Government
Institute for
Economic Research

Tutkimukset 171

The VATTAGE Regional Model VERM – A Dynamic, Regional, Applied General Equilibrium Model of the Finnish Economy

Juha Honkatukia

VATT Tutkimukset 171 | April 2013

VATT RESEARCH REPORTS

171

The VATTAGE Regional Model VERM - A Dynamic, Regional, Applied General Equilibrium Model of the Finnish Economy

Juha Honkatukia

Valtion taloudellinen tutkimuskeskus
Government Institute for Economic Research
Helsinki 2013

ISBN 978-952-274-073-1 (nid.)
ISBN 978-952-274-074-8 (PDF)

ISSN 0788-5008 (nid.)
ISSN 1795-3340 (PDF)

Valtion taloudellinen tutkimuskeskus

Government Institute for Economic Research

Arkadiankatu 7, 00100 Helsinki, Finland

Email: etunimi.sukunimi@vatt.fi

Oy Nord Print Ab

Helsinki, April 2013

The VATTAGE Regional Model VERM - A Dynamic, Regional, Applied General Equilibrium Model of the Finnish Economy

Government Institute for Economic Research
VATT Research Reports 171/2013

Juha Honkatukia

Abstract

Applied general equilibrium models have become a standard tool for the analysis of structural policies in many countries and international research organisations. Their use has been prompted by both developments in economics, but also on the growing need for quantitative policy analysis. The analysis of actual policy options mandates the use of numerical methods, but there are several other reasons to suggest the use of AGE models in particular. Chief among these is the applicability of models that rely on explicit optimisation on the analysis of welfare impacts of structural policies. It may also be the case that many policy issues are intractable by theoretical models, for example, when the policies concern several sectors or regions of the economy, or involve contradicting effects.

This report describes the VERM model used in VATT, the Government Institute for Economic Research. The model has been used to study the effects of various tax policies, regional policies, and environmental policies on the economy. Lately, the model has seen extensive use in the anticipation of regional development, especially in the labour markets. The report contains a full description of the model code and its underlying theory. With the help of examples, it also shows how simulation results can be interpreted.

Key words: AGE models, simulation models

JEL classes: H300, E600, C680, H200, R13

Tiivistelmä

Laskennallisia yleisen tasapainon malleja käytetään laajalti talouspolitiikan vaikutusten ja talouden rakenteessa tapahtuvien muutosten arviointiin ja ennakointiin. Laskennallisten mallien käyttöä on lisännyt niiden soveltuvuus aitojen politiikkavaihtoehtojen arviointiin, mutta niiden suurin vahvuus kumpuaa johdonmukaisesta, talousteoriaan pohjautuvasta rakenteesta. Niinpä tasapainomallit huomioivat vuorovaikutukset johdonmukaisella tavalla läpi koko talouden. Teoreettinen pohja mahdollistaa myös hyvinvointivaikutusten mielekkään analyysin. Mallien yksityiskohtaisuus puolestaan mahdollistaa vaikutusten arvioinnin talouden eri tasoilla makrotalouden tasolta toimialojen ja työmarkkinoiden tasolle. Tässä raportissa kuvataan dynaamisen VERM-mallin tietoaineistoja ja rakennetta. Raportti sisältää mallin koodin ja sen perustana olevan teorian kattavat kuvaukset. Raportissa esitellään mallin käyttöä talouspolitiikan analyysiin myös esimerkkien avulla.

Asiasanat: Yleisen tasapainon mallit

JEL-luokat: H300, E600, C680, H200, R13

Foreword

Development of the VATTAGE and VERM models has been a major undertaking by the Government Institute for Economic Research (VATT). This report summarises the latest version of VERM in its generic form. It is intended to serve two purposes. First, it contains a general outline of the model for those wanting to get an overview; second, it includes a detailed description of the model code for those interested in applying the model themselves.

The idea for developing large, applied general equilibrium models at VATT originally arose from policy questions that could best be analysed with such models. The need to serve real-world policy making and analysis has influenced the development of the model, and the development work has benefited from the comments and suggestions of many people working with policy analysis. This has been a truly collaborative effort. I want to mention especially Pekka Tervo and Ilkka Mella from the Ministry of Employment and the Economy, who have for many years provided us with a policy maker's perspective and concrete suggestions regarding the inclusion of various policy issues. Many people from Statistics Finland have contributed to our work through their excellent collaboration. We have also had a very constructive dialogue with the many users of our results in different parts of the country, whom I would like to all thank. From the many researchers in VATT who have participated in regional policy modelling over the years, I would especially like to thank Jussi Ahokas, Jouko Kinnunen, Kimmo Marttila and Antti Simola.

Throughout the project, we have enjoyed very fruitful collaboration with the staff of the Centre of Policy Studies at Monash University. My thanks go to all of them, especially Mark Horridge and Glyn Wittver, who have helped tremendously with their expertise on constructing and managing regional databases, as well as with ideas for stylizing the model code. VERM also uses Mark's concise notation for much of the AGE core of the model. The dynamics stem from the MONASH-like treatment in VATTAGE, as does the treatment of the public sectors, and these in turn have benefitted enormously from the generous guidance of Peter Dixon and Maureen Rimmer.

Helsinki, February 1, 2013

Juha Honkatukia

Summary

This report describes VERM, the regional, dynamic, applied general equilibrium model used in VATT, the Government Institute for Economic Research. The model has been used to study the effects of especially regional policies on the economy, and lately, has become an integral part of the anticipation of long-term developments in the regional economies The report contains a full description of the model code and its underlying theory. With the help of examples, it also shows how simulation results can be interpreted.

Applied general equilibrium models have become a standard tool for the analysis of structural policies in many countries and international research organisations. Their use has been prompted by both developments in economics, but also on the growing need for quantitative policy analysis. The analysis of actual policy options mandates the use of numerical methods, but there are several other reasons to suggest the use of AGE models in particular. Chief among these is the applicability of models that rely on explicit optimisation on the analysis of welfare impacts of structural policies. It may also be the case that many policy issues are intractable by theoretical models, for example, when the policies concern several sectors of the economy or involve contradicting effects. The AGE approach also lends itself to the construction of scenarios of the future of the economy, where the requirement for explicitly stating the underlying assumptions facilitates a critical study of the driving forces behing economic growth, and where the AGE approach assures the internal consistency of the scenarios.

AGE models have been used in the analysis of a wide variety of issues, but their most natural applications are in the analyses of changes in policies or in the operating environment. Policy analyses have covered the effects on the economy of changes in taxes, tariffs, environmental policies, public expenditures and social security, to name a few, whereas the effects on the economy of changes in world markets, the availability of natural resources or technologies are some of the questions covered by AGE modelling. AGE models also often contain a great deal of detail, allowing results to be examined at different levels ranging from the regional to the sectoral to the national. Often the models also cover distributional issues.

This report takes a two-pronged approach to explaining the theoretical logic of VERM. First, we describe the structure and the behavioural assumptions used in the model in an informal way and show how the model can be used in policy analysis with the help of examples that work through a policy simulation with the help of a simplified, back-of-the-envelope model of the economy that captures the basic assumptions of the simulation model. Second, we explain the theory of the model in detail in algebraic form and in terms of the model code.

The report does not introduce the reader to the simulation software, but it does describe the main features of the programming code, the TABLO language, to facilitate an understanding of the presentation of the behavioural equations used in the model, using examples from the VERM programme code. The report also gives the full VERM programme code. The final chapter of the report explains the derivation of the main behavioural equations of the model, linking the theory to its presentation in the model code.

Contents

1. Introduction

Applied General Equilibrium (AGE) models have become a standard tool for the analysis of structural policies in many countries and international research institutions. More recently, they have also found their uses in forecasting and scenario analysis. This report describes the VERM model used in VATT, the Government Institute for Economic Research. The model has been used to study, for example, the effects of regional policies on the economy, and to anticipate the long term development of the regional economies in general, and the development of communal sector finances in particular. The model has also been used to study scenarios involving the exogenous driving forces of economic growth and the external circumstances of the economy. An example of the former, a recent example is a recent study on the prospects of recovery in the Finnish forest industries (Honkatukia and Simola 2011); of the former, a recent example is the analysis of growing age-related expenditures and their effects on the communal sector (Honkatukia, Ahokas and Kinnunen 2011).

The increased use of AGE models has been prompted by developments in economics, but also on the growing demand for quantitative policy analysis. The analysis of actual policy options mandates the use of numerical methods, but there are several reasons to suggest the use of AGE models in particular. Chief among these is the applicability of models that rely on explicit optimisation in the analysis of welfare impacts of structural policies. It may also be the case that many policy issues are untraceable with theoretical models. For example, when the policies affect several sectors of the economy or include opposing effects, numerical analysis suggests itself.

AGE models can be seen as a natural extension of input-output models, which have been used for policy analysis for decades. This is because both rely on input-output data to capture the basic structure of production. Where IO models use rigid multipliers drawn from the data, AGE models use functions of prices to capture the same input shares. The difference is deeper than appears at first glance, however, since AGE models rely extensively on optimization theories to model the demand for inputs, and cover all other aspects of the economy as well.

The distinguishing features of AGE models can be summarised with three points:

1) Generality – AGE models cover explicitly the optimising behaviour of several agents in the economy. Optimisation draws attention to the significance of the prices of goods and primary factors for demand and production decisions. Often these models also cover the decisions of the public sector, investment, or wage setting.

2) Market equilibrium – AGE models cover the effects of supply and demand on prices in every market.

3) Computability – All of the coefficients and parameters of AGE models are evaluated on the basis of data. While a central part of this data is formed by input-output data on the use of goods and primary factors, IO data alone is not sufficient for the specification of an AGE model. It has to be accompanied by data on income flows and income distribution, for example.

Data are also needed to specify the many behavioral parameters used in the theoretical models of consumption and production that underlie an AGE presentation of an economy.

The first AGE models stem from the work of Johansen (1960), which meets all three criteria above. The development of AGE models is also connected to advances in theoretical GE modelling, and in particular the work of Scarf, who developed algorithms for solving such models. This spawned the first AGE models in the 1970s (Scarf 1973, Shoven and Whalley 1972, 1973, 1974). The rapidly improving computational capabilities and the increasing availability of simulation software speeded up interest in these models and by the early 1990s, AGE models had become one of the standard tools of applied economics used in many areas of applied research and in academic research. AGE modelling had also been included in the curriculum of many universities, and several standard texts had been published (Dervis et al. 1982, Shoven and Whalley 1992 and Dixon et al. 1992).

AGE-models have been applied to the analysis of a wide variety of issues, but their most natural applications are in the analyses of changes in policies or in the operating environment. Policy analyses have covered the effects on the economy of changes in taxes, tariffs, environmental policies, public expenditures and social security, to name a few, whereas the effects on the economy of changes in world markets, the availability of natural resources or technologies are some of the questions covered by AGE modelling. AGE models also often contain a great deal of detail, allowing results to be examined at different levels ranging from the regional to the sectoral to the national. Often the models also cover distributional issues.

This report describes the theoretical logic and underlying data of VERM, an applied general equilibrium model of the Finnish economy. We shall describe the structure of the behavioural assumptions used in the model and relate these to the source data in the national accounts. While we do not introduce the reader to the simulation software, we do describe the main features of the programming code, the TABLO language, to facilitate an understanding of the presentation of the behavioural equations used in the model. The report is organised as follows.

Chapter two contains a general description of the model and the model database and reports its sources in the Finnish national accounts and other official statistics. Chapter two also gives an example of a fairly typical simulation involving changes in taxes, and shows how it can be understood with the help of a simplified, back-of-the-envelope model of the economy that captures the basic assumptions of the simulation model.

Chapter three gives an outline of the main characteristics of the TABLO language using examples from the VERM programme code. The emphasis is on explaining the basic logic of the commands and conventions used in setting up the model. Chapter four reports the full VERM programme code. Chapter five offers the derivation of the main behavioural equations of the model, linking the theory to its presentation in model code.

2. Policy analysis using VERM

2.1 Introduction

VERM, the VATTAGE Regional Model, is a regional, dynamic, applied general equilibrium (AGE) model of the Finnish economy. It can be applied to study the effects of a wide range of economic policies. The VERM database contains detailed information about commodity and income taxes as well as the expenditures and transfers of the public sectora and thus covers most policy instruments available to the government. For regional policy analysis, the model also covers transfers between the three main public sectors in Finland, namely, the central government, local governments, and the social security funds. The aim has been to cover the dues and taxes collected by these institutions as closely as possible.

VERM is based on the dynamic models developed at the Centre of Policy Studies in Monash University. MONASH-type models are used in countries ranging from China and South Africa to the United States (Dixon and Rimmer, 2002). In Europe, models based on MONASH have been developed for Denmark, the Netherlands, and Finland. VERM takes its dynamics and the description of the public sectors from VATTAGE, the MONASH-type model of the Finnish economy. The regional TERM model developed by Mark Horridge at CoPS, in turn, has lent its AGE core to VERM to cover the interlinkages between the regional economies, as well as its many tools for handling the core AGE database.

Several factors explain the popularity of MONASH-type models. The main ones are the advanced and user-friendly software packages that facilitate data handling and the set-up for complicated policy simulations that also allow a very detailed post-simulation analysis of the simulation results. MONASH-type models are also very adaptable to the analyses of different types of policies and different time frames. In forward-looking policy analysis, MONASH-type models offer a disciplined way to forecast the baseline development of the economy. Last, but not least, they also allow the user to replicate and explain the historical development of an economy in great detail, which is not true for most AGE models.

The dynamics of the model lead to gradual adjustment to policies or external shocks to the baseline development of the economy. There are three types of inter-temporal links connecting the consecutive periods in the model: (1) accumulation of fixed capital; (2) accumulation of financial claims; and (3) lagged adjustment mechanisms, notably in the labour markets. Different fiscal rules for the balancing of the public sector budgets can also be specified. The speed of adjustment depends on several parameters: 1) the rates of depreciation of capital at the industry level; 2) the rate of adjustment of returns to capital; and 3) the rate of adjustment of real wages (when sluggish wage adjustment is assumed). These parameters can be derived from national accounts data and econometric studies of, notably, the labour markets. Policies can also affect the rate of adjustment. For example, if it is assumed that the government is willing to run deficits during the adjustment period of the economy to, say, an external shock to raw material prices, the parameter controlling this adjustment process will affect the speed at which the economy converges to a new equilibrium growth path.

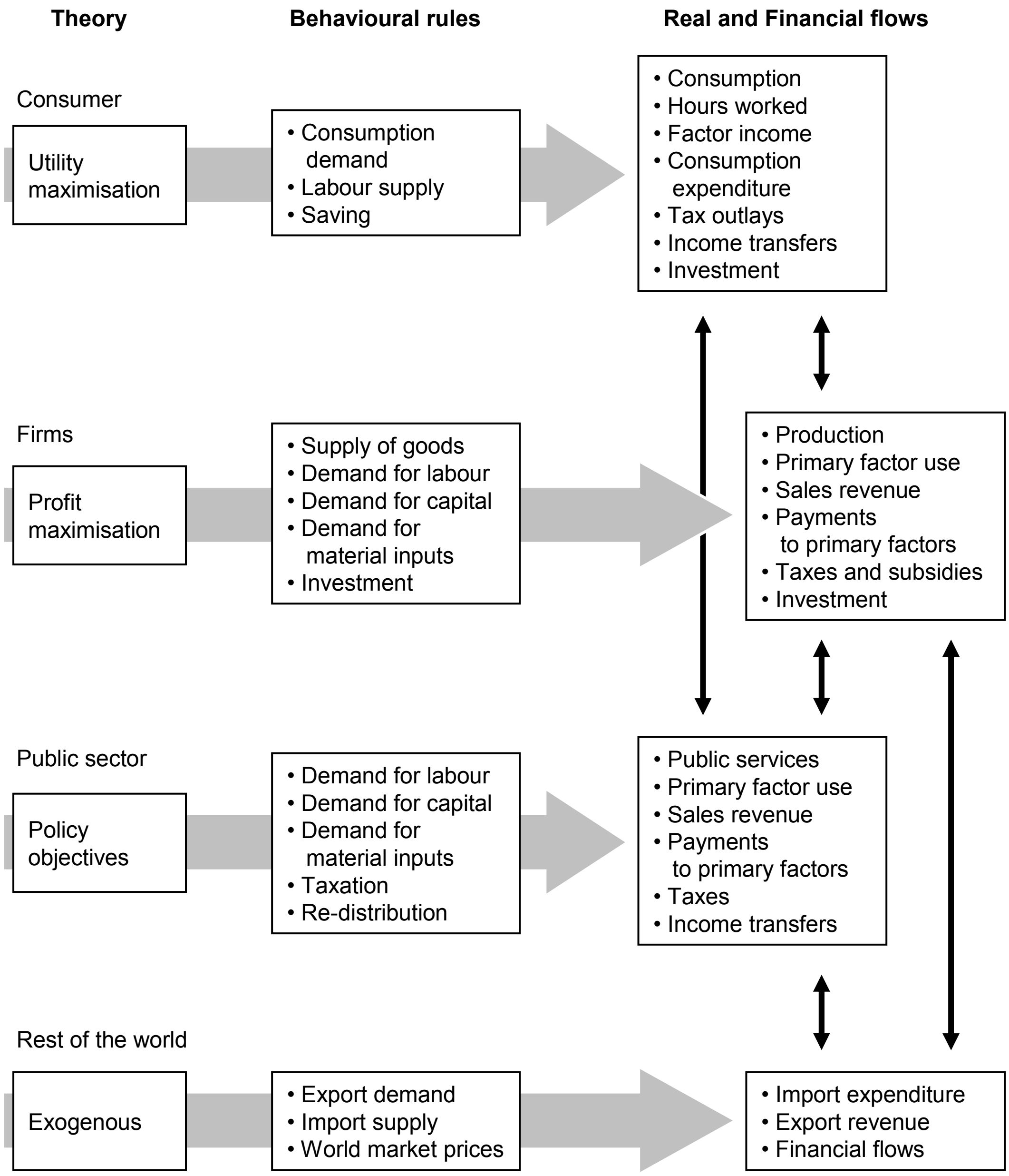
This chapter gives a general outline of VERM. The chapter is divided into three sections. In section 2.2, we describe the VERM database. Section 2.3 gives an overview of the AGE theory behind demand, government finances and labour demand. Each part of this theory is explained more thoroughly in chapter 5, where the implementation of the model in TABLO code is also explained. Section 2.4 is devoted to the dynamic mechanisms of VERM. Section 2.4.1 explains the theory of investment over time, section 2.4.2 asset dynamics, section 2.4.3 deals with the (optional) sluggish wage mechanism. Section 2.5 contains an illustrative application of VERM on the evaluation of tax policies.

2.2 The VERM database

The model is based on an extensive database that describes the transactions between different agents in the economy. In the core of the model are optimization problems of the agents that result in the demand and supply functions of goods and primary factors. The transactions covered by the database and the model are illustrated in Figure 2.1.

The VERM database collects information about the structure of the Finnish economy derived from the national accounts, arranged in a presentation reflecting the theoretical structure of the model. The database also contains the behavioural parameters that are used to operationalise the behavioural assumptions made in the model. National accounts collect data on the use goods and services by industry and by product, but it also contains accounts for production as well as financial positions by institutional sector. (Eurostat 1997, 1.) The institutional sectors are viewed as independent decision-makers (Statistics Finland 2000, 11.), and it is the behaviour of these decision-makers that the model parameters and coefficients derived from the data describe and control.

Figure 2.1 The structure of an Applied General Equilibrium model



A large part of the database uses input-output data to capture the structure of demand for intermediate goods and primary factors by industry and final goods consumption by the consumers, the public sector, and the rest of the world. However, input-output data does not contain data on income flows, which must be obtained from other sources in national accounts; neither does it cover the wide variety of redistribution (of incomes) between the institutional sectors of the economy, which is often in the core of regional policy analyses. Thus a large part of the database is devoted to transactions between the institutional sectors of the economy.

In the database, transactions take place both between domestic sectors, and between domestic and foreign sectors. The domestic sectors are usually divided into five subcategories: Non-financial corporations, Financial corporations, General Government, Households and Non-profit institutions serving households (Eurostat (1997), Eurostat (2003)) whereas the foreign sectors represent foreign countries and multinational and international organisations. These institutional sectors are mutually exclusive and their role in the economy can thus be unequivocally presented. For example, export demand is final demand for domestic goods and services by the foreign sectors.

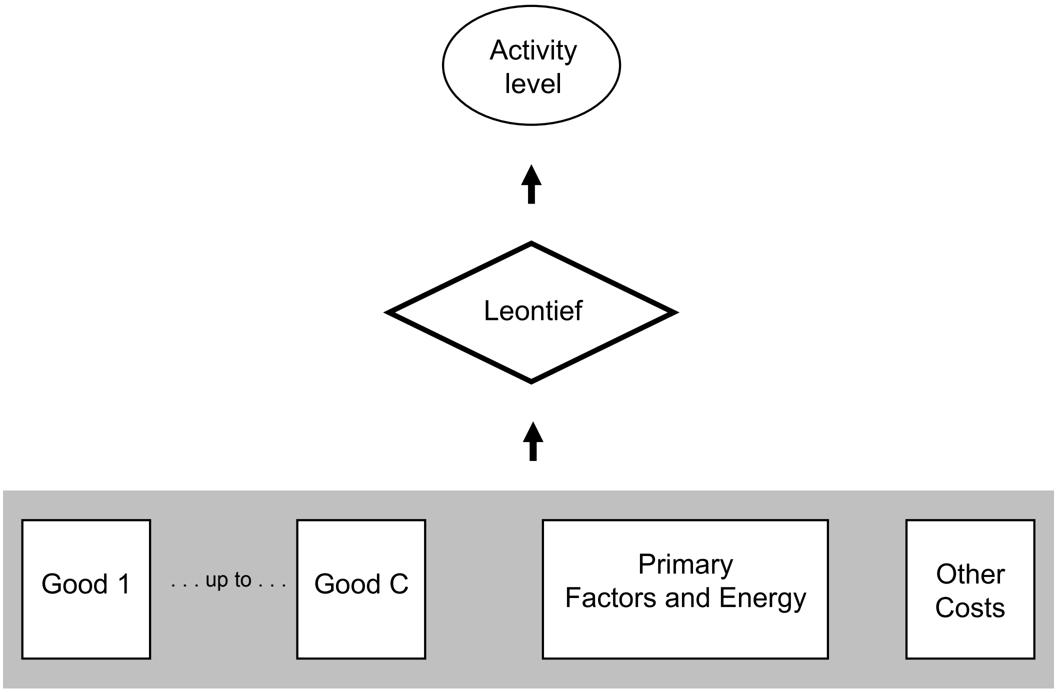
VERM models production with conventional, nested production functions. The idea behind industrial classification is to group activities whose production processes or the products they make are similar. However, VERM also allows for multi-production of commodities. The VERM database uses the national industrial classification TOL 2002, basing on NACE 2002 and ISIC Rev. 3.1, to classify industries, and the CPA-classification to group products. The detailed data on commodities allows us to study the production of goods almost at a process level.

2.3 An overview of the AGE theory of VERM

2.3.1 Demand for intermediate goods and primary factors

VERM models production as consisting of two broad categories of inputs: intermediate inputs and a primary factor-energy bundle (referred to as the KLE-bundle). Firms are assumed to choose the mix of inputs that minimises the costs of production for their level of output. They are constrained in their choice of inputs by a three-level nested production technology. At the first level, intermediate-input bundles and primary-factor bundles are used in fixed proportions to output. These bundles are formed at the second level. Intermediate input bundles are combinations of international imported goods and domestic goods. The VERM recognises two sources for imports, namely, the EU and the rest of the world. The primary-factor bundle is a combination of labour, capital, energy and land. At the third level, the input of labour is formed as a combination of inputs of labour from five different occupational categories.

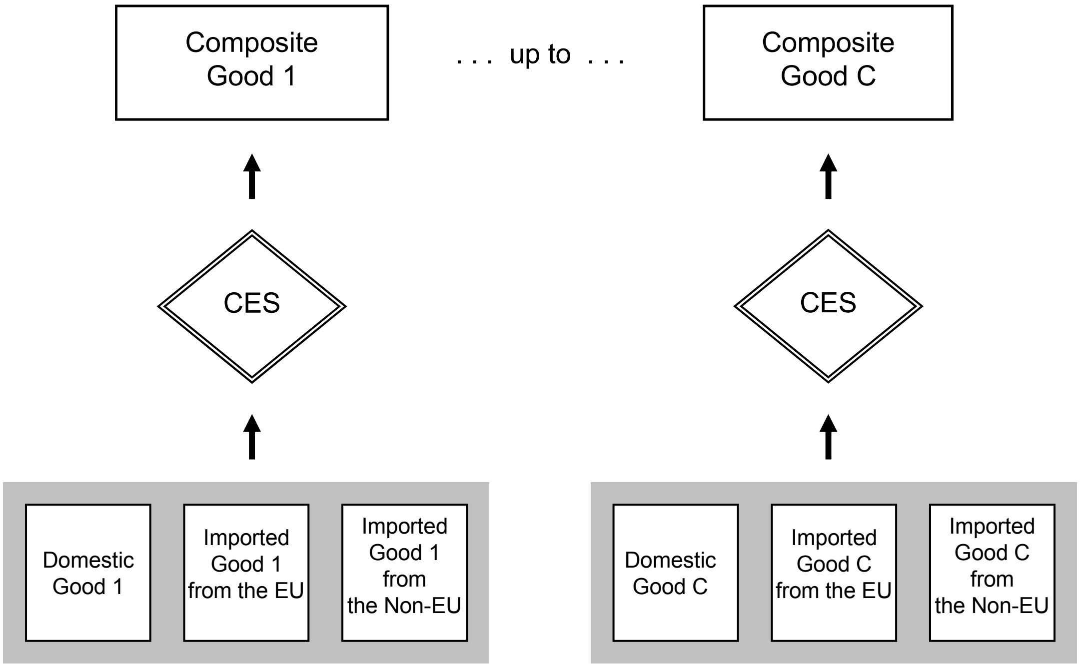
Figure 2.2 Top-level of the input mix



At the bottom level of the intermediate good nest are the demands for commodities from various sources. The firms decide on their demands for the domestic commodities and the foreign imported commodities under a CES assumption, which amounts to the standard Armington assumption that domestic commodities are imperfect substitutes to foreign varieties. Figure 2.3 illustrates the structure giving rise to the demand for the composite goods and individual commodities. We use the well-known GTAP-database as a source for the Armington elasticities (Badri and Walmsley 2008).

