COM}$ $d \in \text{DESTPLUS}$	Sales decomposition
x0loc(c)	$c \in \text{COM}$	Real percent change in LOCSALES (dom+imp)
fandecomp(c,f)	$c \in \text{COM}$ $f \in \text{FANCAT}$	Fan decomposition
contGDPexp(e)	$e \in \text{EXPMAC}$	Contributions to real expenditure-side GDP
contBOT		Contribution of BOT to real expenditure-side GDP
x0gdpinc		Real GDP from the income side
contGDPinc(c)	$c \in \text{CONTINC}$	Income side real GDP decomposition
x0gdpfac		Real GDP at factor cost
contGDPfac(c)	$c \in \text{CONFAC}$	Contributions to real GDP at factor cost: $x0gdpfac$
pLabEff		Effective price of labour, inc. labour-saving tech change
pLabEff_p1prim		$P\text{LabEff} / p1\text{prim}_i$
pLabEff_p3tot		$P\text{LabEff} / p3\text{tot}$
p3tot_p0GNE		$P3\text{tot} / p0\text{GNE}$
p0GNE_p0GDPExp		$P0\text{GNE} / p0\text{GDPExp}$
p0GDPExp_p1prim		$P0\text{GDPExp} / p1\text{prim}_i$
pCap_p1prim		$P\text{Cap} / p1\text{prim}_i$
pCap_p2tot		$P\text{Cap}_i / p2\text{tot}$
p2tot_p0GNE		$P2\text{tot}_i / p0\text{GNE}$
wgos		Total GOS
whougos(h)	$h \in \text{HOU}$	Household receipts/income from GOS
whougos_h		Total Household receipts/income from GOS
wentgos		Enterprises' receipts/income from GOS
wgovgos		Government receipts/income from GOS
went		Total income of enterprises
wenthou(h)	$h \in \text{HOU}$	Enterprise receipts from household
wenthou_h		Total enterprise receipts from households
wentrow		Enterprise receipts from the rest of the world
went_posttax		Enterprise tax payments
whouent(h)	$h \in \text{HOU}$	Household receipts from enterprises
wgovent		Government transfers to enterprises
wtaxent		Enterprise direct income tax payments
wrowent		Row income from enterprises: repatriated profits
wsavent		Enterprise savings
ftaxent		Ad valorem rate of corporation tax
whouent_h		Total household income/dividends from enterprises
w1lab(i,o)	$i \in \text{IND}$ $o \in \text{OCC}$	Labour income
w1labinc(i,o,h)	$i \in \text{IND}$ $o \in \text{OCC}$ $h \in \text{HOU}$	HH Labour income
labslack(i,o)	$i \in \text{IND}$ $o \in \text{OCC}$	Employment rate
w1labh_oh(i)	$i \in \text{IND}$	Labour bills
w1labh_i(h,o)	$h \in \text{HOU}$ $o \in \text{OCC}$	Labour income of hh
w1labinc_io(h)	$h \in \text{HOU}$	Total hh Labour income
winchou(h)	$h \in \text{HOU}$	Pre-tax h'hold income
whougov(h)	$h \in \text{HOU}$	Gov transfers to households
whouhou(hto,hfrm)	$hto \in \text{HOU}$ $hfrm \in \text{HOU}$	Intra-hh transfers
whourow(h)	$h \in \text{HOU}$	Gov transfers to households

.../continued

Appendix J: List of Variables (continued)

Symbol	Dimensions	Description
wgovhou(h)	$h \in \text{HOU}$	Hh transfers to govt
wtaxhou(h)	$h \in \text{HOU}$	Household tax payments
wdispinc(h)	$h \in \text{HOU}$	Post-tax h'hold income
winchou_h		Total pre-tax h'hold income
f_inctaxrate(h)	$h \in \text{HOU}$	Income tax shifter: by income
f_inctaxrate_h		Income tax shifter: overall
wdispinc_h		Total post-tax h'hold income
avetax_h		Average Tax Factor: avedispwager - avewager
f3toth(h)	$h \in \text{HOU}$	Consumption function ratio [consumption/GDP]
f3tot_h		Over-all shifter for consumption
wrowhou(h)	$h \in \text{HOU}$	Household transfers to row
delSAVHOU(h)	$h \in \text{HOU}$	Household saving
wgovrow		Transfers from ROW to gov
wincgov		Government income
wcurgov		Current gov expenditure
wrowgov		GOV transfers to ROW
wcapgov		Investment gov expenditure
s2gov(i)	$i \in \text{IND}$	Gov share of investment by industry
wexpgov		Total gov expenditure
wsavgov		Gov (income - expenditure)
delsavgov		Gov (income - expenditure)/GDP
realsavgov		Real gov (income - expenditure)
wcappriv		Investment private expenditure
wexprow		Total ROW expenditure
wrowinc		Total ROW income
wsavrow		ROW (income - expenditure)
wsamcheck		Global (income - expenditure)
wtaxtot		Sum of commodity and income tax revenue
rtaxtot		Real tax collection: for nom homogeniety test
f_entinctax		Shifter: tax on enterprises, Fgostax follows f_inctaxrate_h
x3totq(d,g)	$d \in \text{DEC} \quad g \in \text{GEN}$	Household consumption
x3totq_d(g)	$g \in \text{GEN}$	Hh cons by gender
x3totq_g(d)	$d \in \text{DEC}$	Hh cons by Deciles
xinchou(h)	$h \in \text{HOU}$	Real household income
xinchouq(d,g)	$d \in \text{DEC} \quad g \in \text{GEN}$	Real Household income
xinchouq_d(g)	$g \in \text{GEN}$	Hh cons by gender
xinchouq_g(d)	$d \in \text{DEC}$	Hh cons by Deciles
xdispinc(h)	$h \in \text{HOU}$	Real household income
xdispincq(d,g)	$d \in \text{DEC} \quad g \in \text{GEN}$	Real Household income
xdispincq_d(g)	$g \in \text{GEN}$	Hh cons by gender
xdispincq_g(d)	$d \in \text{DEC}$	Hh cons by Deciles