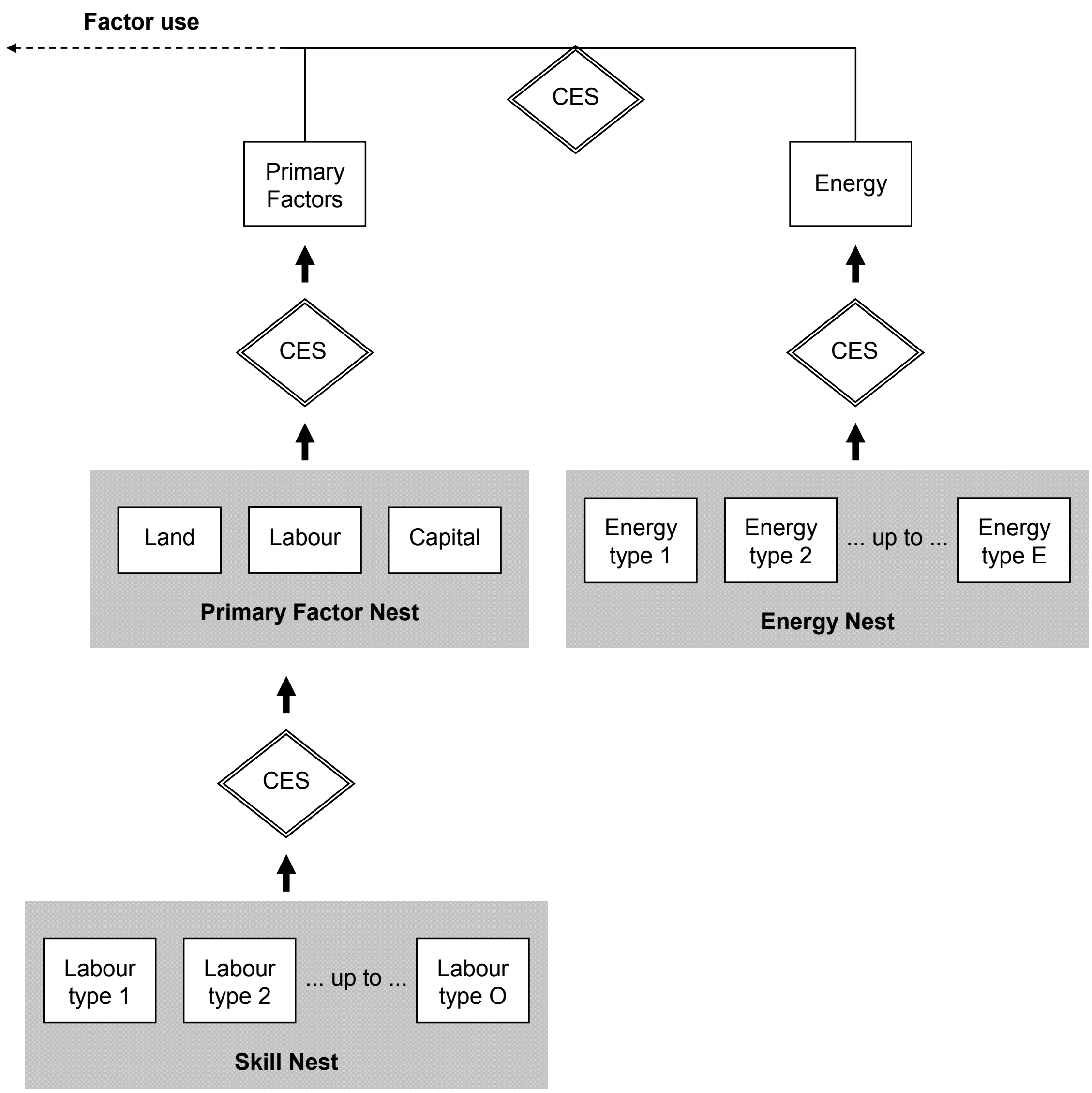
In figure 2.2, an item called other cost tickets is also included. Other costs are costs not related to the use of primary factors or material and energy inputs. In industries with high profitability they often explain profits not directly related to rates of return to capital.

Figure 2.3 The sourcing of inputs in VERM



The demand for primary factors and energy composite are determined by the top nest in figure 2.2. Primary factors and energy are assumed to be combined with energy to form a primary factor-energy nest, often called the KLE-nest, as depicted in figure 2.4 below. The demands for labour of different skills, capital, and energy are derived from this structure. An important characteristic of this structure is that energy and primary factors can be substituted for each other in many industries. Without this assumption, it would be pointless to study policies involving changes in the relative prices of energy and other inputs. At the same time, it is clear the elasticities summarising this substitutability have a potentially large impact on the model's results. We rely on literature for substitution elasticities. The elasticity of substitution between primary factors has been covered in a number of Finnish studies. We follow Jalava et al (2005) who find very low elasticities for Finland; for energy-primary factor elasticities and inter fuel elasticities we use our own estimates.

Figure 2.4 The primary factor-energy composite in VERM



2.3.2 Multiproduct industries and multi-industry products

In the VERM database, many goods are produced by several industries and many industries also produce multiple commodities. This is most notably the case for energy products, where petroleum products stem from refining and where the relative price of the products affects the output mix. There are also some products, such as wood and wood residue used for heating and energy production that can stem from several industries.

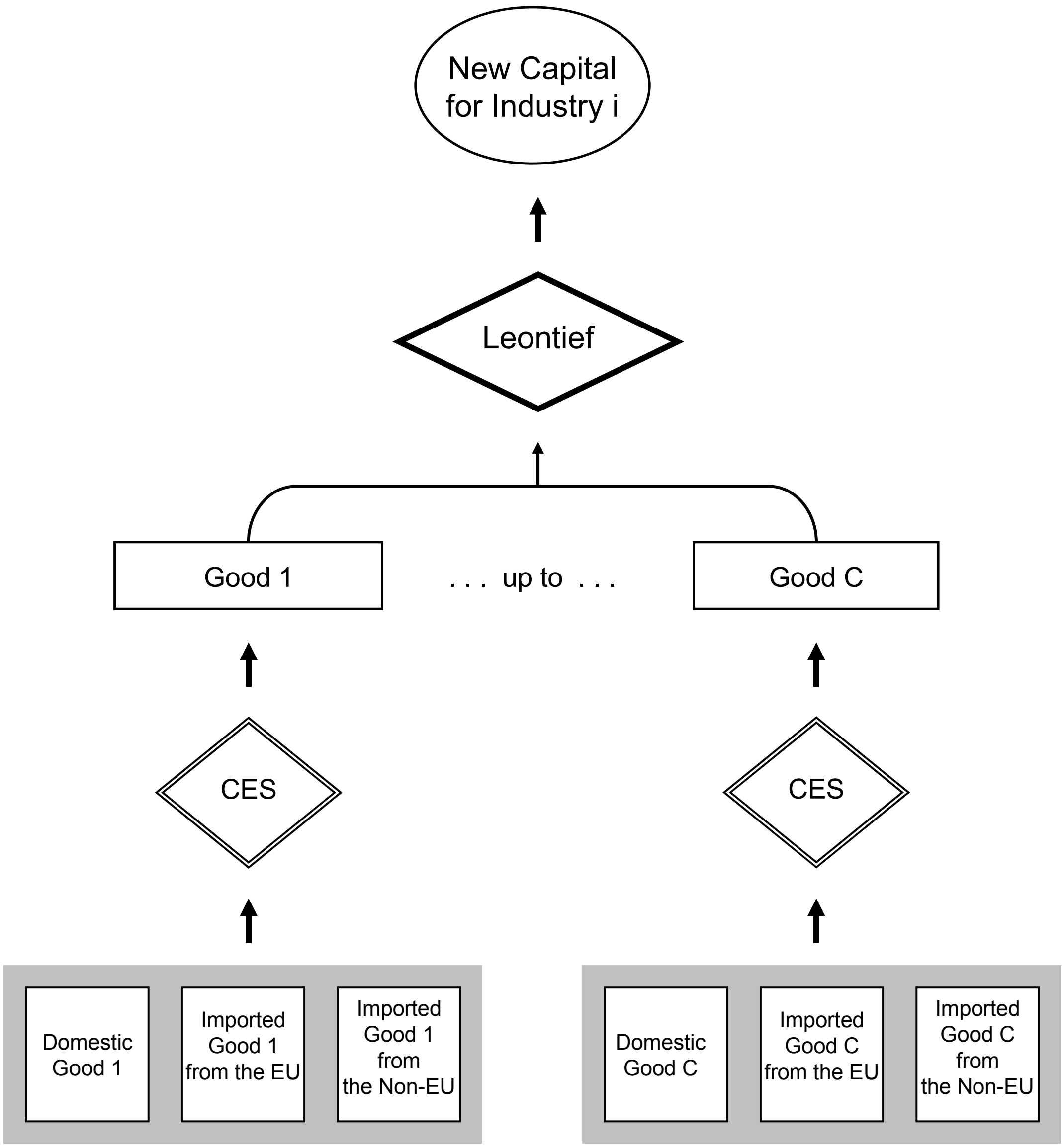
The sourcing of goods from industries raises from national accounts data. The model allows for the possibility that industries are affected by relative prices when deciding their output mix. This decision is modelled as a profit maximisation problem under the assumption of CET transformation technologies between possible outputs.

2.3.3 Demands for inputs to capital creation and the determination of investment

VERM follows standard AGE practice in modelling the production of capital goods with an investment sector, whose task it is to combine inputs to form units of capital. In choosing these inputs they minimise costs subject to a Leontief technology. Figure 2.5 shows the nesting structure for the production of new units of fixed capital.

Capital is produced with inputs of domestically produced and imported commodities. No primary factors are used directly as inputs to capital formation. The use of primary factors in capital creation is recognised through inputs of the commodities, for example, construction services. Where VERM differs from most AGE models is in the description of the capital goods themselves. In VERM, capital is genuinely sector specific, in other words, the commodity inputs for capital to each industry are unique. This means that capital is not malleable but that it will only adjust slowly, over time.

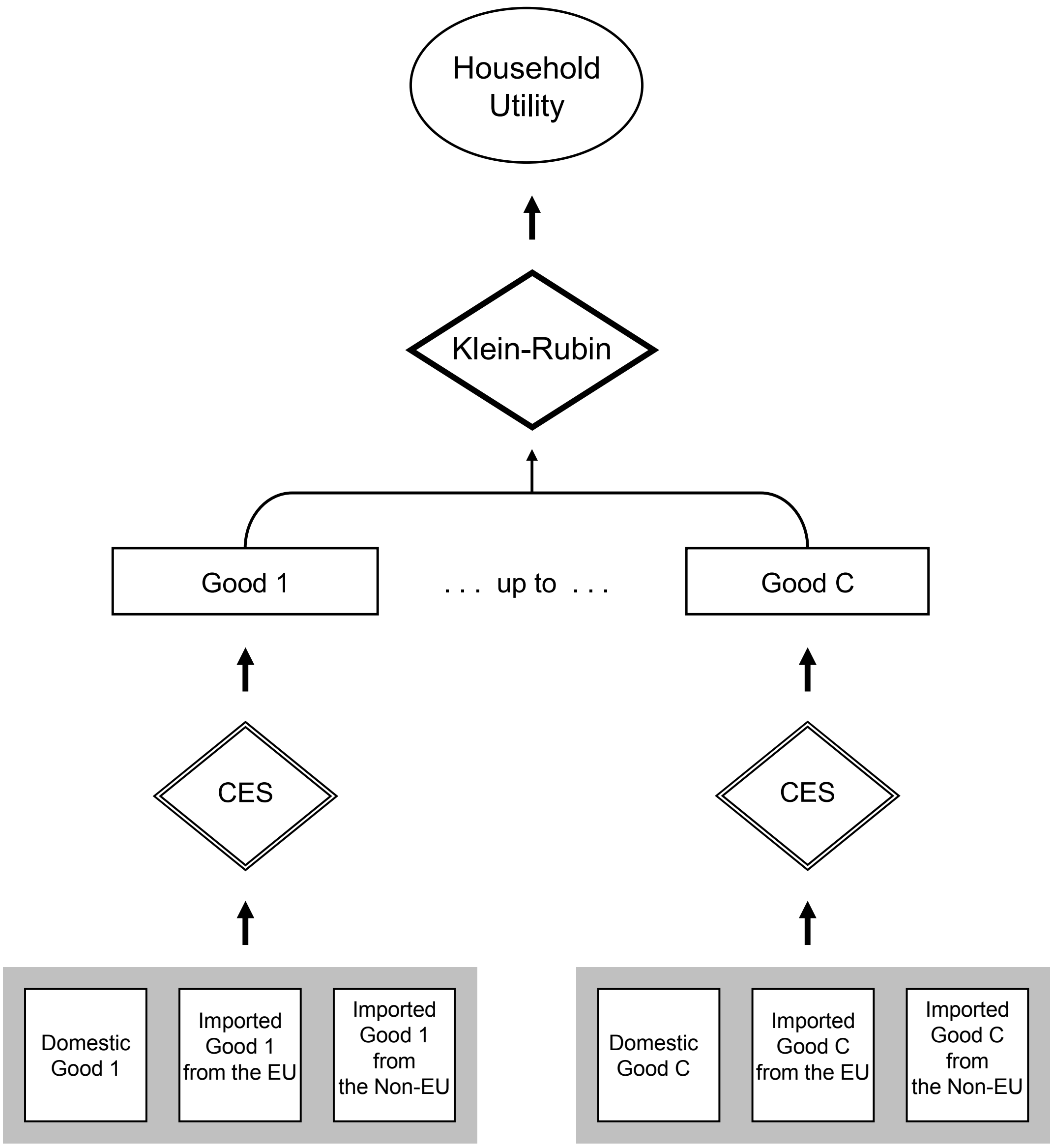
Figure 2.5 Production of investment goods in VERM



2.3.4 Household demand

In VERM, households are assumed to be the recipients of factor incomes. However, they also possess assets and liabilities abroad and domestically, which implies that a part of domestic incomes will be channelled abroad. A Keynesian consumption function then determines the level of household expenditure as a function depending on the (observable) average propensity to consume and on household disposable income, while the demands for individual goods are modelled as a utility maximisation problem subject to a household expenditure constraint. Whether we treat the average propensity to consume as constant depends on the application. When the model is used to accommodate an outside forecast for the macroeconomy, for example, the propensity to consume is endogenous, allowing the model to capture the forecast path of household consumption, whereas in policy applications is usually exogenous. In the regional setting, it may well be the case that there are differences in the propensities to consume between the regions. However, there are also instances where outside information on changes in the propensity to consume, stemming from studies of consumption patterns, can be used in the construction of baseline scenarios. The structure of the utility function is shown in figure 2.6.

Figure 2.6 The structure of household demand in VERM



2.3.5 Export demand

Export demand is modelled by price-sensitive export demand functions. However, there are several possibilities to refine this basic set-up in VERM. First, export demands in VERM can be used to distinguish between traditional and collective exports. For traditional export sectors, each export good faces its own downward-sloping foreign demand curve. Thus a shock that improves price competitiveness of an export sector will result in increased export volume, but at a lower world price. The non-traditional, or collective, exports, on the other hand, face a single export demand function, that is, these exports move together. The composition of collective export demand is also exogenous. The distinction between traditional and collective exports can be used to rule out feedbacks from world prices to domestic prices, which may be of relevance for the service sectors. However, most commodities are modelled with the individual export demand functions. Finally, the supply decision to domestic and exports markets can also be modelled as being price dependent on the relative prices in these markets under the assumption of a CET technology.

2.3.6 Public sector demands for commodities

Commodities are demanded by the public sectors (the central government, the communal sector, and the social secutrity funds). There are several ways of handling these demands, including: (i) endogenously, by a rule such as moving government expenditures with household consumption expenditure or with domestic absorption; (ii) endogenously, as an instrument which varies to accommodate an exogenously determined target such as a required level of government deficit; and (iii) exogenously, by assuming they follow forecasts stemming from outside of the model. In VERM baseline simulations, the last assumption is often used, with official estimates of government spending giving the path that government expenditures take.

2.3.7 Indirect taxes and margin demands

In VERM, supply and demand of commodities are determined through optimising behaviour of agents in competitive markets. The assumption of competitive markets implies equality between the producer's price and marginal cost in each industry and each region. Demand is assumed to equal supply in all markets. However, indirect taxes and margins affect the purchaser’s prices.

The government imposes ad valorem sales taxes on commodities, income and payroll taxes on labour incomes, and capital taxes on capital income. The government also sets production taxes and collects tariffs from imports. These taxes place wedges between the prices paid by purchasers and prices received by the producers. The model recognises margin commodities (e.g., retail trade and road transport freight) which are required for each market transaction (the movement of a commodity from the producer to the purchaser). The costs of the margins are included in purchasers' prices. Needless to say, there are marked inter-regional differences in the cost shares of these margins.

2.4 An overview of VERM dynamics

VERM is a dynamic model that allows the economy to adjust over time to changes in the economic environment or in policies. The most important determinant of this adjustment process is the accumulation of physical capital via investment or disinvestment, and the accumulation of financial assets over time. However, sluggish wage adjustment can also be specified, and there may be an element of sluggishness in policy responses to changes in employment. The dynamic structure of VERM closely follows VATTAGE. This section describes the dynamics in general terms, with a more detailed description being given in chapter 5.

An integral part of dynamic applications of VERM is the baseline, or forecast, scenario of the economy. The baseline forms the reference, to which the effects of changes in policies are compared. In most applications, the baseline is formed on the basis of medium term forecasts and long run scenarios of the development of the macroeconomy that stem from outside of the model, often from the national level forecasts produced using VATTAGE. The baseline also uses forecasts for industry-specific historical trends in productivity, employment, investment, exports, as well as import and export prices, that stem from the process of updating the model’s database. This latter process in effect ensures that the model traces the development of the economy during the past few years. However, in constructing the baseline, it is also possible to introduce industry-specific expert forecasts for particular industries, a feature that has often been used for the large export or regionally dominant industries, and for the sectors producing public services.

2.4.1 Capital stocks, investment and the inverse-logistic relationship

In each year of year-to-year simulations, we assume that industries’ capital growth rates (and thus investment levels) are determined according to functions which specify that investors are willing to supply increased funds to industry j in response to increases in j’s expected rate of return. However, investors are cautious. In any year, the capital supply functions limit the growth in industry j’s capital stock so that disturbances in j’s rate of return are eliminated only gradually.

The VERM treatment of capital and investment in year-to-year simulations can be compared with that in models recognizing costs of adjustment (see, for example, Bovenberg and Goulder, 1991). In costs-of-adjustment models, industry i's capital growth (and investment) in any year is limited by the assumption that the costs per unit of installing capital for industry i in year t are positively related to the i's level of investment in year t. In the MONASH treatment, we assume (realistically) that the level of i's investment in year t has only a negligible effect (via its effects on unit costs in the construction and other capital supplying industries) on the costs per unit of i's capital. Instead of assuming increasing installation costs, we assume that i's capital growth in year t is limited by investor perceptions of risk. Investors are willing to allow the rate of capital growth in industry i in year t to move above i's historically normal rate of capital growth only if they expect to be compensated by a rate of return above i's historically normal level.

This theory is fully explained in chapter 5; here, we note that the treatment of capital goods as being industry-specific also introduces an element of sluggishness.

In every region, the evolution of the industry-specific capital stock follows the familiar equation:

$$K_{i,t+1} = (1 - D_i)*K_{i,t} + I_{i,t} \tag{2.1}$$

where

$K_{i,t}$ is the capital stock at the beginning of year t in industry i;

$K_{i,t+1}$ is the capital stock at the end of year t in industry i;

$I_{i,t}$ is investment during year t in industry i; and

D_i is a parameter giving the rate of depreciation in industry i.

In computations for year t, $K_{i,t}$ is set exogenously to reflect i's end-of-year capital stock in year t–1.

In baseline computations, investments in the reference year are given by the data for that year, whereas for the following years in a baseline forecast, they are determined by the returns to capital.

2.4.2 Asset dynamics

Financial assets – liabilities and deficits – provide another inter-temporal link in VERM. The model recognises current account deficits, with the related foreign liabilities, and public sector deficits, which in turn are related to government debt. These deficits are described in detail, and the dynamics depicting the accumulation of the related financial assets.

Accumulation of financial assets and liabilities is modelled with inter-temporal links of the form:

$$D_{q,t+1} = D_{q,t,t} * V_{q,t,t+1} + \left(\frac{D_{q,t} + D_{q,t+1}}{2}\right) * R_{q,t} + J_{q,t} * V_{q,tm,t+1} , \qquad (2.2)$$

where

$D_{q,t}$ is the level of asset or liability of type q at the beginning of year t

$R_{q,t}$ is the average rate of interest or dividend rate for asset or liability of type q during year t

$J_{q,t}$ is the active accumulation of q during year t

$V_{q,t,t+1}$ is the factor translating the value of q from the beginning of year t to beginning of year t+1

and

$V_{q,tm,t+1}$ is the factor translating the value of q from the middle of year t to the beginning of year t+1.

The factors *V* take into account the effects of exogenous changes in exchange rates and interest rates. For example, since VERM deals separately with the EU and non-EU countries, the effects of an appreciation of the US dollar can be taken into account when the debt portfolio is known.

Active accumulation means here new borrowing or investment beyond accumulation of interest and dividends. For example, in a simple foreign debt equation a deficit on the balance of trade is active accumulation while accrued interest and revaluation effects are passive accumulation. While accounting for assets makes the model more complex, it brings with it considerable benefits. Most importantly, by recording the assets and liabilities, VERM is able to generate results for the wealth of Finns which can be taken into account in welfare analysis.

2.4.3 Labour market dynamics

VERM allows for different treatments of the labour markets. The labour market equations relate population and population of working age, and define unemployment rates in terms of demand and supply of labour.

In dynamic simulations, labour supply is typically taken as exogenous, while wages adjust only gradually and unemployment is determined endogenously.

VERM allows for different specifications of the labour markets. In a dynamic setting, it is not unreasonable to assume that there is an element of sluggishness in real wage adjustment. In Finland, this was very much the case until very recently, when wage setting has become more decentralised. The basic set-up of VERM captures the idea that wage setting may be centralized at the national level.

Specifically, we assume that real, after tax wages are sticky in the short run and flexible in the long run. In this labour market specification, policy shocks generate short-run changes in aggregate employment and long-run changes in real wages. Algebraically, we assume that

$$\left(\frac{W_t}{W_t, old}-1\right)=\left(\frac{W_{t-1}}{W_{t-1}, old}-1\right)+\alpha_1\left[\left(\frac{W_{t-1}}{W_{t-1}, old}-1\right)-\left(\frac{E_t}{E_t, old}-1\right)^{\alpha_2}\right] \tag{2.3}$$

In this equation, *old* indicates a base case forecast value. $W_{t,old}$ and $E_{t,old}$ are the real wage rate and the level of employment in year t in the base case forecasts, and W_t and E_t are the real wage rate and the level of employment in year t in the policy simulation. Under this specification, the adjustment of the real wage rate depends on deviations from the expected real wage development and on the deviation of employment from the expected employment growth. The speed of adjustment is controlled by parameter α_1, whereas α_2 determines, whether employment returns to its expected growth path after a policy shock. The real wage equation is close to NAIRU-theories of unemployment, and its parameters have been estimated for Finland studies such as Alho (2002) and McMorrow and Roeger (2000).

2.4.4 Government finances

The public sector is where VERM most distinctly differs from its Australian counterparts. The reason lies simply in the differences between the division of power between the central government and the regional and local authorities. Finland has a strong central and local government but lacks the regional/state level present in many other countries. In addition, social secutrity funds have a markedly larger role in the economy than in many other countries. This has a bearing on the modeling of the roles and powers of the various levels of government.

In many ways, the similarities to other countries are still there, of course. Thus VERM contains a detailed database on indirect taxes, payroll taxes and income taxes. Indirect taxes on commodities are modelled as *ad valorem* rates of tax levied on the basic price of the underlying flow. The basic price is the price received by the producer. VERM allows for differentiation of indirect taxes to environmental taxes and other taxes for certain commodities. Production taxes are modelled as part of value added, while payroll taxes are directly levied on wages. Income taxes are levied on labour and capital incomes. Finally, import duties are levied as ad valorem taxes on imports.

VERM includes revenue equations for income taxes, sales taxes, excise taxes, taxes on international trade and for receipts from government-owned assets. As described already, the model accounts for public expenditures on commodities (or services). It also contains outlay equations for transfer payments to households (e.g., pensions, social secutrity benefits and unemployment benefits), and transfer payments between the central government, the local authorities, and the social security funds. The specification in VERM of government finances makes the model a suitable tool for analysing the effects of changes in the fiscal policies.

2.5 Applying VERM to policy analysis – an energy tax example

A typical application of AGE models involves evaluating the effects of changes in commodity taxes. In many cases, even uniform changes in taxes may have different effects in different parts of the country, if there are regional differences in the usage of those commodities. In this section, we consider the effects of a recent energy tax reform at the regional and at the national levels.

In 2011, the Finnish energy taxation was revised in a number of ways. The previous tax, which was a combination of a fiscal component and an emission based, energy carrier-specific component was replaced by taxes levied according to the energy and emission content of each energy carrier. Simultaneously, some of the exceptions to the strict emission-based component – most notably for natural gas and diesel fuels - present under the previous tax law were to be phased out. Here, we study a stylized version of the reform and ignore the phasing-out period, assuming the new tax code to have been introduced at once starting in 2011.

In essence, the reform lead to a substantial increase in the effective tax rates on diesel and light fuel oils, natural gas, peat, and electricity. The reformed energy taxation still retains significant reimbursements of energy taxes paid out by sectors that are already covered by the European emission trading scheme. It also retains differentiated electricity taxes for manufacturing sectors on the one hand, and for the service sectors and for household consumption on the other, with services and households facing the relatively higher taxes.

Figure 2.7 shows the results of the simulation at the national level. The first result to note is that, under our assumption of sticky real wages, employment falls. This can be understood with the help of a back-of-the-envelope (BOTE) model of the full model. The main point is that the raising of energy taxes has an effect on the overall share of taxes in GDP as well as the relative prices of output and consumption, which affects employment.

The BOTE model assumes that labour is paid according to its value marginal product. We also assume constant returns to scale in production, with GDP given by

$$Y = A*F(K,L) \tag{2.4}$$

where A is total factor productivity, and K and L are the inputs of capital and labour, respectively. The compensation for labour can be expressed as

$$W = \frac{P_g}{T_g} * A * f\left(\frac{K}{L}\right) \tag{2.5}$$

where

W is the nominal before-tax wage rate;

P_g is the price deflator for GDP;

T_g is the power of the indirect tax rate applying to production in Finland (which can also include input taxes, but in this stylized model we will ignore non-primary-factor inputs); and

A*f (K/L) is the marginal product of labour derived from the constant-returns-to-scale production function, Y=A*F(K,L).

Thus

$$\frac{WAT}{P_c} * T_w = \frac{P_g}{T_g} * A * f\left(\frac{K}{L}\right) * \frac{1}{P_c} \tag{2.6}$$

where

WAT is the nominal after-tax wage rate;

T_w is the power of the tax on labour income; and

P_c is the price deflator for consumption.

Rearranging (2.6) gives

$$f\left(\frac{K}{L}\right)=\frac{WAT}{P_c}*\left(T_w T_g\right)*\left(\frac{1}{A}\right)*\left(\frac{P_c}{P_g}\right) \tag{2.7}$$

We assume that the real wage WAT/P_c is sticky in the short run. Now we can see that energy taxes may have an effect on labour demand, since it affects the relative prices of consumption and GDP. It may also affect the share of taxes in GDP. However, from the employer's point of view, real wages are affected by the price of GDP, whereas real wages from the worker's point of view depend on the price of consumption. In order to see the effect, we must work out the combined effects of these changes.

It is important to note that, in the short run, K is fixed. Thus, any changes in GDP will have to stem from changes in employment or from technical change. To understand how our experiment affects employment, we proceed by analyzing the changes in the marginal product of labour in (2.7).

If we assume a CES production function, it can be shown that the percentage change in the marginal product of labour, f(K/L), when K is held constant, is given by:

$$\%\Delta f=-\frac{S_k}{\sigma}\ell \tag{2.8}$$

where 1 is the percentage change in employment, S_k is the capital share in returns to primary factors and σ is the elasticity of substitution between capital and labour. Next we note that in the first year of the policy shock our sticky wage assumption implies that the percentage change in the real after-tax wage rate is given by

$$wat-p_c=\alpha*\ell \tag{2.9}$$

where α is the parameter on the employment term in the lagged wage-adjustment equation. Putting (2.7) and (2.8) into (2.9), we obtain

$$\ell=m*\left(t_w+t_g-a+p_c-p_g\right) \tag{2.10}$$

where m is a negative coefficient given by

$$m = \frac{-1}{\left(\frac{S_k}{\sigma} + \alpha\right)} . \tag{2.11}$$

The value for parameter m in equation 2.11 lies between –0.6 and –0.7 in our data.

The result obtained from equation 2.10 is dominated by the tax terms, but there is also a small change in the term a, reflecting overall efficiency effects. In this back of the envelope model, the *a* encompasses everything (technical change and efficiency effects) that causes GDP to change relative to factor inputs. In the current situation, a is calculated as

$$a = gdp - S_\ell * \ell . \tag{2.12}$$

S_ℓ, the share of labour in returns to primary factors, is 0.6. On this basis, (2.12) gives a value for *a* which is about –0.1.

For our experiment, $t_w = 0$, $t_g = 0.45$,

$a = gdp - S_\ell * \ell = –0.135.$

$P_c = 0.09$ and $P_g = 0.05$.

The result suggests a fall in the terms of trade, which is reflected by the relative change of the price terms, since the consumer price index includes imports but not exports, whereas the GDP price index does contain exports. With these setting the BOTE calculation for *l* comes out at -0.41, while the model gives us the result at -0.31. Clearly the sectoral and regional details matter, but even the simple argument points to the kind of results that we are observing, namely, that employment is falling. By 2.10, employment is falling because the price of GDP is falling relative to the price of consumption goods; and because power of indirect taxes is increasing. Both of these effects tend to disfavour labour.

The argument behind the fall in GDP is easy to grasp once the fall in employment is established. VERM assumes that it takes time to adjust the capital stock, and, furthermore, that capital stocks are sector specific. From this it must follow that short-run changes in employment have an effect on output. This effect is also apparent in figure 2.7. In fact, GDP falls by more than one might initially expect. The short-run change in output as a response to changes in employment can also be deduced from some basic assumptions. Assuming F(K,L) to be a CES function, we have

$$y = a + S_L l + S_K l \tag{2.13}$$

where S_K is capital's share of valuea added and S_L is lavbour's share of value added, and where *k,* and *l* are the percentage changes (log-differentials) of capital and labour inputs. With K fixed in the short run, and with $S_L = 0.59$, GDP could be expected to fall by about 0.19 per cent as employment falls by 0.31 per cent compared to the baseline. The reason GDP falls more is that in reality GDP also depends on the contribution of production and commodity taxes, as is apparent from figure 2.8, which shows the decomposition of income-side GDP into the contributions of each of its components. From the figure it is easy to see that the contributions of production and commodity taxes are as large as the contribution of primary factors. They are mainly due to the hike in energy taxes.

Figure 2.7. also shows a decrease in private consumption and an increase in exports. The fall in consumption is due to the fact that not only have energy goods become more expensive, but the real income of the consumers has also fallen as consumer prices rise. Thus both the price and income effects tend to cut consumption. The fall in investment is related to the fall in employment, which results in (temporarily) higher K/L-ratios and diminishes the rate of return on capital; to compensate, investment tends to fall.

The slight increase in exports may be somewhat more difficult to understand. It is due to the fall in domestic absorption by about 0.7 per cent, which leaves some capacity underutilized, since GDP only falls by 0.3 per cent. To compensate, net exports must increase. The argument is easy to see from the GDP identity

(1) $Y = C + G + I + X - M$

where Y equals GDP, C consumption demand, G government demand, I investment, X exports, and M imports. In percentage changes, we have

(2) $y = S_C c + S_G g + S_I i + S_X x - S_M m$

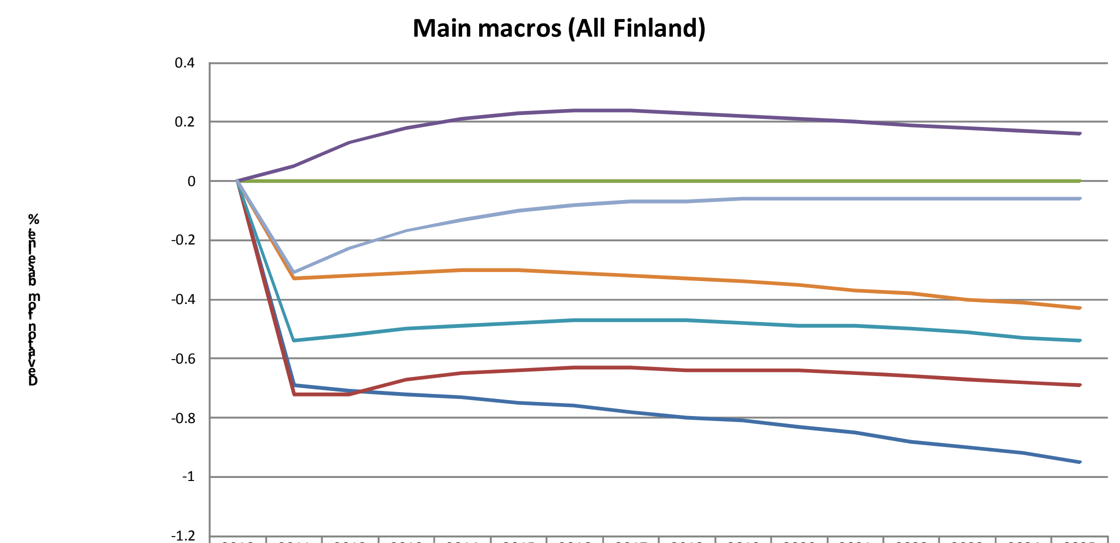
where S_C is the GDP share of consumption demand, S_G the share of government demand, S_I the share of invesmtnet, S_X the share of exports, and S_M the share of imports; and where y, c, g, ,i, x, ja m are the percentage changes in GDP, consumption, government demand, investment, and exports and imports, respectively. With both c and i falling, it is easy to see net exports have to adjust.

The mechanism that allows exports to change is the relative price of domestic exports in the world market. VERM assumes export demands of the Armington form

(3) $x = -\varepsilon(p - p_W) + x_W$

where ε is the export price elasticity; p is the domestic export price level and p_W and x_W the (exogenous) changes in world market prices and demand. Thus for Finnish exports to increase, terms of trade – the relative price of exports – has to fall. As seen above, this is in fact taking place, and hence, the increase in exports. But as can be seen from figure 2.8, which shows the contributions from the expenditure side of GDP, the dominant effect is the fall in domestic consumption.

Figure 2.7 Main macroeconomic effects of the 2011 energy tax reform



	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023	2024	2025
Household demand	0	-0.69	-0.71	-0.72	-0.73	-0.75	-0.76	-0.78	-0.8	-0.81	-0.83	-0.85	-0.88	-0.9	-0.92	-0.95
Investment	0	-0.72	-0.72	-0.67	-0.65	-0.64	-0.63	-0.63	-0.64	-0.64	-0.64	-0.65	-0.66	-0.67	-0.68	-0.69
Government demand	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Exports	0	0.05	0.13	0.18	0.21	0.23	0.24	0.24	0.23	0.22	0.21	0.2	0.19	0.18	0.17	0.16
Imports	0	-0.54	-0.52	-0.5	-0.49	-0.48	-0.47	-0.47	-0.47	-0.48	-0.49	-0.49	-0.5	-0.51	-0.53	-0.54
Real GDP	0	-0.33	-0.32	-0.31	-0.3	-0.3	-0.31	-0.32	-0.33	-0.34	-0.35	-0.37	-0.38	-0.4	-0.41	-0.43
Aggregate employment	0	-0.31	-0.23	-0.17	-0.13	-0.1	-0.08	-0.07	-0.07	-0.06	-0.06	-0.06	-0.06	-0.06	-0.06	-0.06

Figure 2.8 *Contributions of income-side macro variables to change in GDP*

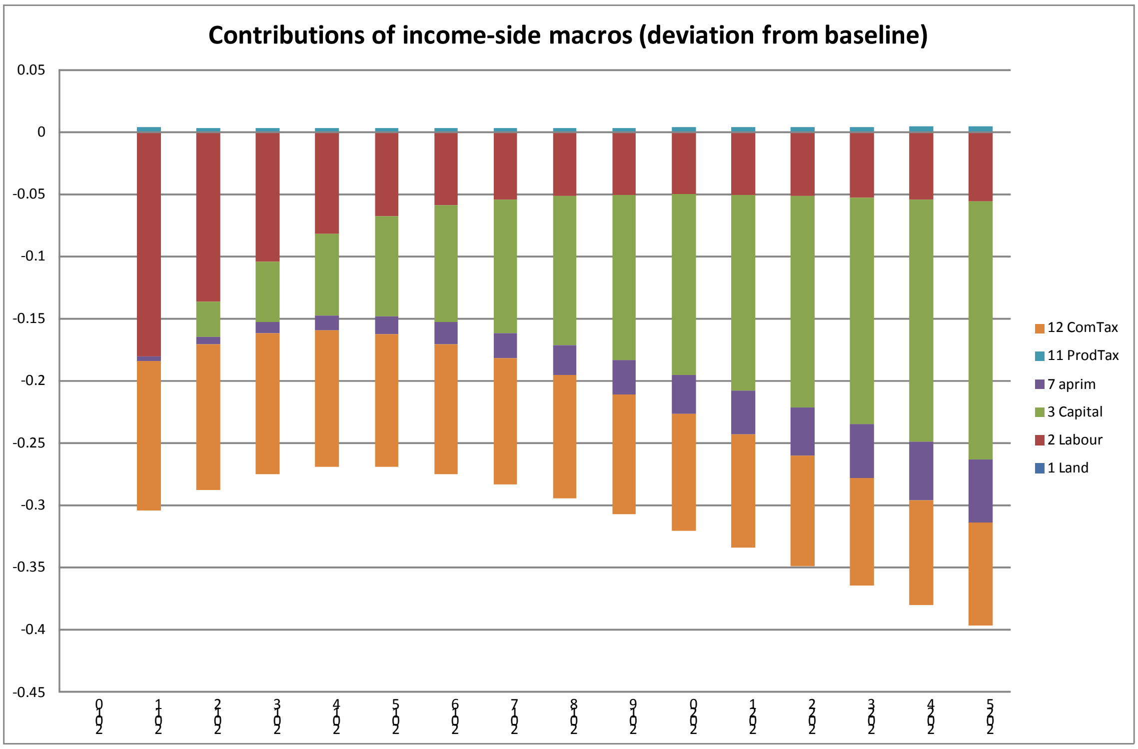
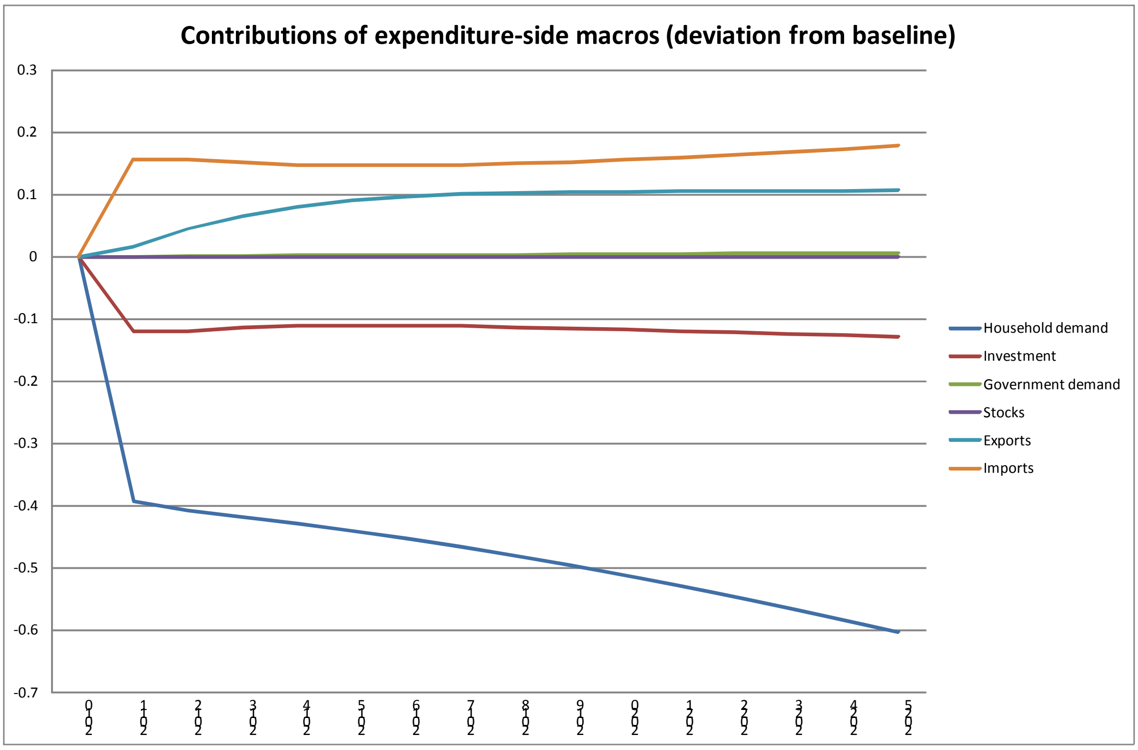


Figure 2.9 *Contributions of expenditure-side macro variables to change in GDP*



Thus far, our focus has been at the national level. However, there are distinct differences in the production and consumption patterns of the 20 (nowadays only 19) Finnish regions originally covered by VERM. For example, the share of energy goods of all material inputs ranges from a low of 3.7 per cent in Etelä-Savo – a south-eastern, rural region – to a high of almost 30 per cent in Itä-Uusimaa in southern Finland – recently annexed to the large southern Uusimaa region and home to a large oil refinery as well as nuclear power plants.

Furthermore, the shares of private and public consumption in regional GDP varies markedly, as does the share of GDP that is exported. These differences affect the way the regions adjust. However, the short-term determination of supply responses still depends on the reactions of employment. We assume that average real wages are determined at the national level, but we do allow for regional real wage differences, since we rule out inter-regional migration in our example. Thus, while employment at the national level tends to return to the baseline growth path – as can be seen from figure 2.7. – this need not be the case for each region.

Figure 2.10 shows the results for regional employment on impact, and for selected consecutive years. The figure show that on impact, some regions are more severely affected than others, and that by the 2020's, employment has in many regions almost returned to its original baseline growth path. This appears to be the case especially in large regions such as Uusimaa, the capital region in the south of the country, and for example, Pirkanmaa, which is characterized by a large share of export-oriented industries.

Figure 2.10 Regional employment (deviation from baseline)

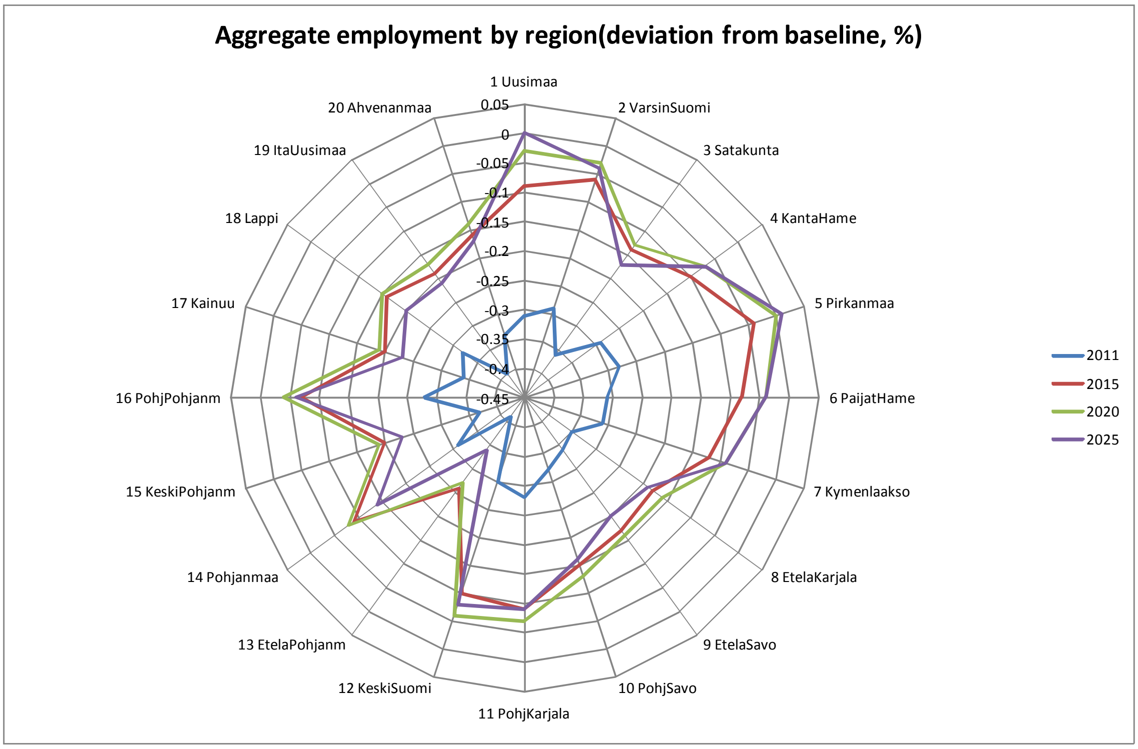


Figure 2.11 shows the effects on regional GDP, and as expected, displays a similar pattern to changes in employment. Figure 2.12 show the effects on household consumption, with its lasting fall evident even at the regional level.

Figure 2.11 Effects on regional GDP (deviation from baseline)

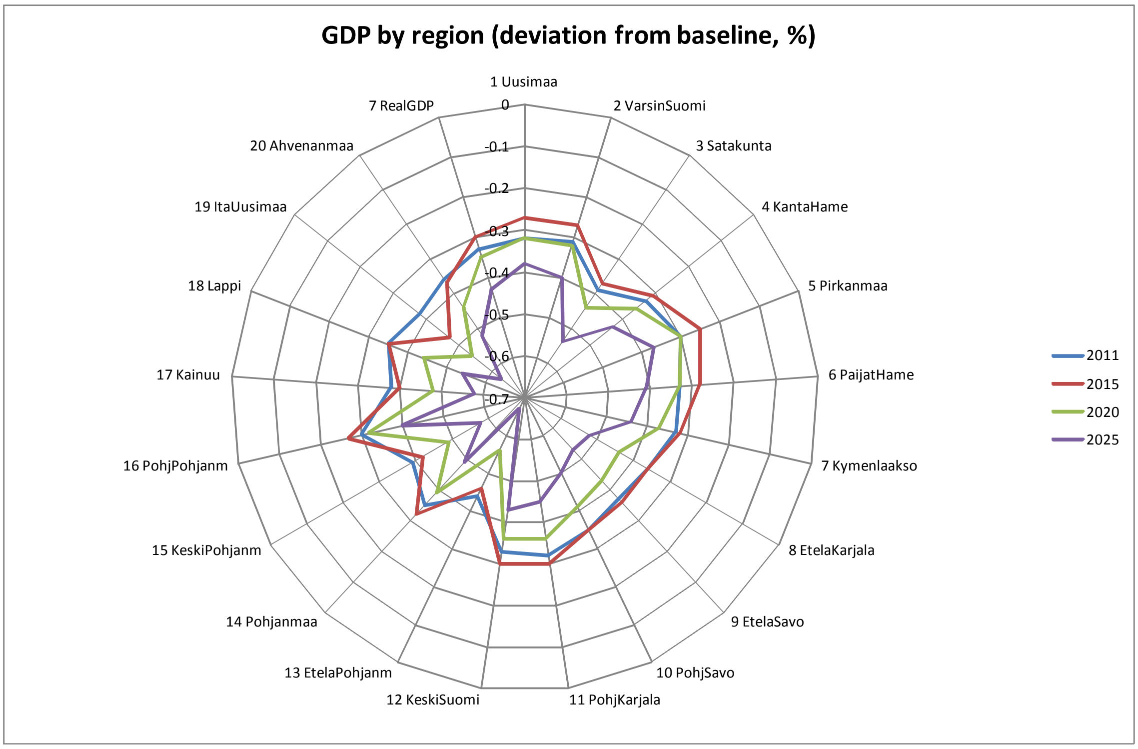
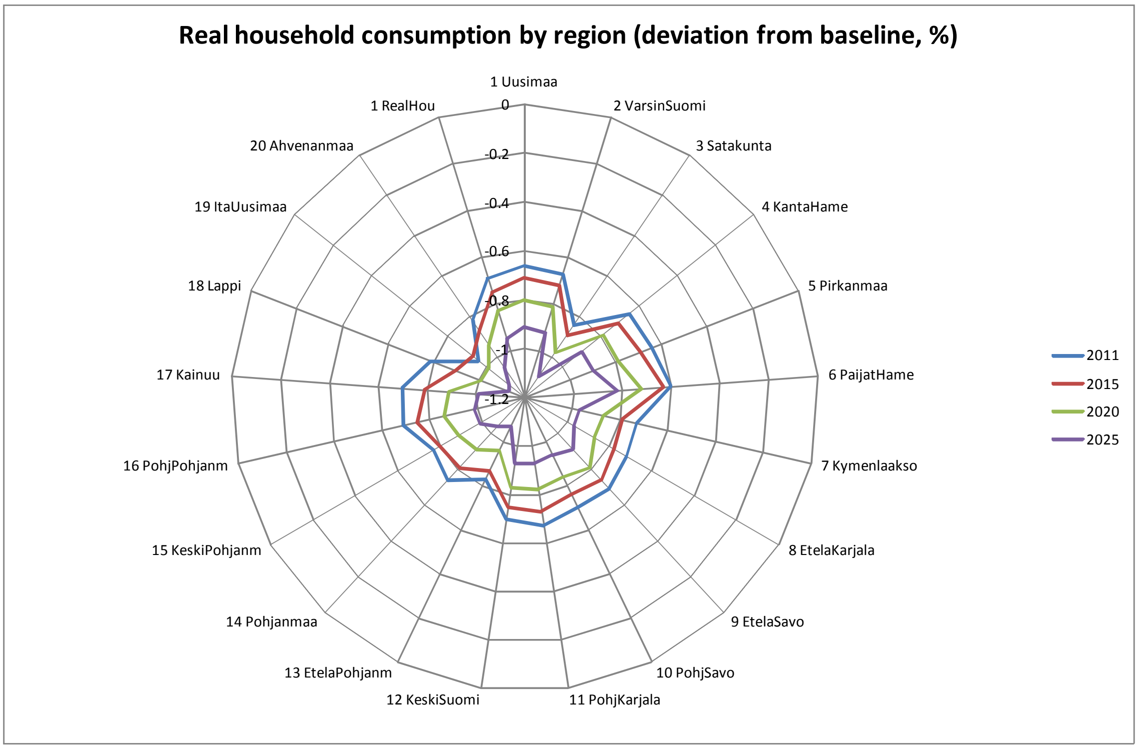


Figure 2.12 *Regional household consumption (deviation from baseline)*



Again, household consumption has an important role, since it not only is the main source of changes in absorption, but since its contribution also affects how much resources are redirected to other uses. To see how regional effects differ on the expenditure side, we pick some regions as examples. Figure 2.13 shows the effects on regional GDP in Pirkanmaa. As is evident, employment in Pirkanmaa returns to baseline growth fairly soon, and, more importantly, as its exports increase, investment is also picking up very soon after the implementation of the tax reform. Figure 2.14 shows the contributions of expenditure aggregates on change in regional GDP in Pirkanmaa. It is readily apparent that the recovery of the local economy is due to increased exports and decreased imports, facilitated by the high share of exports in regional GDP.

This can be contrasted with Pohjois-Pohjanmaa in the North-West of the country in Figure 2.15, which is also relatively export oriented, but with the difference that energy has a higher share in overall material costs than in Pirkanmaa. The recovery of employment is not as rapid as in Pirkanmaa, and, as is apparent from the contributions of expenditure side aggregates in the next figure, adjustment takes place not so much via an increase in exports than via a decrease in imports from other regions and from abroad.

To pick an example from a region that is characterized by concentration on local services and the primary sectors, figure 2. shows the south-easterly Etelä-Savo, which is mostly rural with the few townships mostly providing services for the regional economy. Figure 2. shows that adjustment takes place mostly by cutting down on demand and imports from other regions, even though there is a slight contribution from exports in Etelä-Savo as well.

An interesting feature in all of regions is the contribution of net margins. VERM covers intra-regional trade and transport margins and thus one of the outcomes of the simulation is actually reflecting diminishin trade between the regions. This result is expected, since the energy tax reform increases the relative price of transport fuels, which of course is used in the transport sectors providing transport margins. There are also marked differences between the regions in their use of margins. Figure 2.19 shows the effects on margin demands for 2011, whereas Figures 2.20 and 2.21 show the share of retail and transport margins in household consumption. It is readily apparent, that a large part of the differences in regional responses is due to the varying importance of margins for different parts of the country.

Figure 2.13 Regional main macros (Pirkanmaa)

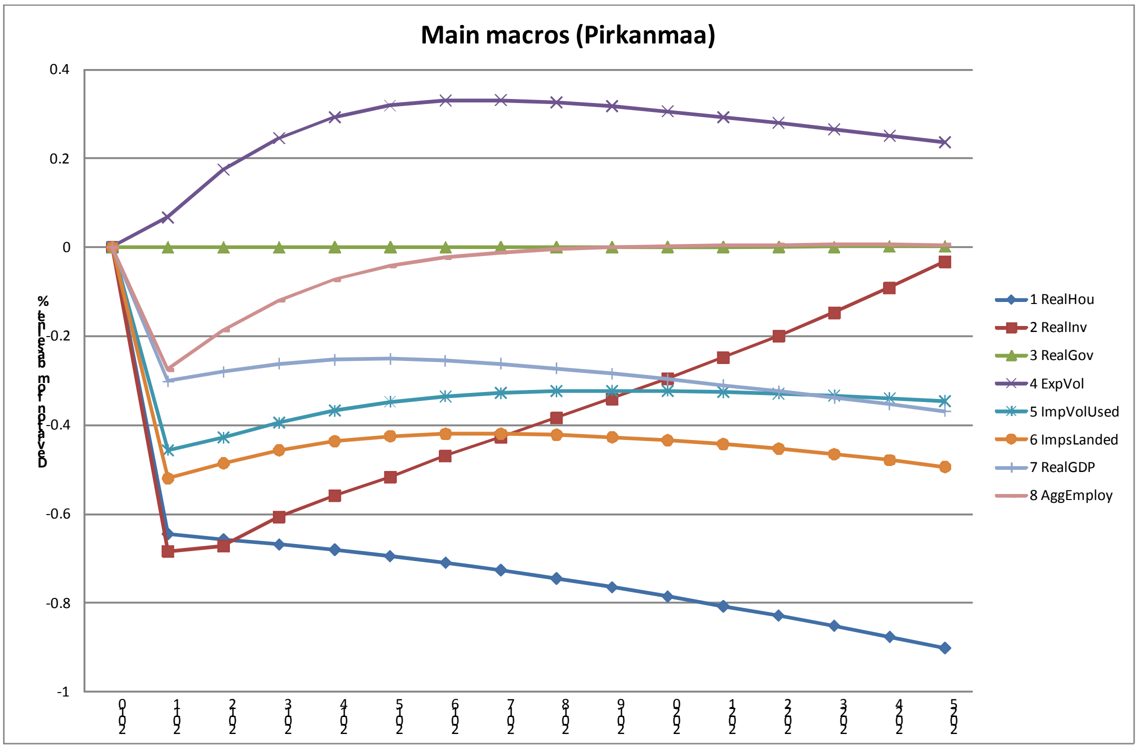


Figure 2.14 Expenditure-side contributions (Pirkanmaa)

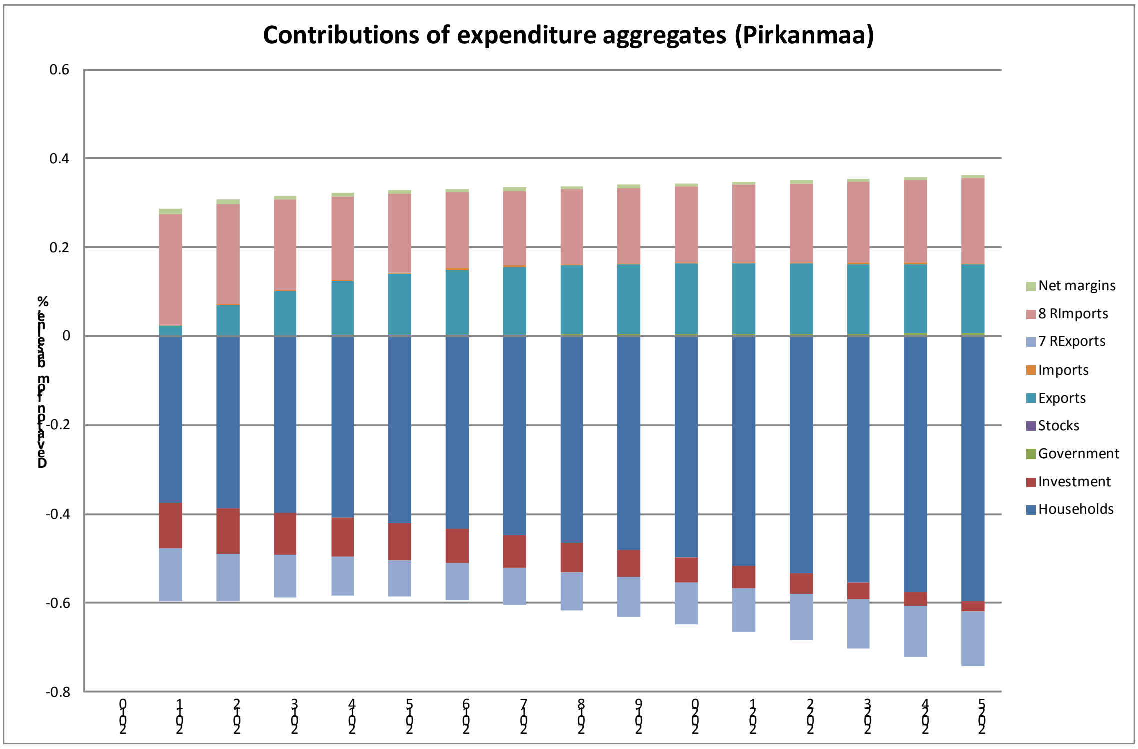


Figure 2.15 Regional main macros (Pohjois-Pohjanmaa)

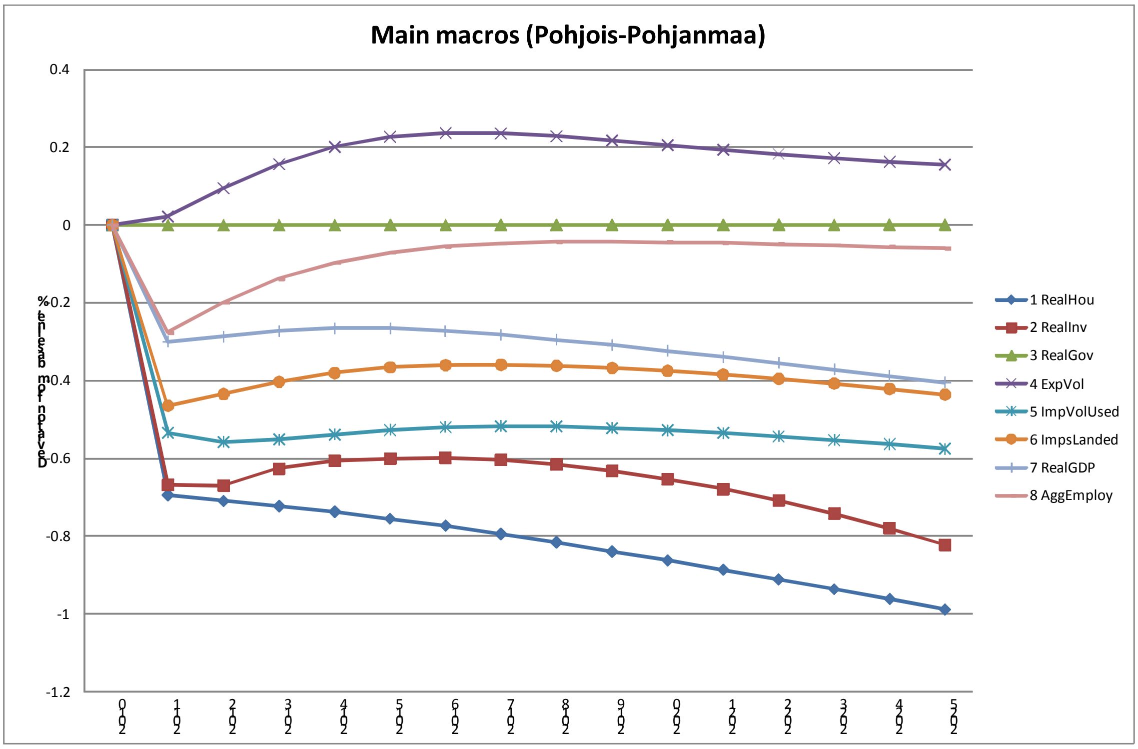


Figure 2.16 Expenditure-side contributions (Pohjois-Pohjanmaa)

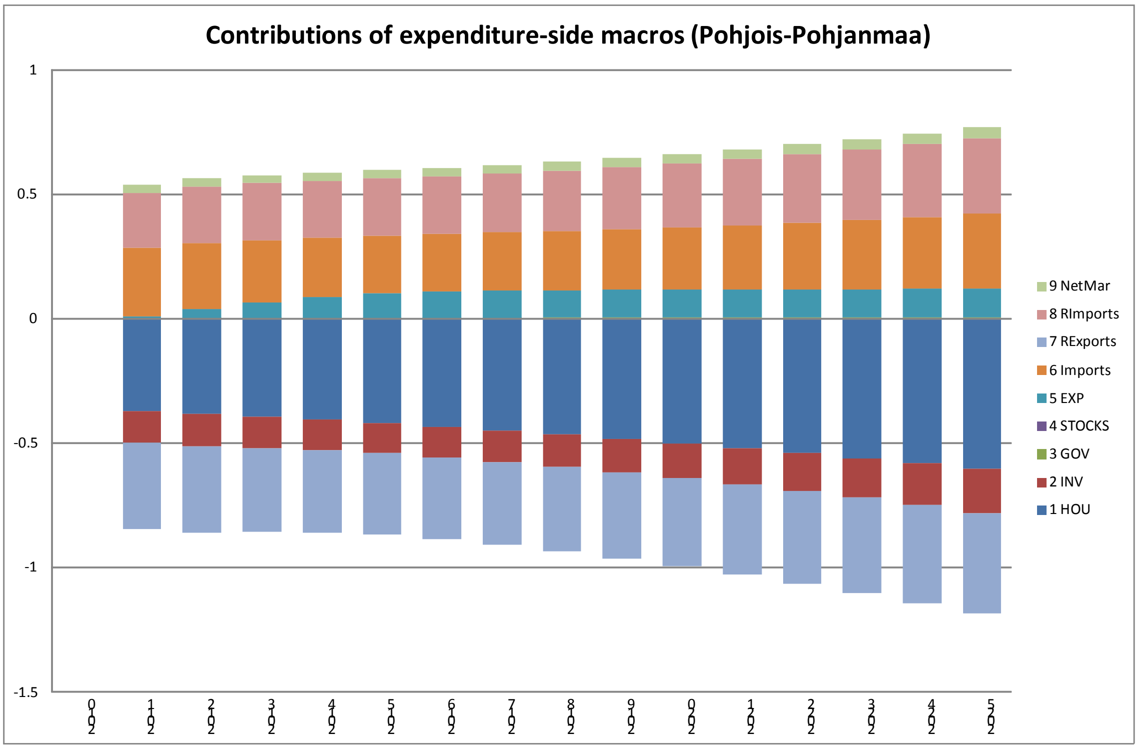


Figure 2.17 Regional main macros (Etelä-Savo)

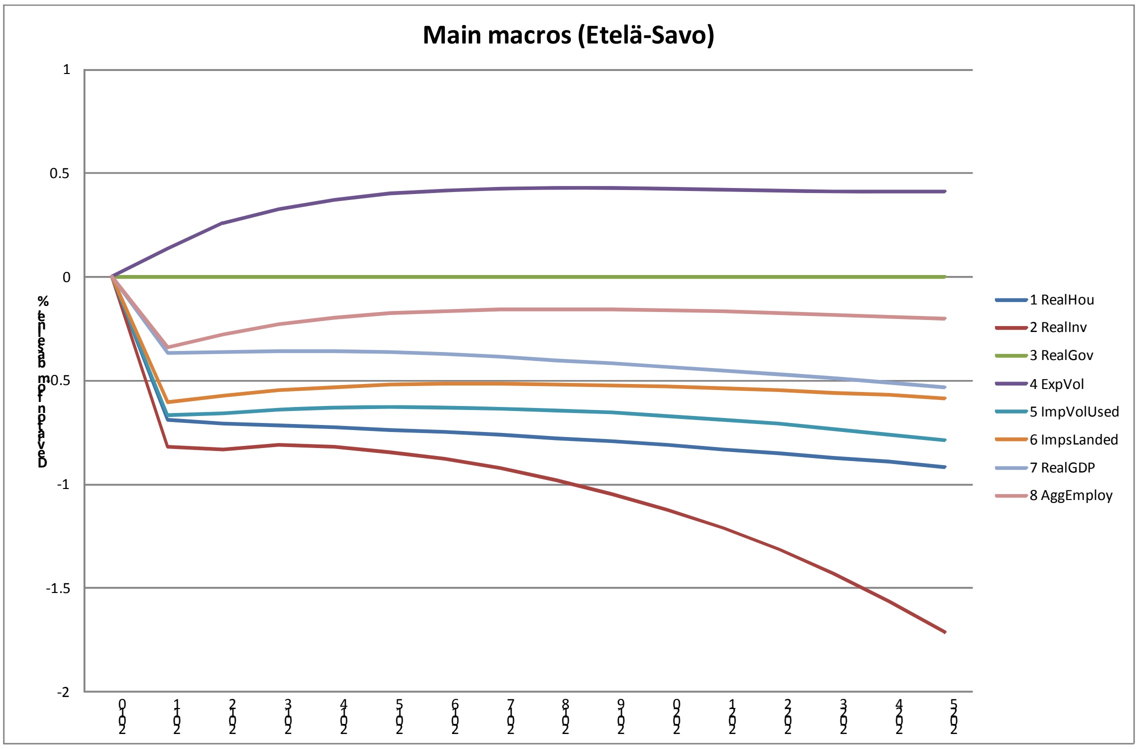


Figure 2.18 Expenditure-side contributions (Etelä-Savo)

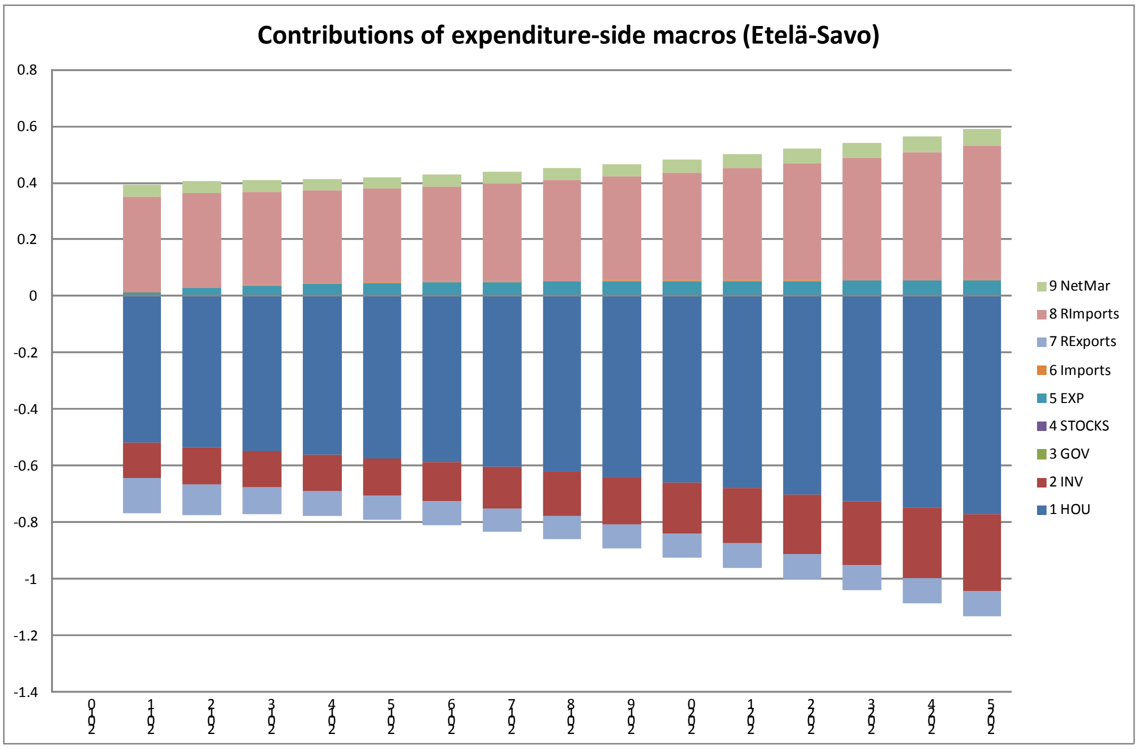


Figure 2.19 Change in regional demand for margins

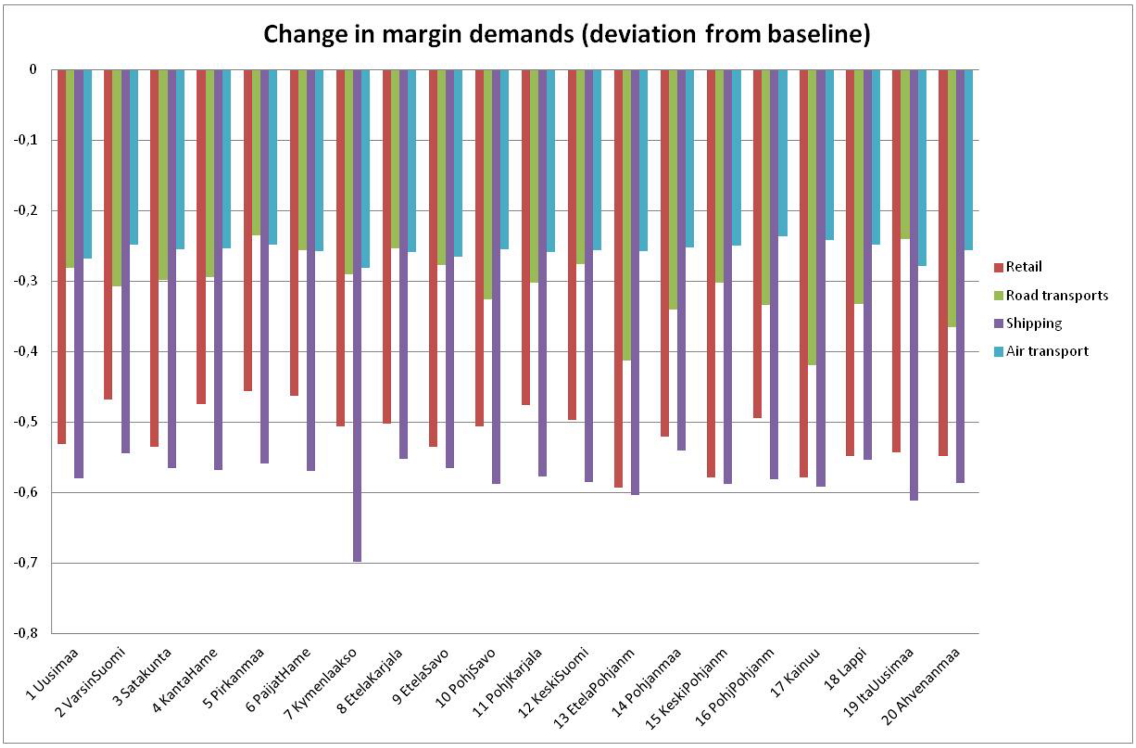


Figure 2.20 Shares of margins in regional CPI at purchasers'prices

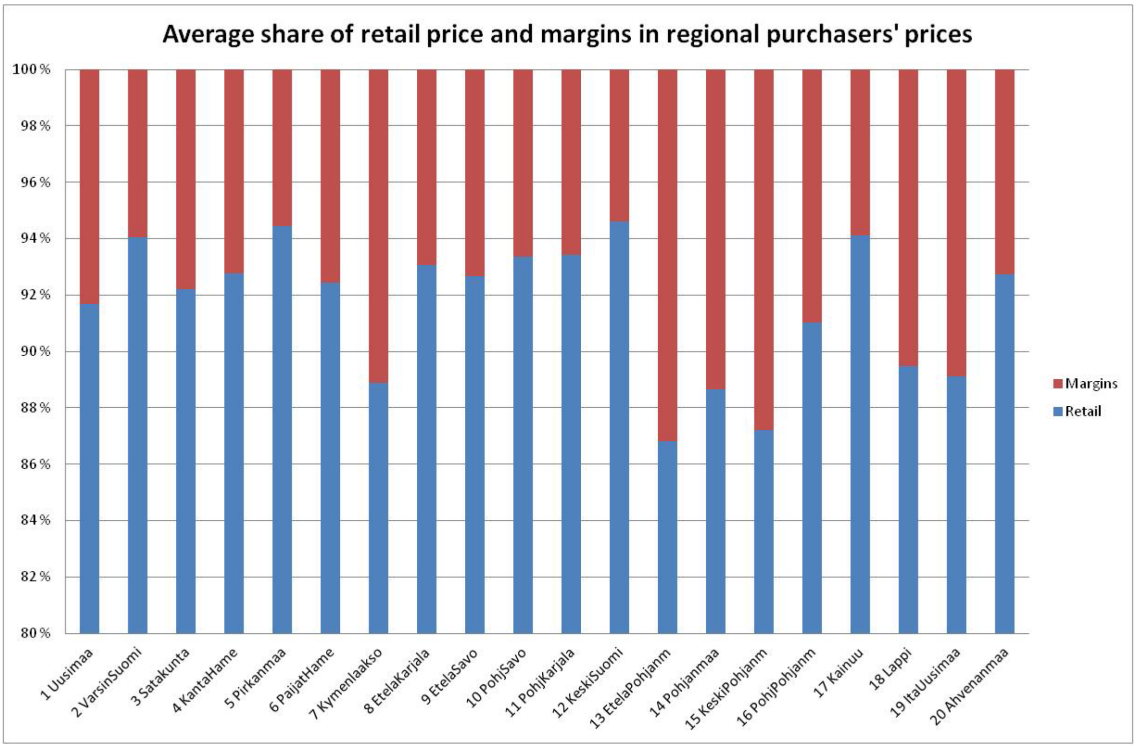
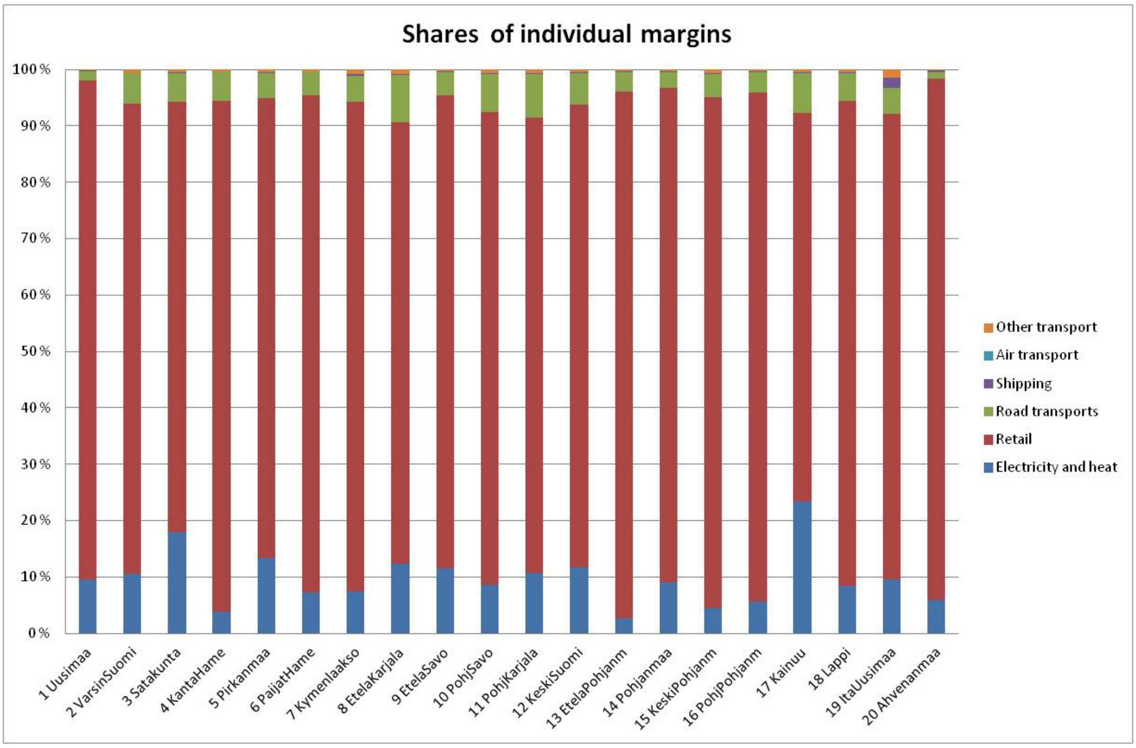


Figure 2.21 Shares of individual margins in household consumption



3. TABLO programming language and its use in the VERM

VERM computations are performed by the GEMPACK programmes. The GEMPACK suite of programmes consists of; text editors that are used for preparing the model and various command files for the simulations; of programmes used for processing the data; of the actual simulation software; and of various programmes used for viewing and analysing the simulation results. In this chapter, we give a brief overview of the GEMPACK package and the methodology of solving large models such as VERM.

However, we concentrate mostly in the TABLO programme, which is used for representing the VERM model itself. Although close to ordinary algebra, the TABLO language has a particular structure and some unavoidable idiosyncrasies in its vocabulary and syntax. A complete explanation is given in the GEMPACK manuals (Harrison and Pearson, 2002, 2005). This chapter gives a general outline of the structure of the VERM model code and of the notation and TABLO commands used in the code with a view of enhancing the tractability of the full VERM programme code in the following chapter.

The chapter is divided into three sections. Section 3.1 places TABLO in the context of the suite of GEMPACK programmes used in VERM computations. It also deals with the concept of closures, showing how the model can be used for studying history, for forecasting, and for policy analysis. Section 3.2 contains notes on TABLO vocabulary and syntax and on the conventions followed in VERM. The final section gives an overview of the structure of the TABLO presentation of VERM and examples of the use of TABLO in implementing the model.

3.1 Overview of the GEMPACK computations for the VERM model

3.1.1 GEMPACK solutions

GEMPACK performs model computations as a sequence of solutions of linear systems of the form

$$A(\overline{Z})z = 0 \tag{3.1}$$

where

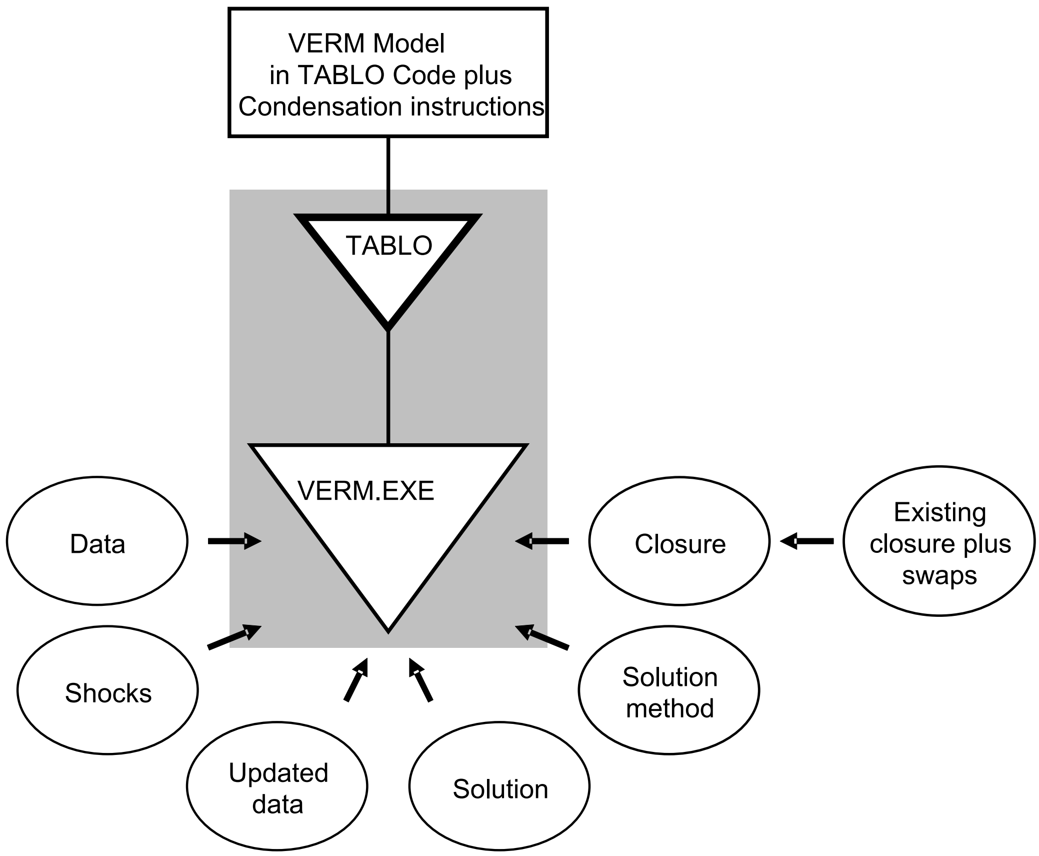
$A(\overline{Z})$ is a matrix of coefficients (e.g., cost and sales shares) evaluated at a solution $\overline{Z}$ of the model and

z is the vector of deviations in the model's variables away from $\overline{Z}$. Thus, while the theoretical models underlying VERM are non-linear, for the purposes of simulation, they are explicitly linearised in their TABLO implementation.

The sequence of solutions may be single- or multi-step computations for a single year, or for a series of years. At each step in the sequence, the A matrix is evaluated at a different vector $\overline{Z}$. The vector ($\overline{Z}(q)$) used in the qth step (q>1) is usually derived substantially from the solution of (3.1) obtained in the (q–1)th step and the vector ($\overline{Z}(1)$) is derived from data. The methodology of solving linear systems in several steps is related to the accuracy of the solutions and is explained in more detail in the next section.

Figure 3.1 is a flow diagram showing how GEMPACK creates and solves the sequence of linked equation systems (3.1).

Figure 3.1 GEMPACK solution of VERM



The GEMPACK TABLO programme creates an executable image of VERM, VERM.EXE, by operating on the TABLO representation given in chapter 4. TABLO also processes condensation instructions to reduce the model to a manageable size. An example of condensation instructions is given in section 3.1.3, where we also discuss model closures more thoroughly.

The executable image programme VERM.EXE uses several inputs to perform the simulation. These include the data files, the command files determining the policy shocks, choice of solution method, and the closure of the model.

The data is used to evaluate the coefficients in the A matrix. It is prepared with the help of the data processing programmes in the GEMPACK suite and contains all the information about the economy arranged in several data files. These data files are described more thoroughly in the next section.

There are several options for the solution method, ranging from single-step Euler to various more elaborate methods such as Gragg or Johansen/Euler. Finally, the values of the shocks are given in separate command files. Changes in policies or technology are referred as shocks and they are mostly given as deviations of the exogenous variables from their initial or baseline values.

With the data, closure, shocks and solution methods specified, VERM.EXE produces the solution showing the effects on the endogenous variables of the shocks to the exogenous variables. It also produces updated data files. These have identical format to the data input files and reflect the post-solution situation. For example, the input-output flows in the updated file are those from the input file altered by the changes in prices and quantities which form part of the solution. The updated files from the (q–1)th step in a sequence of solutions of (3.1) normally become the input data for the qth step.

3.1.2 The Percentage-Change Approach to Model Solution

GEMPACK solves VERM 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 solutions (see Dixon et. al. 1992).

The solution of the model can be represented in the levels as:

$$\mathbf{F}(\mathbf{Y},\mathbf{X}) = \mathbf{0}, \tag{3.2}$$

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 **Y**. The linearised approach starts by recognising that we already possess some solution to the system, $\{\mathbf{Y}^0,\mathbf{X}^0\}$, i.e.,

$$\mathbf{F}(\mathbf{Y}^0,\mathbf{X}^0) = \mathbf{0}. \qquad (3.3)$$

Normally the initial solution $\{\mathbf{Y}^0,\mathbf{X}^0\}$ is drawn from an historical input-output database combined with supplementary historical data – we assume that our equation system was true for some point in the past. With conventional assumptions about the form of the **F** function it will be true that for small changes **dY** and **dX**:

$$\mathbf{F}_{Y}(\mathbf{Y},\mathbf{X})\mathbf{dY} + \mathbf{F}_{X}(\mathbf{Y},\mathbf{X})\mathbf{dX} = \mathbf{0}, \qquad (3.4)$$

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

$$y = 100dY/Y \quad \text{and} \quad x = 100dX/X. \qquad (3.5)$$

Correspondingly, we define:

$$G_Y(\mathbf{Y},\mathbf{X}) = F_Y(\mathbf{Y},\mathbf{X})\mathbf{Y},^{\wedge} \quad \text{and} \quad G_X(\mathbf{Y},\mathbf{X}) = F_X(\mathbf{Y},\mathbf{X})\mathbf{X},^{\wedge}, \qquad (3.6)$$

where **Y,**^ and **X,**^ are diagonal matrices. Hence the linearised system becomes:

$$\mathbf{G}_Y(\mathbf{Y},\mathbf{X})\mathbf{y} + \mathbf{G}_X(\mathbf{Y},\mathbf{X})\mathbf{x} = \mathbf{0}. \qquad (3.7)$$

Such systems are easy for computers to solve, using standard techniques of linear algebra. But they are accurate only for small changes in **Y** and **X**. Otherwise, linearisation error may occur. The error is illustrated by Figure 3.2, which shows how some endogenous variable Y changes as an exogenous variable X moves from X0 to XF. The true, non-linear relation between X and Y is shown as a curve. The linear, or first-order, approximation:

$$y = -\,\mathbf{G}_Y(\mathbf{Y},\mathbf{X})^{-1}\mathbf{G}_X(\mathbf{Y},\mathbf{X})x \qquad (3.8)$$

leads to the Johansen estimate YJ – an approximation to the true answer, Y exact.

Figure 3.2 Linearisation error in a single-step process

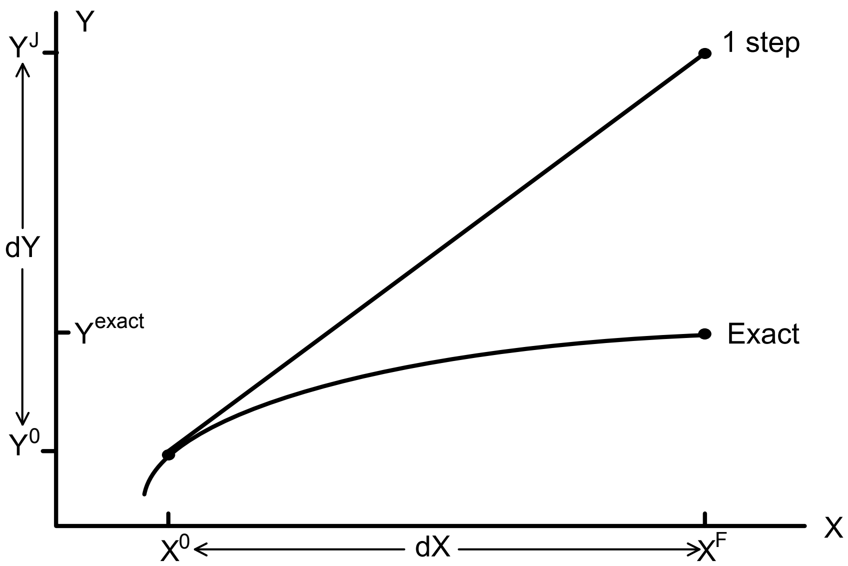


Figure 3.2 suggests that, the larger x is, the greater is the proportional error in y. This observation leads to the idea of breaking large changes in X into a number of steps, as shown in Figure 3.3. For each sub-change in X, we use the linear approximation to derive the consequent sub-change in Y. 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 3.3), the final value of Y, Y3, is closer to Y exact than was the Johansen estimate YJ. 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.

The technique illustrated in Figure 3.3, 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 $\{\mathbf{Y}^0,\mathbf{X}^0\}$, formulae for the derivative matrices $\mathbf{G}_Y$ and $\mathbf{G}_X$, and the total percentage change in the exogenous variables, $\mathbf{x}$. The levels functional form, $\mathbf{F}(\mathbf{Y},\mathbf{X})$, need not be specified, although it underlies $\mathbf{G}_Y$ and $\mathbf{G}_X$.

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).

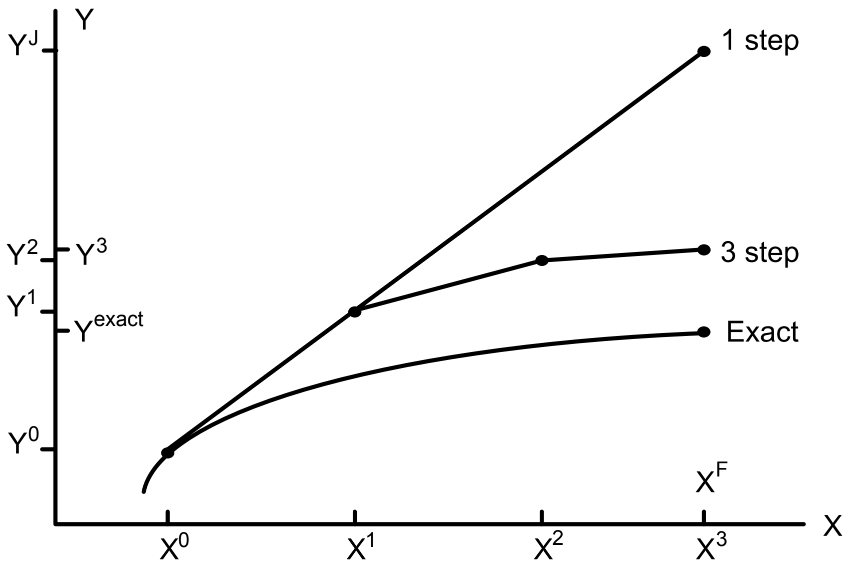
Extrapolation suggests that the 32-step solution would be:

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

and that the exact solution would be:

y(∞-step) = 4.1%.

Figure 3.3 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 (3.7); and use of that solution to update the data (**X**,**Y**).

In practice, for typical AGE models, it is unnecessary, during a multistep computation, to record values for every element in **X** and **Y**. Instead, we can define a set of *data coefficients* **V**, which are functions of **X** and **Y**, i.e., **V** = **H**(**X**,**Y**). Most elements of **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 **V**; often indeed identical to elements of **V**. After each small change, **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 **V**, rather than **X** and **Y**, are twofold:

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

3.1.3 Closures and condensation instructions

In dynamic mode, VERM contains hundreds of thousands of equations. It is not practical to solve directly equations systems of this size. The problem is made manageable in two ways: by omitting arrays of exogenous variables that are not shocked and by substituting out arrays of endogenous variables that are not of interest. The arrays that are targeted for these treatments typically have large numbers of components.

An example of an array that is usually omitted is

fa1mar(c,s,i,m) for $c\in$ COM, $s\in$ SOURCE, $i\in$ IND and $m\in$ MAR.

This is an array of technical change shifters concerned with the usage of margin commodity m to facilitate the flow of commodity c from source s to industry i for the purpose of current production. The array is occasionally useful in simulating technical changes but in most applications it is set exogenously on zero. Thus, in most applications it can be deleted. Alternatively, if fa1mar(c,s,i,m) is endogenous, it can be substituted out, which means that it is replaced by the variables determining it in equation

E_fa1mar(c,s,i,m) in the model code.

The concept of closure is central to simulations performed with GEMPACK. By closure we mean the specification of variables as exogenous and endogenous. The need for a closure specification arises from several reasons. First, VERM does not contain explicit equations for all of its variables. For example, shifts in technology or tastes are typically treated as exogenous. Secondly, closure changes provide a practical way of modifying the model to suit to specific applications.

The concept of closure can be illustrated by considering the solution to the model from a slightly different point of view than that in equation 3.8. For each year, the solution takes the form

$$\mathrm{F}(X) = 0 \tag{3.9}$$

where F is an m-vector of differentiable functions of n variables X, and $n>m$. The variables X include prices and quantities applying for a given year and the m

equations in (3.9) impose the usual AGE conditions such as: demands equal supplies; demands and supplies reflect utility and profit maximising behaviour; prices equal unit costs; and end-year capital stocks equal depreciated opening capital stocks plus investment. It is important to realise that there always exists a solution ($X_{initial}$) of (3.8) derived mainly from input-output data for a particular year. In simulations we compute the movements in *m* variables (the endogenous variables) away from their values in the initial solution caused by movements in the remaining $n - m$ variables (the exogenous variables) away from their values in the initial solution. By closure we mean the division of the model's variables into endogenous and exogenous. There is no single way to do this. Instead, the closure depends largely on the application.

As MONASH-type dynamic models usually do, so does VERM recognise four types of closures:

- decomposition closure,
- historical closure,
- forecasting closure and
- policy closure.

In a decomposition closure, we include in the exogenous set all naturally exogenous variables, that is variables not normally explained in a AGE model. These may be observable variables, such as tax rates, or unobservables, such as technology and preference variables.

Historical closures include in their exogenous set two types of variables: observables and assignables. Observables are those for which movements can be readily observed from statistical sources for the period of interest. Historical closures vary between applications depending on data availability but typically include a wide array of macro and industry variables, as well as intermediate input flows between industries.

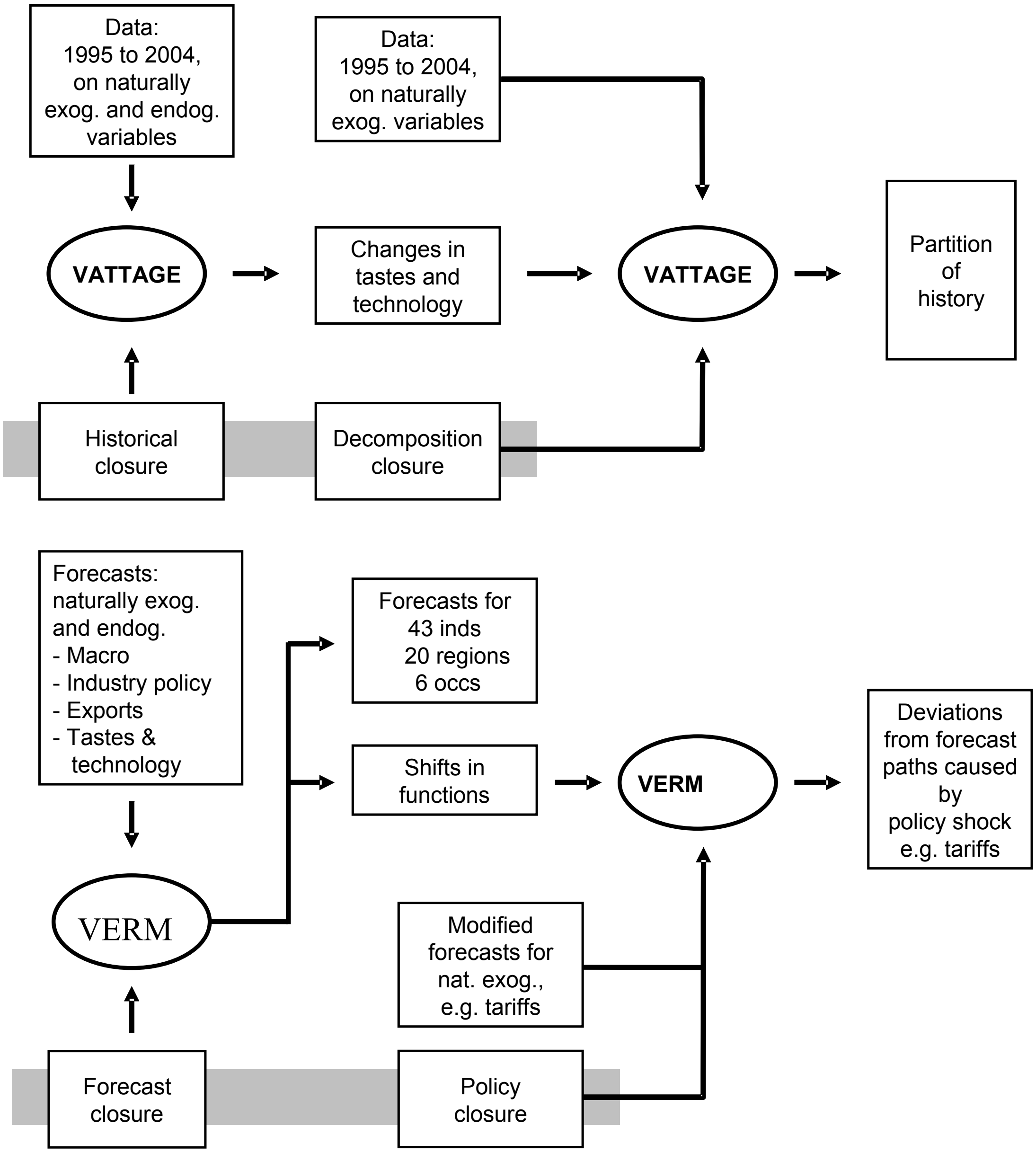
Forecasting closures are close in philosophy to historical closures. Instead of exogenising everything that is known about the past, in forecasting closures we exogenise everything that we think we know about the future. Thus in forecasts, we exogenise numerous naturally endogenous variables, including, for example, export volumes (where outside forecasts or scenarios are available), and most macro variables (where medium and long term forecasts prepared by ministries or the EU can be used). To allow these variables to be exogenous, a number of naturally exogenous variables need to be endogenised, for example the positions of foreign demand curves, the positions of domestic export supply curves, and many macro coefficients such as the average propensity to consume.

Policy closures are similar to the decomposition closures. In policy closures naturally endogenous variables, such as exports and macro variables, are endogenous, since they must be allowed to respond to the policy change under consideration. Correspondingly, in policy closures naturally exogenous variables, such as the positions of foreign demand curves, the positions of domestic export supply curves and macroeconomic coefficients, are exogenous, and are set at the values that they have in the forecasts.

The relationship between forecasting and policy simulations is similar to that between historical and decomposition simulations. Historical simulations provide values for exogenous variables in corresponding decomposition simulations. Similarly, forecasting simulations provide values for exogenous variables in corresponding policy simulations. However there is one key difference between the relationships. An historical simulation and the corresponding decomposition simulation produce the same solution. This is because all the exogenous variables in the decomposition simulation have the values they had (either endogenously or exogenously) in the historical solution. In a policy simulation, most, but not all, of the exogenous variables have the values they had in the associated forecast solution. The policy variables of interest are set at values that are different from those they had in the forecasts. Thus policy simulations generate deviations from forecasts.

There is an important limitation stemming from the infrequent updates of regional data. While at the national level it is actually possible to use consecutive, yearly, supply and use tables (including input-output matrixes) to conduct a historical simulations, at the regional level, this is not feasible. Thus in practice VERM uses the detailed industry-level results from VATTAGE as a starting point for its forecasting/baseline simulations, as illustrated in figure 3.4 below. While this is a limitation, the approach does have advantages as well, chief among them the resulting consistency between national level and regional forecasts.

Figure 3.4 Historical and Decomposition Simulations



3.2 Introduction to TABLO syntax and conventions

While the TABLO syntax resembles many other simulation programmes, it does have certain specific rules and conventions that it is useful to be aware of. This section gives a summary of the most common ones. The examples in the next section demonstrate how TABLO handles data and parameters, while the following section gives some examples of equations expressed in TABLO syntax. The points we have chosen to list here will be helpful to readers when they are following the examples in the next two sections, and for looking for the first time at the TABLO code in chapter 4. Chapter 5 contains more specific examples of the expressions derived from the model's theoretical equations.

The TABLO code contains instructions for the handling of data and for performing actual numerical evaluations following the formulas and equations of the model. Instructions are expressed with key words. Often, the TABLO also contains comments and remarks that are not processed. These remarks and explanations come in two varieties. First, material in the TABLO code enclosed by exclamation marks (!) is not processed by TABLO. It merely provides explanatory comments. Second, material enclosed by cross-hatches (#) does not play a role in computations. However, it is recorded by TABLO and used in the labelling of various GEMPACK printouts.

Among the key words in the TABLO language are: File, Set, Coefficient, Read, Formula ,Variable, Update and Equation. These are followed by statements giving instruction for the programme. All TABLO statements end with a semicolon (;).

The key word File is used to assign logical names to the various data sources used in the computations. Examples of the use of File are given in section 3.3.1.

Key word Set is used to define the dimensions for the coefficients contained in the data. Sets can be either labelled explicitly in the programme or command files, or they can be read from data files.

The model data are handled as coefficients that are to be evaluated and updated. Coefficients must always be declared with the key word Coefficient, and they must always be assigned values either by instructing GEMPACK to use specific data sources for the evaluation of them (the key word Read) or to evaluate them with formulas using already evaluated coefficients, in which case the key word Formula is used. Coefficients are updated by the model's variables unless they stem from formulas.

The key word Variable declares a specific variable, whence the variable is explained in an equation unless it is exogenous. Key words need not always be repeated when they are applicable to an unbroken sequence of statements. For example, if we are declaring two variables, a and b, we can write

```
Variable a          # example variable # ;
Variable b          # another example variable # ;
```

Alternatively we can write

```
Variable
a # example variable # ;
b # another example variable # ;
```

TABLO contains several devices that can be used for exception handling. For example, in evaluating coefficients, it is often the case that a vector of values may contain some elements with a zero value. TABLO includes a device which is sometimes convenient for avoiding zero-divide difficulties. This is illustrated by the following TABLO code:

```
Zerodivide Default 1.0  ;
          Formula
          A = B/C  ;
          E = F/G  ;
Zerodivide off  ;   .
```

If C happens to be zero, then A is evaluated as 1.0. Similarly, if a division by zero occurs on the RHS of any other formula listed before the command Zerodivide off, then the LHS of the formula is evaluated as 1.0. Thus if G is zero then E is 1.0. Another approach used in TABLO code to deal with potential divisions by zero is to add the coefficient TINY to denominators. TINY is set at a very small number, 10^{-12}.

TABLO restricts the lengths of names and of comments contained between cross-hatches. This requires the use of abbreviations in the TABLO code. Furthermore, TABLO does not allow Greek letters or subscripts and superscripts.

TABLO does not distinguish between upper- and lower-case letters. To enhance the readability of the TABLO representation of VERM, a useful convention is to use lower-case letters for variable names and upper-case letters for the names of coefficients, sets and files. For TABLO key words, we the first letter, e.g., Sum, Equation, Update, etc. is often capitalised.

3.3 Overview of the structure of the TABLO representation of VERM

The TABLO code in chapter 4 is concerned with a single step in the sequence of solutions of (3.1).

It specifies rules: for reading the data input (such as input-output flows and substitution parameters); for forming the equation system (3.1); and for revising the data in preparation for the next step in the sequence of solutions. Here, we give examples of typical TABLO usage as they appear in VERM code.

3.3.1 Data files

The VERM model code in chapter 4 starts by indicating that the data inputs needed for creating the system (3.1) are drawn from nine files, with logical names BASEDATA, EXTRA, ..., ROREXT. In implementing the TABLO representation of VERM we must specify the specific files in our computer to play the roles of BASEDATA, EXTRA, etc. The contents of these specific files are indicated in by comments between cross-hatches (#).

The model code also lists a data output file (New) with the logical name SUMMARY. The specific file which plays the role specified in the TABLO code for SUMMARY collects information useful in checking and explaining results. Implicit in the TABLO code are instructions for the creation of several other output files.

The contents of the most important data file, the BASEDATA, are illustrated in figure 3.5.

The figure shows how the use of goods and services is distributed within industries and institutional sectors. As explained earlier, the basic structure in VERM consists of the general equilibrium theory that underlies the demand structure of the model, and theory explaining the dynamic evolvement of the economy over time. The theory behind the model is reflected in the database. The database shown in figure 3.5 identifies the following agents in the model economy:

- domestic producers divided into I industries;
- investors divided into I industries;
- regional, representative households;
- foreign purchasers of exports;
- foreign sources of imports;
- demand categories corresponding to local and central government demands, as well as demand by social security funds.
- C commodities, stemming from 2 sources
- O occupational groups

In naming variables and coefficients, we use almost invariably the following conventions: 0 indicates production; 1 indicates intermediate usage; 2 indicates demand for use in capital creation; 3 indicates demands by households; 4 indicates exports; 5 indicates government demands; and 6 indicates demands for inventories. The letters c, s and i are often used to indicate commodity, source and industry. For example, x1(c,s,i) refers to intermediate usage distinguished by commodity, source and industry.

The exception to the simple notational convention concerns the bottom-level flows of goods, where we follow the compact notation stemming from Mark Horridge's TERM model, where trade flows are denoted by `xtrad(c,s,r,d)` - where the dimensions are commodity by source by region of origin by region of use. This notation is also adapted for the related price variables.

In figure 3.5 the rows show the structure of the purchases made by each of the agents identified in the columns. Each of the c commodity types identified in the model can be obtained from domestic producers 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 and are exported. Commodities stem from either a domestic source, or from abroad. Margins are treated as margins goods (M). Intermediate use of goods is presented by using industry (I). Finally, household consumption, public consumptions, and inventories are all covered by single row entries.

Figure 3.5 The combined supply and use matrix

		Supply and use matrix					
		1	2	3	4	5	6
		Producers	Investors	Households	Export	Government	Stocks
	Size	□□□□I□□ □□	□□□□I□□ □□	□□1□□□□	□□□□ V□□□□□	□□□1□□	□□□1□□
Basic Flows	CxS	V1BAS	V2BAS	V3BAS	V4BAS	V5BAS	V6BAS
Margins	CxSxM	V1MAR	V2MAR	V3MAR	V4MAR	V5MAR	n/a
Taxes	CxS	V1TAX	V2TAX	V3TAX	V4TAX	V5TAX	n/a
Labour	O	V1LAB					
Capital	1	V1CAP					
Land	1	V1LND					
Production tTx	1	V1PTX					
Other Cost	1	V1OCT					

C=Number of Commodities
I=Number of Industries

S=3: Domestic, Imported from the EU, Imported from the Rest of the World
O=Number of Occupations

M=Number of Commodities used as Margins
V=2: Export to the EU, Export to the Rest of the World

	Make Matrix
Size	□□□□□□□□□□I□ □□□□□□□□□
C	MAKE

	Import Duty
Size	□□□□□□1□□□ □□□□
C	V0TAR

The other data files contain data on the public sector – such as taxes and transfers – as well as the balance of payments. There are also several files that are used for inputs in dynamic simulations. These files contain data on capital stocks, rates of return and the like. The reason for several files stems from the need to keep track of timing. While there are no restrictions in TABLO governing the number of data files or the distribution of data across them, the interdepencies of data elements do have a practical effect on the division of data inputs. For example, if data items X and Y are both in the same input file Z, then we are limited to solutions of $\overline{v}$ that reflect values of X and Y arising from the process which generates the file Z. If this process imposes a link between X and Y which should not apply at each step in the sequence of solutions, then X and Y should be in different files. For example, VERM computations with forward-looking expectations involve a sequence of solutions in which the values of rates of return used in forming $\overline{v}_q$, q>1, are derived from the solution in the (q–1)th step. On the other hand, some of the data, for example, rates of return, used in the qth step are not always generated in the (q-1)th step. Consequently, rates of return are placed in a different file from the main data, thus allowing the data to be independent of the data for rates of return.

3.3.2 Sets and subsets

The next section of the TABLO code first declares in alphabetical order the names of most sets and either specifies their contents or indicates where these can be found. For example, SOURCE is declared explicitly as a set consisting of three objects, "dom", "eu" and "non_eu"; in contrast, the set COM is a set consisting of objects whose names will be found in a section of the BASEDATA file labelled "COM".

The set declarations give a good overview of the objects with which VERM is concerned. For example, these include commodities (COM) produced by the domestic industries (IND) or imported from foreign sources. The set declarations are followed by statements indicating that some sets are subsets of others. For example, TRADEXP is a subset of COM. Via this subset declaration, TABLO knows that the object "C_01" in TRADEXP is the same object as "C_01" in COM.

The main sets defined in VERM are:

COM	Commodities
DST	Regions of use
IND	Industries
MAR	Margin commodities (a subset of COM)
OCC	Occupations
ORG	Regions of origin
PRD	Regions of margin production
FUEL	Energy commodities (a subset of COM)
DEST	Destinations of goods

LOCUSER	Domestic destinations of goods (a subset of DEST)
PSEC	Government sectors (Central,Local,Social security fund)
SRC	Source of commodities

3.3.3 Coefficients

Section 3 of VERM code contains declarations for the coefficients in the model. Coefficients are the main building blocks in the construction of $A(\overline{V})$ in equation 3.1. Apart from numerous zeros and ones, and a few other numbers, the components of $A(\overline{V})$ are coefficients or functions of coefficients.

For each coefficient, the TABLO code gives evaluation instructions. Evaluation can either be from a data file via a read statement, or in terms of other coefficients via a formula. Some formulae are implemented in every step of a multi-step computation whereas others are implemented only in the first step for each year.

For many of the coefficients evaluated via read statements, the TABLO code contains update instructions, i.e., instructions on how the post-solution value of the coefficient should be computed. Where an update instruction is provided, we have included a U in the cross-hatched comments. The update statements are in section 7 of the code and are discussed in subsection 3.3.6. below.

To finish the present subsection, we give a few examples that may be helpful in clarifying the meaning of the coefficient declarations.

Example 1

AGGCAP is interpreted in our economic theory as the total payments to capital. It is evaluated as a function of coefficients according to a formula given in subsection 5.10 of the code. Because in each step of a sequence of solutions of (3.1) the value of AGGCAP is obtained via a formula, AGGCAP is not updated.

Example 2

USE(

```
(all,c,COM)(all,s,SRC)(all,d,DST)
   USE(c,s,"hou",d)          =          x3(c,s,d)*puse(c,s,d);
```

USE(c,s,"hou",d) is interpreted in our economic theory as the basic value of the flow of good c from source s for use in consumption by the household in region d.

Example 3

SIGMA1(c) is interpreted in our economic theory as the elasticity of substitution between domestic and imported good c for use in current production. In TABLO, SIGMA1 is a vector taking values for all c in the set COM. Values for SIGMA1 are obtained from data accessed via a Read statement. No update instructions are given. Import/domestic substitution elasticities are usually held constant throughout a sequence of solutions of (3.1), i.e., they are treated as parameters.

3.3.4 Read statements and formulas

As already mentioned, Section 4 of the VERM code contains instructions for the evaluation of coefficients from data files. The command Read, used throughout the code instructs the programme to access numerical data. By contrast, the command Read Elements, instructs post-TABLO programmes to access alpha-numeric information.

If a coefficient is determined by a formula, the instructions for evaluating the coefficient are given after the key word Formula. Contrary to the declarations for coefficients and variables, formulas cannot be arranged in a simple order, such as alphabetical. This is because no coefficient can appear on the RHS of a formula unless instructions for its evaluation have appeared earlier in the TABLO code. Thus, if coefficient A is to be set equal to coefficient B, and B is to be set equal to 10, we cannot use the alphabetical ordering:

```
Formula  A = B ;
Formula  B = 10 ;     .
```

Instead we must write

```
Formula  B = 10 ;
Formula  A = B ;     .
```

The TABLO code allows a wide range of operators to be used in formula statements. The meaning of most of these is clear. Here we interpret a few of the formulas from section 5 in the code to give a flair of some of the operators commonly used in formulas.

Example 1

This example shows how the keyword Initial can be used. The example stems from subsection 5.18 of VERM code and gives instructions for evaluating the maximum growth rate of the capital stock in industry i:

```
Formula
  (initial)(all,i,IND)(all,d,DST)
   K_GR_MAX(i,d)=TREND_K(i,d)+DIFF+if(QCAPATT(i,d)<=0.00001,1.0);
```

In the formula, the growth rate is given a value that is the sum of the trend growth rate of the capital stock in industry i, a difference parameter, and, if there is no capital in the industry, the value 1. The word Initial in parentheses indicates that in a multi-step solution for year t, the coefficient K_GR_MAX(i), the maximum growth rate of the capital stock in the investment function, is evaluated via the formula above only in the first step. Provided there are no further instructions in our TABLO code regarding its evaluation, the coefficient will retain this first-step value throughout the remaining steps for year t.

The Initial option is particularly valuable in the modelling of processes involving lags. Examples in VERM can be found in several sections, including 5.7, as well as section 5.8, which deals with sticky real wages. With lags, we often need to hold a coefficient in the year t computation at a value reflecting the solution for year t–1. This can be achieved via the Initial option provided the initial solution for year t is the final (required) solution for year t–1.

Example 2

The next example show how a parameter can be directly defined. This example is from subsection 5.3 of the model code:

```
Formula TINY = 0.000000000001 ; .
```

TINY, which is permanently set at 10^{-12}, is often used to avoid division by zero or the occurrence of an endogenous variable with a zero coefficient in all equations.

Example 3

The next example shows usage of TINY in a formula that evaluates export shares of commodity c to destination d. The example is from subsection 5.13 of VERM code:

```
Formula
  (all,c,COM)(all,d,XDEST)
   EXPSHR(c,d) = V4BAS(c,d)/[MAKE_I(c)+TINY];
(all,c,COM)(all,r,REG)        EXPSHR(c,r)        =
ROWDEM_D(c,r)/[0.001+MAKE_I(c,r)];
```

In the formula, V4BAS is the basic flow of exports of commodity c to destination d and MAKE is the total production of commodity c.

3.3.5 Variables

In TABLO, variables have to be declared before they can be used. The declarations give names and dimensions to the components of v in (3.1), or, equivalently, they give names and widths to collections of columns of $A(\overline{V})$.

Most of the variables are interpreted in our economic theory as percentage changes. For example, p1cap(i) is the percentage change, away from its value in $\overline{V}$, of payments to capital in industry i.

However, many variables are changes. For example, d_govsav is the change in the (level of) government saving, not the percentage change. For variables that are to be interpreted as changes, TABLO requires us to include in their declarations the word Change in parentheses. We have also elected to give change variables names starting with either d_ or del_.

Failure to distinguish in the variable declarations between change and percentage-change variables will not affect single-step solutions for year t. Difficulties arise, however, in the calculation of results from multi-step solutions. If the (Change) instruction is omitted from the equation for d_eror(I,d), the change in expected returns to capital, given by

```
(all,i,IND)(all,d,DST) d_eror(i,d) =  d_ror_se(i,d) + d_ff(i,d);
```

then, in a two-step calculation, we would obtain:

$$d\_ eror(i,d)_{result2} = 100*\{(1 + d\_ eror(i,d)_1/100)(1 + d\_ eror(i,d)_2/100) - 1\},$$

where $d\_eror(i,d)_{result2}$ is the result for the two-step computation and $d\_eror(i,d)_1$ and $d\_ eror(i,d)_2$ are the solutions in steps 1 and 2. With d_ eror(i,d) declared as a change variable, we obtain the correct two-step result; i.e.,

$$d\_eror(i,d)_{result2} = d\_ eror(i,d)_1 + d\_ eror(i,d)_2 .$$

The following conventions are used (as far as possible) in naming variables. Names consist of a prefix, a main user number and a source dimension. The prefixes are:

a ⇔ technological change, change in preferences;

d_, del_ ⇔ ordinary (rather than percentage) change;

f ⇔ shift variable;

p ⇔ prices;

x ⇔ quantity demanded;

q ⇔ quantity supplied.

The main user numbers are the same as in the database, namely:

1 ⇔ firms, current production;

2 ⇔ firms, capital creation;

3 ⇔ households;

4 ⇔ foreign exports;

5 ⇔ government;

The number 0 is also used to denote basic prices and values. The source dimensions are:

s ⇔ all sources, (i.e., 1 domestic and 2 foreign);

Variable names may also include an (optional) suffix description. These are:

cap ⇔ capital;

imp ⇔ imports;

lab ⇔ labour;

lnd ⇔ agricultural land;

lux ⇔ linear expenditure system (supernumerary part);

mar ⇔ margins;

oct ⇔ other cost tickets;

prim ⇔ all primary factors (land, labour or capital);

sub ⇔ linear expenditure system (subsistence part).

Variables that are not related to production, for example, variables that are used to describe the public sector and its policy instruments such as direct taxes, assets and benefits, are given names that bear a resemblance to their true meaning. For example, changes in age benefits are given by a variable called age_ben, whereas changes in the income tax rate for labour income is given by a variable called tax_l_rate.

3.3.6 Update statements

Earlier in this section, we saw that VERM takes data from nine files with logical names BASEDATA, EXTRA etc. For each of these data input files, the TABLO code contains implicit (hidden from the GEMPACK user) instructions for the creation of ten post-solution files with logical names: updated BASEDATA; updated EXTRA, etc. The implicit TABLO instructions require that the updated files have structures identical to those of the corresponding input files, i.e., identical header (location) names and data arrangements. The data values will also be identical except where explicit instructions are given for their alteration.

Update instructions are given by Update statements, which are collected in alphabetical in order in section 7 of VERM code. Here we provide a few illustrative interpretations.

Example 1

This example shows how an ordinary change variable updates a flow, in this case, a flow of indirect tax revenue from consumption:

```
(change)(all,c,COM)(all,s,SRC)(all,d,DST)
   TAX(c,s,"hou",d)          =          delTAXhou(c,s,d);
```
(3.10)

In our current example, the RHS of (3.10) is the change that should be made to the value of the coefficient TAX(c,s,"hou",d), to derive its post-qth-solution value.

The post-qth-solution value of TAX(c,s,I,d) is given by

$$TAX(c,s,\text{"hou"},d)_{post\_q} = TAX(c,s,\text{"hou"})_q + RHS(3.10)_q$$

Because $TAX(c,s,i,d)_q$ is read from the header "TAX" in the BASEDATA file, TABLO's implicit instructions mean that $TAX(c,s,i,d)_{post\_q}$ will be stored in the corresponding position in the header designated "TAX" in the updated BASEDATA file.

Example 2

An example of a percentage change variable is used for updating a coefficient is given by

```
Update LEV_CPI = p3tot  ;   .
```

which means that the post-qth-solution value of LEV_CPI (the level of consumer prices) is given by

$$LEV\_CPI_{post\_q} = LEV\_CPI_{q}*(1+p3tot_q/100),$$

where p3tot is the percentage change in consumer prices.

Example 3

An example, where two percentage change variables are used to update a coefficient is

```
Update (All,i,IND)   V1CAP_(i) = p1cap(i)*x1cap(i);
```
(3.11)

The absence of either (Change) or (Explicit) means that (3.11) is in short-hand. It implies that the post-qth-solution value of V1CAP(i) is given by

$$V1CAP(i)_{post\_q} = V1CAP(i)_q\ (1 + p1cap(i)_q/100 + x1cap(i)_q/100)\ \ .$$

Because the qth-step value of V1cap(i) is read from file BASEDATA Header "1CAP", the post-qth-solution value is stored in the corresponding position in updated file BASEDATA Header "1CAP".

In the economic interpretation of the TABLO code, V1CAP(i) is the rental on capital in industry j. This is the product of the rental price and the quantity of capital in industry i.

3.3.7 Display and Write statements

Display and Write commands can be used to name specific coefficients to be reported. Their values are written to a Display file which is automatically created (no logical name need be specified) and labelled in a user-friendly manner by the GEMPACK programmes. The purpose of the Display file is to provide information for checking and interpreting results. The second list of coefficients are written to files with logical names nominated by the model-builder in the TABLO code. For VERM we have only one Write file with logical name SUMMARY. The labelling in Write files is less user friendly than that in Display files. However Write files are in a suitable form to be processed by other GEMPACK programmes.

In a simulation for year t, the values that appear in Display and Write files are the values of coefficients reflecting the execution of reads and formulas in the first step of the computation up to the point where the Display or Write command occurs. These values are usually those used in forming the first step's $A(\overline{V})$ matrix. An exception occurs when we Display or Write the value of a coefficient and then subsequently alter this value via a formula occurring after the Display or Write command.

3.3.8 Equations

TABLO requires that each equation has a name. Unlike formulas, equations can be listed in any order. Unlike sets, coefficients, reads, variables and updates, equations cannot be listed in an order which is both mechanical (e.g. alphabetical) and useful. VERM equations are arranged in thematic groups, which are:

- producer's demands for produced inputs and primary factors;
- producer’s demands for inputs to capital creation;
- household demands;
- export demands;
- government demands;
- demands for margins;
- market-clearing conditions for commodities and primary factors; and
- zero pure profits in production and distribution;
- indirect taxes;
- macroeconomic variables and price indices.
- investment dynamics

- labour market dynamics
- government accounts
- balance of payments
- income and saving aggregates
- miscellaneous variables
- reporting variables.

In VERM, most equations are named following the convention that the name of an equation reflects that of the variable it determines in a standard long run closure. In other words, we use names of the form E_xx...x where xx...x is the name of the variable determined by the equation. While the idea that each equation determines a particular variable is not precise, as in a simultaneous equation system, the value of each endogenous variable is determined by the whole system, the convention is useful in finding a closure to the model. In fact, TABLO offers a very useful facility that can be used to obtain a tentative closure for a model. The facility uses the naming convention to pair variables to the equations that appear to determine them.

At an informal level, we have little difficulty in associating a particular variable with each equation. There are a few cases in which our equation-naming system runs into difficulties because the determination of an endogenous vector variable is spread over more than one vector equation. Another potential difficulty is that some vector variables may be split between the endogenous and exogenous categories.

Each equation in the TABLO model description is linear in the changes (percentage or absolute) of the model's variables. For example, the industry labour demand equations appear as:

```
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)]
   - SIGMA1PRIMEN(i)*[p1prim(i)-p1prim_f(i)];
```

Within the equation, we generally distinguish between change variables and coefficients by using lower-case script for variables and upper-case script for coefficients. In the equation above, the expression (all,i,IND) signifies that the equations are defined over all elements of the set IND (the set of industries) (Note, however, that the gempack solution software ignores case). Thus in the example above, the variables are x1lab_o(i), x1prim(i), a1lab_o(i), p1lab_o(i), p1prim(i) and p1prim_f(i). There are two coefficients, SIGMA1PRIM(i), which is the fixed elasticity of substitution between labour and other primary factors, and SIGMA1PRIMEN(i), which is the fixed elasticity of substitution between an energy aggregate and primary factors. A semi-colon signals the end of the TABLO statement.

VERM equation use the same operators as the formulas above. However, it may be helpful to have an understanding of the derivation of the percentage change form for some common non-linear expressions.

There are three basic rules for deriving the linear equations from the non-linear equations:

the product rule: $X = \beta YZ \Rightarrow x = y + z$, where β is a constant,

the power rule: $X = \beta Y\alpha \Rightarrow x = \alpha y$, where α and β are constants, and

the sum rule: $X = Y + Z \Rightarrow Xx = Yy + Zz$.

The *percentage* change variables x, y and z above represent deviations from the levels values X, Y and Z. The levels values (X, Y and Z) are solutions to the model's underlying levels equations.

Using the product-rule equation as an example, values of 100 for X, 10 for Y and 5 for Z represent an initial solution for a value of 2 for β. Now assume that we perturb our initial solution by increasing the values of Y and Z by 3 per cent and 2 per cent respectively, i.e., we set y and z at 3 and 2. The linear representation of the product-rule equation would give a value of x of 5, with the interpretation that the initial value of X has increased by 5 per cent for a 3 per cent increase in Y and a 2 per cent increase in Z up to a linearization error. Values of 5 for x, 3 for y and 2 for z in the corresponding percentage change equation means that the levels value of X has been perturbed from 100 to 105, Y from 10 to 10.3 and Z from 5 to 5.1.

The theory behind the equations of VERM is the topic of chapter 5. The next chapter contains the VERM TABLO code in its entirety.

4. The TABLO code of VERM

```
!#########################################################################!
!                                                                         !
! VERM:  a dynamic, multi-region model of Finland by Juha Honkatukia      !
! Based on the TERM model (Mark Horridge 2011)                            !
! and MONASH model (Peter Dixon and Maureen Rimmer 2002)                  !
! June 2011                                                               !
!                                                                         !
! ########################################################################!
!Contents                                                                 !
!Section 1: Files                                                         !
!Section 2: Sets and subsets                                              !
!Section 3: Coefficiencts                                                 !
!Section 4: Read insctructions                                            !
!Section 5: Formulas in thematic order                                    !
!Section 6: Variables                                                     !
!Section 7: Updates                                                       !
!Section 8: Equations in thematic order                                   !
!Section 9: Reporting variables                                           !
!                                                                         !
!*************************************************************************!
! Section 1: Files                                                        !
!*************************************************************************!
!#########################################################################!

File INFILE;
File (new) SUMMARY;
File REGSETS # Sets file #;

File  EXTRA      # Input for investments #;
File  EXTRA3     # Alternative for EXTRA in forecast#;
File  EXTRA4     # Lagged version of EXTRA file #;
File  EXTRA5     # Lagged version of EXTRA3 file #;

file PSECDATA #Data on all public sectors#;
File ITER # Iteration number, used in forward-looking expectations #;
ROREXT # Foward-looking expect.: guess of RORs and parameters for algorithm#;

file POPUDATA #Data on age structure#;
file VOSDATA #Data on communal sectors#;

File  BOPACC     # Balance of payments accounts #;
```

```
!#########################################################################!
!*************************************************************************!
! Section 2: Sets and subsets                                             !
!*************************************************************************!
!#########################################################################!

Set                                          ! subscript !
 SRC (dom,imp);                                  !s!
 COM # Commodities #
    read elements from file REGSETS header "COM"; !c!
 MAR # Margin coms #
    read elements from file REGSETS header "MAR"; !m!
 IND # Industries #
    read elements from file REGSETS header "IND"; !i!
 OCC # Labour skill categories #
    read elements from file REGSETS header "OCC"; !o!
 DST # Regions of use #
    read elements from file REGSETS header "REGD";!d!
 ORG # Regions of origin #
    read elements from file REGSETS header "REGS";!r!
 PRD # Regions of margin production #
    read elements from file REGSETS header "REGP";!p!

Set REG = DST intersect ORG; ! special set for comparing DST ORG PRD !
Subset
 MAR is subset of COM;
 REG is subset of PRD;
 PRD is subset of ORG;
Set
 FINDEM # Final demanders #  (HOU, INV, GOV, EXP);!f!
 INT # Intermediate # (INT);
 USR # Users #     = IND + FINDEM;                !u!
 MAINUSR # Main users #    = INT + FINDEM;
 NONMAR # Non-margin goods # = COM-MAR;

Set PRFS (Labour,Capital);
Set FAC(LND,LAB,CAP);
Set COSTCAT # Cost Categories #
 (IntDom, IntImp, ComTax, LAB, CAP, LND, PRODTAX);
Set HOU # Households # (AllHou);

Set NATREG (National);
```

```
Set TOTIND (TotalInds);
Set REGPLUS = REG+NATREG;
Set INDPLUS = IND+TOTIND;
Set INVFACT (Invest,CapRental,Output,InvOverCap,InvOverOut);
Set ELSTRAD # Trade elasticities # (SIGMADOMIMP,SIGMADOMDOM,EXP_ELAST);

Set PSEC # Public sectors govt, local, soc sec funds # (S1311,S1313,S1314);

SET GSEC # sectors collecting commodity taxes # (S1311);
Subset GSEC is subset of PSEC;

Set LSEC # Local municipalities # (S1313);
Subset LSEC is subset of PSEC;

Set FSEC # Soc sec funds # (S1314);
Subset FSEC is subset of PSEC;

Set TIME # Sequence of numbers 0,1,2, ...,25 #
read elements from file ROREXT header "TIME";

Set MAINMACROS # Convenient macros for reporting #
 (RealHou, RealInv, RealGov, ExpVol, ImpVolUsed, ImpsLanded, RealGDP,
AggEmploy,
  realwage_io, p1lab_io, AggCapStock,GDPPI, CPI,ExportPI, ImpsLandedPI,
  Population, NomHou, NomGDP);


Zerodivide off;

!#########################################################################!
!*************************************************************************!
! Section 3: Coefficient declaration listed alphabetically                !
!*************************************************************************!
!#########################################################################!

!Coefficient DEFREGSHR # 1/[No of regions] = default regional share #;
Formula     DEFREGSHR = 1/[sum{r,ORG,1}];!

! Excerpt 2 of TABLO input file: !
! Values from flows data file !

!Domestic basic prices = Output prices
 Imported basic prices = CIF prices
 Delivered values = Basic + Margins
```

```
 Purchasers' values = Basic + Margins + Tax = Delivered + Tax !

Coefficient
(parameter) (all,d,DST)(all,j,IND)
   ADJ_COEFF(d,j) # Rate of disappearance of disequilibrium in rors RE#;
(parameter)(all,d,DST)(all,j,IND)
   ADJ_RE(d,j) # Controls adjustment of expected rors RE#;
(parameter) ADJDUMYEAR1 # Adjusts DUM_YEAR1, one in first year, then zero #;


(all,d,DST)(all,s,PSEC) AGEBENS(S,d) #Age benefits#;
  AGGLAB #Labour demand at national level#;
  ALPHA1 # Controls wage response to gaps between labour demand & supply#;
  ALPHA2 # Slope of long-run labour supply curve #;
(all,d,DST) AV_PROP_CON(d) #Average propensity to consume#;

(all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST) BASSHR(c,s,r,d)
# Share of basic value in all-user delivered price #;
(all,c,COM)(all,d,DST)(all,h,HOU)
    BLUX(c,d,h) # Ratio, (supernumerary expenditure/total expenditure) #;
(all,c,COM)(all,d,DST)(all,h,HOU)
    BUDGSHR(c,d,h) # Household average budget shares #;
(all,i,IND)(all,d,DST)            CAP(i,d)     # Rentals to capital #;
(all,d,DST) CAP_I(d)  # Total rentals to capital #;
(all,d,DST)(all,j,IND) CHKGR1(d,j)# One if K_GR_MIN(j) >= K_GR(j),else zero
Fr#;
(all,d,DST)(all,j,IND) CHKGR2(d,j)# One if K_GR_MAX(j) <= K_GR(j),else zero
Fr#;
(all,c,COM)(all,d,DST) CHKHOUPUR(c,d) # Household demand check #;
(parameter)(all,d,DST)(all,i,IND)
   COEFF_SL(d,i)  # Coefficient in capital supply curve #;
(all,t,TIME)COEFF_TIME(t) # Vector of consecutive numbers 0,1,2,....,NYEARS
R#;
(parameter) COEFF_NYEAR # Zero in the last year of a simulation, else one #;
(all,d,DST)(all,i,IND)(all,s,PSEC) COL_PAYROLLS(i,d,s) #Payroll taxes#;
    COL_PAYRTOT # Aggregate collection of payroll taxes F #;
(all,d,DST)(all,s,PSEC) COL_PAYRTOTS(D,s)#Payroll taxes#;



(all,i,IND)(all,co,COSTCAT)(all,d,DST) COSTMAT(i,co,d) # Cost Matrix #;
    DEFHOUSHR # Default household share #;
(all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
    DELIVRD(c,s,r,d) # Trade + margins #;
```

```
(all,c,COM)(all,s,SRC)(all,d,DST)
   DELIVRD_R(c,s,d) # Demand in region d for delivered goods from all regions
#;

(parameter)(all,d,DST)(all,i,IND)
   DEP(d,i)# rate of depreciation#;
(parameter) DIFF    # Max. difference from trend rate of growth #;

(parameter)(all,d,DST)(all,i,IND)
   DISEQRE_B(d,i)   # Diseq. in rational expect. version of ror in base year
#;
(parameter)(all,d,DST)(all,i,IND)
   DISEQSE_B(d,i)      # Diseq. in static expect. version of ror in base year
#;
(parameter) DUM_IT1 # One in 1st iter of forecast or policy simulation #;
(parameter)(all,t,TIME)
   DUM_TIME(t) # Equals 1 if t  = YEAR, 0 otherwise #;
(parameter)(all,t,TIME)
   DUM_TIME_LAG(t) # Equals 1 if t= YEAR-1, else 0 #;
   DUM_YEAR1 # Zero in first year, then one #;
(parameter) DUMMY_DEC # Zero in simulations off 1/2-way database, else one#;
(parameter)(all,d,DST)(all,j,IND) EROR_B(d,j)
            # Expected. ror in init. sol for t, usually exptd. ror for t-1#;
(parameter)(all,d,DST)(all,j,IND) EROR_F(d,j)
            # Expected ror, imposed in soln. for year t, beyond iter. one #;
(parameter)(all,d,DST)(all,j,IND)(all,t,TIME) EROR_G_B(d,j,t)
            # Matrix for transfer of info on expected rors between iterations
#;
(all,d,DST)(all,j,IND)(all,t,TIME) EROR_G(d,j,t)
            # Matrix for transfer of info on expected rors between iterations
#;
(all,d,DST) F_EEQROR(d) # Scalar shifter in EEQROR  #;
(all,d,DST)(all,i,IND)EEQROR(d,i) # Equilibrium expected rate of return, SE
#;
            EMPL_RATE # Employment rate #;
(all,d,DST)(all,i,IND)F_EEQROR_I(d,i) # Industry specific shifter in EEQROR
#;
   EMPLOY # Aggregate employment: 1 in init. sol. for yr 1 #;
   EMPLOY_OLD # Aggregate employment, f'cast: 1 in init.sol.yr 1#;
(parameter)     EMPLOY_B # Aggregate employment in year 1, base #;
(parameter)     EMPLOY_L_B # Aggregate employment in year t-1, base #;
(parameter)     EMPLOY_O_B
    # Aggregate employment in year t, forecast value, base #;
(parameter)     EMPLOY_O_L_B
```

```
    # Aggregate employment in year t-1, forecast value, base R#;
(all,d,DST) EMPLOYS(d) #Employment#;
(all,d,DST) EMP_RATE(d) #Employment rate#;
   EMP_RATEN #National employment level#;
(all,d,DST)(all,h,HOU) EPSAVE(d,h) # Average EPS: should be 1 #;
(all,c,COM)(all,d,DST)(all,h,HOU)
   EPSH(c,d,h)  # Household expenditure elasticities #;
(parameter)(all,c,COM) EXP_ELAST(c)   # Export demand elasticities #;
(all,c,COM)(all,r,REG) EXPSHR(c,r) # Share of output eventually exported #;
   FEMPADJ # Level of the shift variable in E_d_f_empadj #;
(parameter)     FEMPADJ_B # Level of the shift variable in E_d_f_empadj in t-
1#;
(parameter)     FEMPADJ_O # Level of the shift variable in
E_d_f_empadj,forec#;

(all,d,DST)(all,h,HOU)
   FRISCHH(d,h) # Frisch LES 'parameter'= - (total/luxury) #;
(all,d,DST)(all,s,PSEC) GOV_DEFS(D,S) #Public sector deficits#;
(all,d,DST)(all,i,IND)(all,s,PSEC) GOVSHRINVS(D,i,s) #Public share from inv#;
(all,d,DST) GOVTOMUN(d);
(all,d,DST) GOVTOSSF(d);

(all,d,DST)(all,s,PSEC) GRANTSS(s,d) #Grants from public sectors#;
(all,c,COM)(all,d,DST)(all,h,HOU) HOUPUR(c,d,h) #Household demands#;
(all,d,DST)(all,h,HOU) HOUPUR_C(d,h) # Household demand totals #;
(all,c,COM)(all,d,DST) HOUPUR_H(c,d) # Household demand totals #;
(all,d,DST) HOUS_DIS_INC(d) #Household disposable income#;
(all,d,DST) HOUS_SAV(d) #Household saving#;
(all,c,COM)(all,d,DST)(all,h,HOU) HOUSHR(c,d,h) #Household shares#;
(all,r,ORG) IMPLANDED_C(r) # Imports landed in r #;
(all,d,DST) IMPUSED_C(d) # Imports used in d #;
(all,d,DST)(all,s,PSEC) INCTAXS(d,S) #Income taxes#;
    INF   # Rate of inflation measured by CPI #;
    INF_L   # Rate of inflation, lagged #;
    INTR      # Nominal interest rate#;
    INTR_L       # Rate of interest, lagged #;
    INT_PSD # Interest on public sector debt Fs #;
(all,c,COM)(all,i,IND)(all,d,DST)
   INVEST(c,i,d) # Investment at purchasers prices #;
(all,c,COM)(all,d,DST) INVEST_I(c,d) # Investment by commodity and region #;
(all,i,IND)(all,d,DST) INVEST_C(i,d) # Investment by industry and region #;
(all,r,REGPLUS)(all,i,INDPLUS)(all,z,INVFACT)
   INVFACTS(r,i,z) # Investment ratio summary #;
(all,i,IND)(all,z,INVFACT) INVFACTS_r(i,z) # temp total #;
```

```
(all,r,REGPLUS)(all,z,INVFACT) INVFACTS_I(r,z) # temp total #;
(parameter)     ITER_ADJUST # Used in adjusting the iteration number #;
    ITER_NUM   # Iteration number, used when expectations are forward-looking
#;
(parameter)
    ITER_NUM_B # Iteration number, used when expectations are forward-looking
#;
(all,d,DST)(all,i,IND)K_GR(d,i)  # Growth rate of capital stock in ind. i#;
(parameter)(all,d,DST)(all,i,IND)
    K_GR_MIN(d,i)# Minimum growth rate for capital stock in ind. i#;
(parameter)(all,d,DST)(all,i,IND)
    K_GR_MAX(d,i)# Maximum growth rate for capital stock in ind. i#;

(all,i,IND)(all,o,OCC)(all,d,DST) LAB(i,o,d)   # Wage matrix #;
(all,i,IND)(all,d,DST) LAB_O(i,d)   # Total labour bill in industry i #;
(all,o,OCC)(all,d,DST) LAB_I(o,d) # Total wages by skill #;
(all,d,DST) LAB_IO(d) # Total wages #;
(all,d,DST) LND_I(d)  # Total rentals to land #;
(all,o,OCC) LAB_ID(o) # Total wages by occupation #;
(all,i,IND) LAB_OD(i)  # Total national wage cost #;
(all,d,DST) LAB_SUP(d) #Labour supply#;
    LAB_SUPN # Labour supply, national#;
    LAB_SUPN_O # Labour supply, forecast run #;
(all,c,COM)(all,s,SRC)
    LEVP0(c,s) # Levels basic prices #;
    LEV_PLAB   # Nominal wage level#;
    LEV_PLAB_L  # Nominal wage level, lagged #;
(parameter)     LEV_PLAB_B  # Nominal wage level, start of the year#;
(parameter)     LEV_PLAB_L_B # Nominal wage level, lagged start of the year#;
(all,i,IND)(all,d,DST)           LND(i,d)     # Rentals to land #;
    LEV_CPI    # Consumer price level  #;
(parameter)     LEV_CPI_B  # Consumer price level, start of the year, FI#;
    LEV_CPI_L  # Consumer price level, lagged, #;
(parameter)     LEV_CPI_L_B # Consumer price level, lagged start of the
year#;
    LEV_CPI_2L # CPI double lagged, that is for t-2#;
(parameter)     LEV_CPI_2L_B # CPI double lagged, base #;

(all,c,COM)(all,s,SRC)(all,d,DST)
    LOCUSE(c,s,d) # Non-export delivered value of regional composite c,s in d
#;
(all,c,COM)(all,d,DST)   LOCUSE_S(c,d) # Non-export delivered value of good c
#;
(all,c,COM)   LOCUSE_SD(c) # Non-export national delivered value of good c #;
```

```
(all,c,COM)(all,i,IND)(all,d,DST)
    MAKE(c,i,d) # MAKE multiproduction matrix #;
(all,i,IND)(all,d,DST) MAKE_C(i,d) # All production by industry i #;
(all,c,COM)(all,i,IND) MAKE_D(c,i) # National MAKE matrix #;
(all,c,COM)(all,d,DST) MAKE_I(c,d) # Total production of commodities #;
(all,c,COM) MAKE_IR(c) # National commodity outputs #;
(all,c,COM)(all,i,IND)(all,d,DST) MAKESHR1(c,i,d) # Com share in Ind output
#;
(all,c,COM)(all,i,IND)(all,d,DST) MAKESHR2(c,i,d) # Ind share in Com supply
#;
(all,c,COM)(all,s,SRC)(all,m,MAR)(all,r,ORG)(all,d,DST)
    MARSHR(c,s,m,r,d) # Share of margin m in all-user delivered price #;

(all,d,DST) MUNTOGOV(d);
(all,i,IND) NATVTOT(i) # Total national industry cost plus tax #;
(all,d,DST)(all,s,PSEC)  NETINTS_G(s,d);
(all,d,DST)(all,s,PSEC) NETINTS_HS(d,s);

(all,d,DST)(all,s,PSEC) NET_TAXTOTS(d,s);

(integer,parameter)(all,c,COM) NINDPROD(c)# No of industries that make good
c#;
(integer,parameter)(all,i,IND) NCOMPROD(i)# No of commodities made by ind i
#;
(parameter)     NOFITERS # N of forec. iterations in RE sims #;
(parameter)     NYEARS # Length in years of the simulation horizon #;
(parameter)
    ONE_IT1_REP # Used in rat. exp.: one in 1st iter of policy, else zero #;
(parameter)
    ONE_ITER1  # Used in rat. exp.: one in 1st iter of forecast, else zero #;
(all,d,DST)(all,s,PSEC) OTHBENS(s,d) #Other public sector benefits#;
(all,d,DST)(all,s,PSEC) OTHCAPGOVS_(s,d) #Other public sector investments#;
(all,d,DST)(all,s,PSEC) OTHCAPGOVS(s,d) #Other public sector investments#;
(all,d,DST)(all,s,PSEC) OTHGOVREVS(S,d) #Other public sector revenues#;
(all,d,DST)(all,i,IND)     PCAP_AT_T(d,i)
    # Asset price of capital stocks, start of forecast year #;
(parameter)(all,d,DST)(all,i,IND) PCAP_AT_T_B(d,i)
     # Asset price of capital stocks, start of data year, base #;
(all,d,DST)(all,i,IND) PCAP_AT_T1(d,i)
     # Asset price of capital stocks, end of data year, base #;
(parameter)(all,d,DST)(all,i,IND) PCAP_AT_T1_B(d,i)
     # Asset price of capital stocks, end of data year, base #;
(all,d,DST)(all,i,IND)     PCAP_I(d,i)
    # Asset price of capital by industry, average in year #;
```

```
(parameter)(all,d,DST)(all,i,IND)
                PCAP_I_B(d,i)
    # Asset price of capital by industry, average in year, base #;
(all,d,DST)(all,i,IND)     PCAP_I_L(d,i)
     # Asset price of capital stocks, average of  year t-1 #;
(parameter)(all,d,DST)(all,i,IND)
                PCAP_I_L_B(d,i)
     # Asset price of capital stocks, average of  year t-1, init sol#;
(all,d,DST)(all,i,IND)  POW_PAYROLL(i,d) #Power of payroll taxes#;
(all,d,DST)(all,i,IND)(all,s,PSEC) POW_PAYROLLS(s,i,d) #Pow payroll taxes#;

(parameter)(all,d,DST) POP(d)         # Population #;
(all,i,IND)(all,d,DST) PRIM(i,d) # Total factor input to industry i#;
(all,i,IND)(all,f,FAC)(all,d,DST) PRIMCOST(i,f,d) # Total factor inputs #;
(all,d,DST) PRIM_I(d) # Regional GDP at factor cost #;
(all,i,IND) PRIM_D(i)     # Total factor input to industry i #;
(all,i,IND)(all,d,DST) PRIMSHR(i,d) # PRIM(i,d)/PRIM_I(d) #;
(all,i,IND)(all,d,DST)           PRODTAX(i,d) # Taxes on production #;
(all,d,DST)(all,s,PSEC) PSDATT(s,d) # Public sector debt, start of year #;
(parameter)
(all,d,DST)(all,s,PSEC) PSDATT_1_B(d,s)#Public sector debt, end of year, base
#;
(parameter)
(all,d,DST)(all,s,PSEC) PSDATT_B(d,s) #Publ sector debt, start of year, base
#;
(all,d,DST)(all,s,PSEC) PSDATTPLUS1(d,s) # Public sector debt, end of year #;

(all,i,IND)(all,d,DST) PTXRATE(i,d)   # Rate of production tax #;
(all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST) PUR(c,s,u,d) # Purchasers
values#;
(all,c,COM)(all,u,USR)(all,d,DST) PUR_S(c,u,d) # Purch. values, sum over s #;
(all,u,USR)(all,d,DST) PUR_CS(u,d) # Expenditure on goods #;
(all,d,DST)(all,i,IND)     QCAPATT(d,i)
    # Quantity of capital stocks, start of forecast year, ie K(t)#;
(all,d,DST)(all,i,IND)     QCAPATTPLUS1(d,i)
     # Quantity of capital stocks, start of forecast year, ie K(t+1)#;
(parameter)(all,d,DST)(all,i,IND) QCAPATT_B(d,i)
    # Quantity of capital stocks, start of forecast year, ie K(t), base#;
(parameter)(all,d,DST)(all,i,IND) QINV_BASE(d,i)
    # Quantity of investments, start of forec year#;
(all,d,DST)(all,i,IND)QINVEST(d,i)
    # Quantity of investment by industry i in forec #;
(parameter)     RALPH   # Share of depreciation that is tax deductible R#;
(all,d,DST)(all,h,HOU) RATIOCPI(d,h)# (Current/initial) CPI#;
```

```
(all,m,MAINMACROS) RATIOMMACRO(m) # Initial/final ratio #;
(all,d,DST) RATIOPRIM_I(d) # Ratio (current/initial) real GDP at factor cost
#;
(all,i,IND) RATIOPRIM_D(i) # (Current/initial) ratio #;
    RINT_PSD #Interest on public sector debt#;
    RINT #Real rate of interest#;
(parameter)     R_DEFGDP_B
    # Level of the ratio of the government deficit to GDP  #;
(parameter)     RINT_B      # Real interest rate, base#;
                RINT_L     # Real interest rate, lagged#;
(parameter)     RINT_L_B   # Real interest rate, base, lagged #;
                RINT_PT_SE # Post-tax real interest rate, SE#;
(all,d,DST)(all,i,IND) ROR_SE(d,i)
   # Rates of return in year t: static expectations#;
(parameter)(all,d,DST)(all,i,IND) ROR_SE_BASE(d,i)
   # SE of rors in init. sol. for year t#;
(parameter)(all,d,DST)(all,i,IND) RORN(d,i)
  # Normal rates of return, given as data #;
(all,d,DST)(all,i,IND) ROR_ACT_L(d,i) # Lagged rate of return, ror in t-1 #;
(parameter)(all,d,DST)(all,i,IND) ROR_ACT_L_B(d,i)
  # Lagged rate of return, init. soln., usually ror in t-2#;
(all,c,COM)(all,r,ORG)(all,d,DST) ROWDEM(c,r,d) # Eventually exported goods
#;
(all,c,COM)(all,r,ORG) ROWDEM_D(c,r) # Eventually exported goods #;
    RWAGE   # Real wage in year t, CPI deflated #;
(parameter)     RWAGE_B # Real wage in year t, CPI deflated, base #;
(parameter)     RWAGE_L_B # Real wage in year t-1, CPI deflated,base  #;
    RWAGE_OLD # Real wage in year t, CPI deflated, forecast #;
(parameter)     RWAGE_OLD_B # Real wage in year t, CPI deflated, forec,
base#;
(parameter)     RWAGE_O_L_B # Real wage in year t-1, CPI deflated, forec,
base#;
    RWAGE_PT # Real post-tax wage in year t, CPI deflated #;
(parameter)     RWAGE_PT_B # Real post-tax wage in year t, CPI deflated, base
#;
(parameter)     RWAGE_PT_L_B # Real post-tax wage, year t-1,CPI
deflated,base#;
(parameter)     RWAGE_PT_O_B # Real post-tax wage,year t,CPI
defl.,forec,base#;
   RWAGE_PT_OLD # Real post-tax wage,year t, CPI defl, forecast#;
(parameter)     RW_PT_O_L_B # Real post-tax wage, t-1, CPI defl, forec,
base#;
(all,d,DST)(all,c,COM)(all,r,SRC)(all,s,PSEC) SCRSHR5(d,c,r,s);
(all,i,IND)(all,d,DST)(all,p,prfs) SHRLK(i,d,p)#Primary factor shares#;
```

```
(parameter)(all,i,IND) SIGMA1LAB(i)    # CES substitution between skill types
#;
(parameter)(all,i,IND) SIGMAPRIM(i)   # CES substitution, primary factors #;
(parameter)(all,c,COM) SIGMADOMIMP(c) # Substitution elast between dom/imp #;
(parameter)(all,c,COM) SIGMADOMDOM(c) # Substitution elast between origins #;
(parameter)(all,m,MAR) SIGMAMAR(m)    # Subst elast between margin origins #;
(parameter)(all,i,IND) SIGMAOUT(i)    # CET transformation elasticities #;
(all,i,IND)(all,o,OCC)(all,d,DST) SLAB_I(i,o,d) # Industry shares of wages #;
(all,o,OCC)(all,d,DST) SLAB_ID(o,d) # Region shares of wages #;
(all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST) SRCSHR(c,s,u,d) # Imp/dom
shares#;
(all,c,COM)(all,d,DST)(all,h,HOU)
   SLUX(c,d,h) # Marginal household budget shares #;
(parameter)(all,d,DST)(all,i,IND) SMURF(d,i)
                # Recip. of slopes of cap. supply curves at
K_GR(i)=TREND_K(i)#;

(all,d,DST) SSFTOGOV(d);
(all,i,IND)(all,d,DST)           STOCKS(i,d)  # Domestic inventories #;
(all,m,MAR)(all,r,ORG)(all,d,DST)(all,p,PRD)
   SUPPMAR(m,r,d,p) # Margins supplied by p on goods passing from r to d #;
(all,m,MAR)(all,r,ORG)(all,d,DST)    ! above should equal below !
    SUPPMAR_P(m,r,d)  # Total demand for margin m on goods from r to d #;
(all,m,MAR)(all,r,ORG)(all,p,PRD)
    SUPPMAR_D(m,r,p)  # Total demand for margin m (from p) on goods from r #;
(all,m,MAR)(all,p,PRD)
    SUPPMAR_RD(m,p)   # Total demand for margin m produced in p #;
(all,m,MAR)(all,d,DST)(all,p,PRD)
    SUPPMAR_R(m,d,p) # Margins supplied by p on all goods passing to d #;
(all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST)
   TAX(c,s,u,d) # Commodity taxes #;
(all,d,DST) TAX_CSI(D)#Commodity taxes#;
(all,d,DST) TAXEXP(d)#Export taxes#;
(all,d,DST) TAXGOV(d)#Taxes on govt#;
(all,d,DST) TAXHOU(d)#Taxes on households#;
(all,d,DST) TAXINT(d)#Taxes on intermediates#;
(all,d,DST) TAXINV(d)#Taxes on investments#;
(all,d,DST) TAXPROD(d)#Production taxes#;
(all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST) TAXRATE(c,s,u,d) # Tax rate #;
(all,d,DST)(all,s,PSEC) TAXS_CAP(D,S)#Tax on capital income#;
(all,d,DST)(all,s,PSEC) TAXS_AB(d,s) #Tax on age benefits#;
(all,d,DST)(all,s,PSEC) TAXS_AB_RATE(s,d) #Tax rate on age benefits#;
(all,d,DST)(all,s,PSEC) TAXS_K_RATE(s,d) #Capital income tax rate#;
   TAX_K_RATE  # Rate of tax on capital and land income#;
```

```
   TAX_LAB # Tax collected from labour income #;
   TAX_CAP # Tax collected from labour income #;
   TAX_LND # Tax collected from land income #;
   TAX_L_RATE # Rate of tax on labour income, year t#;
(parameter)     TAX_L_RATE_L    # Rate of tax on lab income, year t-1, base#;
   TAX_L_RATE_O # Rate of tax on labour income, forecast, year t#;
   TAX_L_R_O_L # Rate of tax on lab income, t-1, forecast, base #;
(all,d,DST)(all,s,PSEC) TAXS_LAB(d,s) #Labour income tax#;
(all,d,DST)(all,s,PSEC) TAXS_LND(d,s) #Capital income tax on land#;
(all,d,DST)(all,s,PSEC) TAXS_L_RATE(s,d) #Labour income tax rate#;
(all,d,DST)(all,s,PSEC) TAXS_OB(D,S) #Tax on other benefits#;
(all,d,DST)(all,s,PSEC) TAXS_OB_RATE(S,d)#Tax rate on other benefits#;
(all,d,DST)(all,s,PSEC) TAXS_UB(D,S) #Tax on unemployment benefits#;
(all,d,DST)(all,s,PSEC) TAXS_UB_RATE(s,d)#Tax rate on unemployment benefits#;
    TINY # Small number to prevent zerodivides or singular matrix #;
(all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
   TRADE(c,s,r,d) # Sourcing matrix (at basic prices) #;
(all,s,SRC)(all,d,DST) TRADE_CR(s,d) # Total direct demands dest d #;
(all,c,COM)(all,s,SRC)(all,r,ORG) TRADE_D(c,s,r) # Total direct demands #;
(all,c,COM)(all,s,SRC)(all,d,DST) TRADE_R(c,s,d) # Total direct demands #;
(all,c,COM)(all,s,SRC)          TRADE_RD(c,s) # Total national direct demands
#;
(all,c,COM)(all,e,ELSTRAD) TRADELAST(c,e) # Trade elasticities #;
(all,c,COM)(all,s,SRC)(all,m,MAR)(all,r,ORG)(all,d,DST)
   TRADMAR(c,s,m,r,d) # Margins on trade matrix (at basic prices)#;
(all,m,MAR)(all,r,ORG)(all,d,DST)
   TRADMAR_CS(m,r,d) # Total demand for margin m on goods from r to d #;
(all,d,DST)(all,s,PSEC)(all,z,PSEC) TRANSPSECS(d,s,z);
(all,d,DST)(all,s,PSEC) TRANSS_G(d,s) #Transfers between public sectors#;
(all,d,DST)(all,s,PSEC) TRANSS_H(d,s) #Transfers between public sectors#;

(parameter)(all,d,DST)(all,i,IND) TREND_K(d,i)
   #'Historical' trend gr. rate for capital stock in i#;
(all,d,DST)(all,s,PSEC) UNEMPBENS(S,d) #Unemployment benefits#;
(all,d,DST) UN_RATE(d) #Unemployment rate#;
            UN_RATEN  #National unemployment rate#;
(all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST)
   USE(c,s,u,d) # Delivered value of demands: basic + margins (ex-tax) #;
(all,c,COM)(all,s,SRC)(all,d,DST)
   USE_U(c,s,d) # Total delivered value of regional composite c,s in d #;
(all,c,COM)(all,s,SRC)(all,d,DST)
   USE_I(c,s,d) # All-intermediate delivered value of reg. composite c,s in d
#;
(all,d,DST) V0GDPINC(d);
```

```
(all,d,DST)(all,s,PSEC) V0TAX_CSIS(D,s);
(all,d,DST)(all,i,IND)     V1CAP(d,i)  # Capital rentals #;
(all,d,DST)(all,i,IND)     V1CAP_(d,i) # Capital rentals #;
                V1CAP_I   # Total payments to capital #;

(all,d,DST) V1PRIM_I(d);
(all,d,DST)(all,i,IND)(all,s,PSEC) V2SHRSG_(d,i,s);
(all,d,DST)(all,i,IND)(all,s,PSEC) V2SHRSG(d,i,s);
(all,d,DST)(all,i,IND)     V2TOT(d,i)   # Investment #;
(all,d,DST)(all,s,PSEC) V2TOT_G_IS(d,s);
(all,d,DST)(all,i,IND)(all,s,PSEC) V2TOTSG_(d,i,s);
(all,d,DST) V3TOT(d);
(all,d,DST)(all,c,COM)(all,s,PSEC) V5SHRS_(D,c,S);
(all,d,DST)(all,c,COM)(all,s,PSEC) V5TOT(D,c,S);
(all,d,DST)(all,c,COM)(all,r,SRC)(all,s,PSEC) V5TOTR_(D,c,r,S);
(all,d,DST)(all,c,COM)(all,r,SRC)(all,s,PSEC)V5TOTR(D,c,r,S);
(all,d,DST)(all,s,PSEC) V5TOTS(D,S);
(parameter)(all,d,DST)(all,c,COM)(all,s,PSEC) V5TOTS_(D,c,S);

(all,i,IND)(all,d,DST)  VARCST(i,d)  # Shortrun variable cost of industry i
#;
(all,d,DST)(all,i,IND)     VCAP_AT_T(d,i)
    # Start of year capital stocks valued at start of yr prices #;
(all,d,DST)(all,i,IND)     VCAP_AT_TM(d,i)
   # Val of capital, start of f'cast year in mid-yr prices #;
(all,i,IND)(all,d,DST)  VCST(i,d)    # Total cost of industry i #;
(all,c,COM)(all,r,REG) VDOMEXP(c,r)
  # Value good c made in r sent to other domestic regions (non-margin) #;
(all,c,COM)(all,d,REG) VDOMIMP(c,d)
# Value domestic good c used in d made in other domestic regions (non-
margin)#;
(all,r,REG) VDOMEXP_C(r)
  # Value goods made in r sent to other domestic regions (non-margin) #;
(all,d,REG) VDOMIMP_C(d)
# Value domestic goods used in d made in other domestic regions (non-
margin)#;
(all,i,IND)(all,d,DST)  VTOT(i,d)    # Total industry cost plus tax #;
(all,m,MAINMACROS)(all,q,REG) WMAINMACRO(m,q) # Weights to aggregate macros
#;
(all,m,MAINMACROS) WNATMACRO(m) # Total of WMainMacro #;
   YEAR # Year of current solution #;
(parameter) YEAR_B # Year of current solution FIt#;
(parameter) YR_POLICY # Year in which policy shock is first anticipated#;
(parameter) ZERO_PYR1 # Zero in 1st yr of a rational-exp. policy sim, else
```

```
1#;


!##############################################################################!
!******************************************************************************!
! Section 4: Read instructions for coefficients listed alphabetically          !
!******************************************************************************!
!##############################################################################!

Read ADJ_COEFF  from file EXTRA header"ADJC";
Read ADJ_RE from file ROREXT header"ADRE";
Read AGEBENS from file PSECDATA header "AGES";
Read CAP     from file INFILE header "1CAP";
Read COEFF_TIME from file ROREXT header"TYME";
Read DEP         from file EXTRA header "DEPR";
Read DUM_YEAR1   from file EXTRA header"0045";
Read DUMMY_DEC   from file ROREXT header"DMDC";
Read EMPLOY from file EXTRA header"EMPL";
Read EMPLOY_L_B from file EXTRA4  header"EMPL";
Read EMPLOY_O_L_B from file EXTRA5 header"EMPL";
Read EMPLOY_OLD from file EXTRA3 header"EMPL";
Read EMPLOYS from file PSECDATA header "REMP";
Read EPSH from file INFILE header "XPEL";
Read EROR_G from file ROREXT header"RORG";
Read EXP_ELAST   from file INFILE header "P018";
Read F_EEQROR from file EXTRA header"FCSE";
Read F_EEQROR_I from file EXTRA header"FSTA";
Read FEMPADJ from file EXTRA  header"EADJ";
Read FEMPADJ_O from file EXTRA3 header"EADJ";
Read FRISCHH from file INFILE header "P021";
Read GOVTOMUN from file PSECDATA header "G2M";
Read GOVTOSSF from file PSECDATA header "G2S";
Read GRANTSS from file PSECDATA header "GRNS";
Read INVEST  from file INFILE header "2PUR";
Read ITER_NUM from file ITER header"ITNO";
Read LAB     from file INFILE header "1LAB";
Read LAB_SUP from file PSECDATA header "POPW";
Read LEV_CPI    from file EXTRA header "LCPI";
Read LEV_CPI_L  from file EXTRA header "CPIL";
Read LEV_CPI_2L from file EXTRA header"LCP2";
Read LEV_PLAB   from file EXTRA header "PLAB";
Read LEV_PLAB_L from file EXTRA header "PLAL";
```

```
Read LND      from file INFILE header "1LND";
Read MAKE     from file INFILE header "MAKE";
Read MUNTOGOV from file PSECDATA header "M2G";
Read NETINTS_G from file PSECDATA header "NINS";
Read NOFITERS from file ROREXT header"NFIT";
Read NYEARS from file ROREXT header"HORZ";
Read OTHBENS from file PSECDATA header "OTHS";
Read OTHCAPGOVS_ from file PSECDATA header "OGIS";
Read OTHCAPGOVS from file PSECDATA header "OGIF";
Read OTHGOVREVS from file PSECDATA header "OTGS";
Read PCAP_I       from file EXTRA header "PCAI";
Read PCAP_I_B     from file EXTRA header"PCAB";
Read PCAP_I_L     from file EXTRA header "PCAL";
Read PCAP_AT_T    from file EXTRA header "PCAP";
Read PCAP_AT_T1   from file EXTRA header "PCAT";
Read POP          from file INFILE header "PO01";
Read POW_PAYROLL from file PSECDATA header "POPR";
Read POW_PAYROLLS from file PSECDATA header "POPS";
Read PRODTAX from file INFILE header "1PTX";
Read PSDATT       from file PSECDATA header "MDEP";
Read RALPH       from file EXTRA header "RLPH";
Read RINT from file EXTRA header "RINT";
Read RINT_L      from file EXTRA header"RNTL";
Read RORN        from file EXTRA header"RORN";
Read RWAGE       from file EXTRA  header"RWAG";
Read RWAGE_L_B from file EXTRA4 header"RWAG";
Read RWAGE_O_L_B from file EXTRA5 header"RWAG";
Read RWAGE_OLD from file EXTRA3 header"RWAG";
Read SIGMA1LAB     from file INFILE header "SLAB";
Read SIGMAPRIM    from file INFILE header "P028";
Read SIGMADOMIMP from file INFILE header "P015";
Read SIGMADOMDOM from file INFILE header "SGDD";
Read SIGMAMAR     from file INFILE header "SMAR";
Read SIGMAOUT     from file INFILE header "SCET";
Read SMURF       from file EXTRA header"SMRF";
Read SSFTOGOV from file PSECDATA header "S2G";
Read STOCKS  from file INFILE header "STOK";
Read SUPPMAR from file INFILE header "MARS";
Read TAX     from file INFILE header "UTAX";
Read TAXS_AB_RATE from file PSECDATA header "TLAS";
Read TAXS_K_RATE from file PSECDATA header "TXKS";
Read TAXS_L_RATE from file PSECDATA header "TLRS";
Read TAX_L_RATE  from file EXTRA header "TLRT";
Read TAX_L_RATE_L from file EXTRA4 header"TLRT";
```

```
Read TAX_L_RATE_O from file EXTRA3 header"TLRT";
Read TAX_L_R_O_L from file EXTRA5 header"TLRT";
Read TAXS_OB_RATE from file PSECDATA header "TLOS";
Read TAXS_UB_RATE from file PSECDATA header "TLUS";
Read  TRADE   from file INFILE header "TRAD";
Read  TRADMAR from file INFILE header "TMAR";
Read TREND_K    from file EXTRA header"TRNK";
Read UNEMPBENS from file PSECDATA header "UBES";
Read USE      from file INFILE header "BSMR";
Read V2TOTSG_ from file PSECDATA header "RGI";
Read V5TOTS_ from file PSECDATA header "RGC";
Read VCAP_AT_T   from file EXTRA header "VCAP";
Read YEAR from file EXTRA header"YEAR";
Read YR_POLICY from file ROREXT header"YRPL";



Assertion ! various sign checks on data !
(initial) # MAKE>=0 # (all,c,COM)(all,i,IND)(all,d,DST) MAKE(c,i,d)>=0;
(initial) # USE>=0 # (all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST)
   USE(c,s,u,d) >=0;
(initial) # USE=0 --> TAX=0 # (all,c,COM)(all,s,SRC)(all,u,USR)
  (all,d,DST:USE(c,s,u,d)=0) TAX(c,s,u,d)=0;
(initial) # TRADE>=0 # (all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
   TRADE(c,s,r,d)>=0;
(initial) # TRADMAR>=0 #
(all,c,COM)(all,s,SRC)(all,m,MAR)(all,r,ORG)(all,d,DST)
   TRADMAR(c,s,m,r,d)>=0;
(initial) # TRADE=0 --> TRADMAR=0 # (all,c,COM)(all,s,SRC)(all,r,ORG)
   (all,d,DST:TRADE(c,s,r,d)=0)(all,m,MAR) TRADMAR(c,s,m,r,d)=0;
(initial) # SUPPMAR>=0 # (all,m,MAR)(all,r,ORG)(all,d,DST)(all,p,PRD)
   SUPPMAR(m,r,d,p) >=0;
(initial) # INVEST>=0 # (all,c,COM)(all,i,IND)(all,d,DST)
   INVEST(c,i,d)>=0;

(initial) # LAB>=0 # (all,i,IND)(all,o,OCC)(all,d,DST) LAB(i,o,d) >=0;
(initial) # CAP>=0 # (all,i,IND)(all,d,DST)            CAP(i,d)   >=0;
(initial) # LND>=0 # (all,i,IND)(all,d,DST)            LND(i,d)   >=0;

(initial) # SIGMA1LAB>=0 #    (all,i,IND) SIGMA1LAB(i)    >=0;
(initial) # SIGMAPRIM>=0 #   (all,i,IND) SIGMAPRIM(i)   >=0;
(initial) # SIGMADOMIMP>=0 # (all,c,COM) SIGMADOMIMP(c) >=0;
(initial) # SIGMADOMDOM>=0 # (all,c,COM) SIGMADOMDOM(c) >=0;
(initial) # SIGMAMAR>=0 #    (all,m,MAR) SIGMAMAR(m)    >=0;
```

```
(initial) # POP>=0 #        (all,d,DST) POP(d)        >=0;
(initial) # SIGMAOUT>=0 #   (all,i,IND) SIGMAOUT(i)   >=0;
(initial) # EXP_ELAST>=0 #  (all,c,COM) EXP_ELAST(c)  >=0;


!##########################################################################!
!************************************************************************!
! Section 5: Formulas in thematic groups                                 !
!************************************************************************!
!##########################################################################!
!**********************************************************************!
!Section 5.1 Formulas used for defining                                !
! purchasers prices and imp/dom shares                                 !
!**********************************************************************!

Zerodivide default 0.5;
Formula
(all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST)
            PUR(c,s,u,d) = USE(c,s,u,d) + TAX(c,s,u,d);
(all,c,COM)(all,u,USR)(all,d,DST)
            PUR_S(c,u,d) = sum{s,SRC,PUR(c,s,u,d)};
(all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST)
            SRCSHR(c,s,u,d) = PUR(c,s,u,d)/PUR_S(c,u,d);
(all,u,USR)(all,d,DST)
            PUR_CS(u,d)=sum{c,COM,PUR_S(c,u,d)};
Zerodivide off;

Zerodivide default 0.0;
Formula
(all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST)
            TAXRATE(c,s,u,d) = TAX(c,s,u,d)/USE(c,s,u,d);
Zerodivide off;
Write TAXRATE to file SUMMARY header "TRAT";

!**********************************************************************!
!Section 5.2 Formulas used for defining factor demands                 !
!**********************************************************************!

Formula
 (all,i,IND)(all,d,DST) LAB_O(i,d) = sum{o,OCC, LAB(i,o,d)};

Formula
 (all,i,IND)(all,d,DST) PRIM(i,d) = LAB_O(i,d)+ CAP(i,d) + LND(i,d);
```

```
Formula
(all,i,IND)(all,d,DST)
SHRLK(i,d,"labour")= LAB_O(i,d)/(LAB_O(i,d)+CAP(i,d)+0.0000000001);
(all,i,IND)(all,d,DST)
SHRLK(i,d,"capital")= CAP(i,d)/(LAB_O(i,d)+CAP(i,d)+.0000000001);

Formula
 (all,i,IND)(all,d,DST) PRIMCOST(i,"Lnd",d) = LND(i,d);
 (all,i,IND)(all,d,DST) PRIMCOST(i,"Lab",d) = LAB_O(i,d);
 (all,i,IND)(all,d,DST) PRIMCOST(i,"Cap",d) = CAP(i,d);
 Write PRIMCOST to file SUMMARY header "PCST";

Formula
 (all,i,IND)(all,d,DST)  VARCST(i,d)  = LAB_O(i,d)  + PUR_CS(i,d);
Formula
 (all,i,IND)(all,d,DST)  VCST(i,d)  = PRIM(i,d)  + PUR_CS(i,d);
Zerodivide default 1.0;
Formula
 (all,i,IND)(all,d,DST)  VTOT(i,d)    = VCST(i,d) + PRODTAX(i,d);
 (all,i,IND)(all,d,DST)  PTXRATE(i,d)  = PRODTAX(i,d)/VCST(i,d);
Zerodivide off;
Write PTXRATE to file SUMMARY header "PTXR";
Formula
 (all,i,IND)(all,d,DST) COSTMAT(i,"IntDom",d)  = sum{c,COM,
USE(c,"dom",i,d)};
 (all,i,IND)(all,d,DST) COSTMAT(i,"IntImp",d)  = sum{c,COM,
USE(c,"imp",i,d)};
 (all,i,IND)(all,d,DST)
                  COSTMAT(i,"ComTax",d)  = sum{c,COM, sum{s,SRC,
TAX(c,s,i,d)}};
 (all,i,IND)(all,d,DST) COSTMAT(i,"Lab",d)     = LAB_O(i,d);
 (all,i,IND)(all,d,DST) COSTMAT(i,"Cap",d)     = CAP(i,d);
 (all,i,IND)(all,d,DST) COSTMAT(i,"Lnd",d)     = LND(i,d);
 (all,i,IND)(all,d,DST) COSTMAT(i,"ProdTax",d) = PRODTAX(i,d);
Write COSTMAT to file SUMMARY header "CSTM";
Write VTOT    to file SUMMARY header "VTOT";

!************************************************************************!
!Section 5.3 Formulas used for defining household demands!
!************************************************************************!

Formula (all,c,COM)(all,d,DST)(all,h,HOU) HOUPUR(c,d,h) = PUR_S(c,"Hou",d);

Formula (all,d,DST)(all,h,HOU) HOUPUR_C(d,h) = sum{c,COM, HOUPUR(c,d,h)};
```

```
Formula
 (all,c,COM)(all,d,DST)(all,h,HOU) BUDGSHR(c,d,h) =
HOUPUR(c,d,h)/HOUPUR_C(d,h);
 (all,d,DST)(all,h,HOU) EPSAVE(d,h) = sum{c,COM, EPSH(c,d,h)*BUDGSHR(c,d,h)};
Formula
(initial) (all,c,COM)(all,d,DST)(all,h,HOU)
EPSH(c,d,h)=EPSH(c,d,h)/EPSAVE(d,h);
 (all,c,COM)(all,d,DST)(all,h,HOU)  BLUX(c,d,h) =
ABS[EPSH(c,d,h)/FRISCHH(d,h)];
 (all,c,COM)(all,d,DST)(all,h,HOU)  SLUX(c,d,h) = EPSH(c,d,h)*BUDGSHR(c,d,h);
Write  SLUX    to file SUMMARY header "LSHR";
       BUDGSHR to file SUMMARY header "BSHR";
       EPSAVE  to file SUMMARY header "AVEP";

Formula  (initial)(all,d,DST)(all,h,HOU) RATIOCPI(d,h) = 1;

Formula     DEFHOUSHR = 1/sum{h,HOU,1};
Zerodivide default DEFHOUSHR;
Formula (all,c,COM)(all,d,DST) HOUPUR_H(c,d) = sum{h,HOU, HOUPUR(c,d,h)};


Formula (all,c,COM)(all,d,DST)(all,h,HOU)
           HOUSHR(c,d,h)=HOUPUR(c,d,h)/HOUPUR_H(c,d);
Zerodivide off;

Formula   (all,c,COM)(all,d,DST)
   CHKHOUPUR(c,d) = HOUPUR_H(c,d) - PUR_S(c,"Hou",d);
Write CHKHOUPUR to file SUMMARY header "CHOU";
Formula   (all,c,COM)(all,d,DST)
   CHKHOUPUR(c,d) = 200*CHKHOUPUR(c,d)/ID01[HOUPUR_H(c,d) +
PUR_S(c,"Hou",d)];
Write CHKHOUPUR to file SUMMARY header "CHO2" longname "HOUPUR error as %";
Assertion # Check HOUPUR_H = PUR_S("Hou") #  (all,c,COM)(all,d,DST)
   ABS[CHKHOUPUR(c,d)]<0.1;

!************************************************************************!
!Section 5.4 Formulas used for defining                                  !
!Investment demands and indices                                           !
!************************************************************************!

Formula
(all,c,COM)(all,d,DST) INVEST_I(c,d) = sum{i,IND,INVEST(c,i,d)};
(all,i,IND)(all,d,DST) INVEST_C(i,d) = sum{c,COM,INVEST(c,i,d)};
```

```
!*************************************************************************!
!Section 5.5 Formulas used for defining                                  !
!total regional demands for delivered goods                              !
!*************************************************************************!

Formula     (all,c,COM)(all,s,SRC)(all,d,DST)
   USE_U(c,s,d) = sum{u,USR, USE(c,s,u,d)};
Formula     (all,c,COM)(all,s,SRC)(all,d,DST)
   USE_I(c,s,d) = sum{i,IND, USE(c,s,i,d)};

Formula
(all,c,COM)(all,s,SRC)(all,d,DST) LOCUSE(c,s,d) = USE_U(c,s,d)-
USE(c,s,"exp",d);
(all,c,COM)(all,d,DST)            LOCUSE_S(c,d) = sum{s,SRC, LOCUSE(c,s,d)};
(all,c,COM)                       LOCUSE_SD(c)  = sum{d,DST, LOCUSE_S(c,d)};


Zerodivide default 0; ! alternative: DEFSHR !

Formula     (all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
         DELIVRD(c,s,r,d)= TRADE(c,s,r,d) + sum{m,MAR, TRADMAR(c,s,m,r,d)};

Formula
(all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
 BASSHR(c,s,r,d) = TRADE(c,s,r,d)/DELIVRD(c,s,r,d);
(all,c,COM)(all,s,SRC)(all,m,MAR)(all,r,ORG)(all,d,DST)
 MARSHR(c,s,m,r,d) = TRADMAR(c,s,m,r,d)/DELIVRD(c,s,r,d);
Zerodivide off;

Formula (all,c,COM)(all,s,SRC)(all,d,DST)
  DELIVRD_R(c,s,d) = sum{r,ORG,DELIVRD(c,s,r,d)};



!*************************************************************************!
!Section 5.6 Formulas used for defining margin production and demands
!
!*************************************************************************!

Formula
 (all,m,MAR)(all,r,ORG)(all,d,DST)
    TRADMAR_CS(m,r,d) = sum{c,COM, sum{s,SRC, TRADMAR(c,s,m,r,d)}};
 (all,m,MAR)(all,r,ORG)(all,d,DST)
```

```
   SUPPMAR_P(m,r,d)  = sum{p,PRD, SUPPMAR(m,r,d,p)};
 (all,m,MAR)(all,r,ORG)(all,p,PRD)
   SUPPMAR_D(m,r,p)  = sum{d,DST, SUPPMAR(m,r,d,p)};
 (all,m,MAR)(all,p,PRD)
   SUPPMAR_RD(m,p)   = sum{r,ORG, SUPPMAR_D(m,r,p)};
 (all,m,MAR)(all,d,DST)(all,p,PRD)
   SUPPMAR_R(m,d,p)  = sum{r,ORG, SUPPMAR(m,r,d,p)};

!***********************************************************************!
!Section 5.7 Formulas used for multi-production (MAKE)          !
!***********************************************************************!
Formula
 (all,i,IND)(all,d,DST) MAKE_C(i,d) = sum{c,COM, MAKE(c,i,d)};
 (all,c,COM)(all,d,DST) MAKE_I(c,d) = sum{i,IND, MAKE(c,i,d)};
Coefficient RECIPNCOM;
Formula     RECIPNCOM = 1/sum{c,COM,1};
Coefficient RECIPNIND;
Formula     RECIPNIND = 1/sum{i,IND,1};
Zerodivide default RECIPNCOM;
Formula
(all,c,COM)(all,i,IND)(all,d,DST) MAKESHR1(c,i,d) = MAKE(c,i,d)/MAKE_C(i,d);
Zerodivide default RECIPNIND;
Formula
(all,c,COM)(all,i,IND)(all,d,DST) MAKESHR2(c,i,d) = MAKE(c,i,d)/MAKE_I(c,d);
Zerodivide off;


Formula
(all,c,COM)(all,i,IND) MAKE_D(c,i) = sum{d,DST, MAKE(c,i,d)};
(initial) (all,c,COM)  NINDPROD(c) = sum{i,IND:MAKE_D(c,i)<>0,1};
(initial) (all,i,IND)  NCOMPROD(i) = sum{c,COM:MAKE_D(c,i)<>0,1};

Set
 MIND # Multi-product industries #   = (all,i,IND:NCOMPROD(i)>1);
 SIND # Single-product industries #  =  IND - MIND;
 MINDCOM # Commodities produced by any industry in MIND #
     = (all,c,COM:sum{i,MIND,ABS[MAKE_D(c,i)]}>0);
Mapping SIND2COM from SIND to COM;
Formula (all,i,SIND) SIND2COM(i) = sum{c,COM:ABS[MAKE_D(c,i)]>0,
$pos(c,COM)};

Set
 MCOM # Commodities made by several industries #   =
(all,c,COM:NINDPROD(c)>1);
```

```
 SCOM # Commodities made by only one industry #  =  COM - MCOM;
 MCOMIND # Industries that produce a commodity in MCOM #
     = (all,i,IND:sum{c,MCOM,ABS[MAKE_D(c,i)]}>0);
Mapping SCOM2IND from SCOM to IND;
Formula (all,c,SCOM) SCOM2IND(c) = sum{i,IND:ABS[MAKE_D(c,i)]>0,
$pos(i,IND)};


Write !debug output !
 (Set) MIND to file SUMMARY header "MIND";
 (Set) SIND to file SUMMARY header "SIND";
 (Set) MCOM to file SUMMARY header "MCOM";
 (Set) SCOM to file SUMMARY header "SCOM";
 (Set) MCOMIND to file SUMMARY header "MCI";
 (Set) MINDCOM to file SUMMARY header "MIC";
 (by_elements) SCOM2IND to file SUMMARY header "SC2I";
 (by_elements) SIND2COM to file SUMMARY header "SI2C";

COEFFICIENT (all,c,COM)(all,i,IND) ISMADE(c,i);

Formula (all,c,COM)(all,i,IND) ISMADE(c,i)=0;
Formula (all,c,COM)(all,i,IND) ISMADE(c,i)=if(MAKE_d(c,i)>0,1);


COEFFICIENT (all,c,COM)(all,i,IND) MAKESHR_i(c,i);

Formula (all,c,COM)(all,i,IND)
MAKESHR_I(c,i)=0;

Formula (all,c,COM)(all,i,IND)
MAKESHR_I(c,i)=if((sum(cc,COM,MAKE_d(cc,i))>0),
MAKE_d(c,i)/(sum(cc,COM,MAKE_d(cc,i))));


!**************************************************************************!
!Section 5.8 Formulas used for defining market equilibrium                 !
!Total demand for commodity c produced in r=supply commodity c produced in r!
!**************************************************************************!

Formula
(all,c,COM)(all,s,SRC)(all,r,ORG) TRADE_D(c,s,r) = sum{d,DST,
TRADE(c,s,r,d)};
(all,c,COM)(all,s,SRC)(all,d,DST) TRADE_R(c,s,d) = sum{r,ORG,
TRADE(c,s,r,d)};
```

```
(all,c,COM)(all,s,SRC)           TRADE_RD(c,s)  = sum{r,ORG,
TRADE_D(c,s,r)};

!***************************************************************************!
!Section 5.9 Formulas used for defining                                     !
!Sectoral contributions to regional GDP at factor cost                      !
!***************************************************************************!

Formula     (all,d,DST) PRIM_I(d) = sum{i,IND, PRIM(i,d)};
   (initial)(all,d,DST) RATIOPRIM_I(d)= 1;
Formula     (all,i,IND)(all,d,DST) PRIMSHR(i,d) = PRIM(i,d)/PRIM_I(d);

Formula
 (all,o,OCC)(all,d,DST) LAB_I(o,d) = sum{i,IND, LAB(i,o,d)};
 (all,d,DST) LAB_IO(d) = sum{o,OCC, LAB_I(o,d)};
 (all,d,DST) LND_I(d)  = sum{i,IND, LND(i,d)};
 (all,d,DST) CAP_I(d)  = sum{i,IND, CAP(i,d)};
 (all,o,OCC) LAB_ID(o) = sum{d,DST, LAB_I(o,d)};

Formula
 (all,i,IND)(all,o,OCC)(all,d,DST) SLAB_I(i,o,d) = LAB(i,o,d)/LAB_I(o,d);

Formula
 (all,o,OCC)(all,d,DST) SLAB_ID(o,d) = LAB_I(o,d)/LAB_ID(o);

!***************************************************************************!
!Section 5.10 Formulas for GDP aggregates                                   !
!***************************************************************************!
! Nominal Income-side GDP !

Set GDPINCCAT # GDP income categories # (Land,Labour,Capital,PRODTAX,ComTax);
Coefficient (all,i,GDPINCCAT)(all,d,DST) GDPINCSUM(d,i)# Income GDP breakdown
#;
Formula     (all,i,GDPINCCAT)(all,d,DST) GDPINCSUM(d,i) = 0;
(all,d,DST) GDPINCSUM(d,"Land")    = LND_I(d);
(all,d,DST) GDPINCSUM(d,"Labour")  = LAB_IO(d);
(all,d,DST) GDPINCSUM(d,"Capital") = CAP_I(d);
(all,d,DST) GDPINCSUM(d,"ProdTax") = sum{i,IND,PRODTAX(i,d)};
(all,d,DST) GDPINCSUM(d,"ComTax")  =
              sum{u,USR, sum{c,COM,sum{s,SRC,TAX(c,s,u,d)}}};
Write GDPINCSUM to file SUMMARY header "GNSM";

Coefficient (all,d,DST) GDPINC(d)# Income GDP #;
Formula     (all,d,DST) GDPINC(d) = sum{i,GDPINCCAT, GDPINCSUM(d,i)};
```

```
Set GDPEXPCAT (HOU, INV, GOV, STOCKS, EXP, Imports, RExports, RImports,
NetMar);
Subset FINDEM is subset of GDPEXPCAT;
Coefficient (all,i,GDPEXPCAT)(all,q,DST) GDPEXPSUM(q,i) # Expend GDP
breakdown#;
Formula     (all,i,GDPEXPCAT)(all,q,DST) GDPEXPSUM(q,i) = 0;
(all,q,DST)(all,f,FINDEM) GDPEXPSUM(q,f) = sum{c,COM, PUR_S(c,f,q)};
(all,q,DST) GDPEXPSUM(q,"Stocks") = sum{i,IND, STOCKS(i,q)};
(all,q,REG) GDPEXPSUM(q,"Imports") =
     -sum{c,COM, sum{d,DST, TRADE(c,"imp",q,d)}};
(all,q,REG) GDPEXPSUM(q,"NetMar") = sum{m,MAR, sum{r,ORG,
       SUPPMAR_D(m,r,q) - SUPPMAR_P(m,r,q) }};
(all,q,REG) GDPEXPSUM(q,"Rexports") =
     sum{c,COM,sum{s,SRC, TRADE_D(c,s,q) - TRADE(c,s,q,q)}};
(all,q,REG) GDPEXPSUM(q,"Rimports") =
    - sum{c,COM,sum{s,SRC, TRADE_R(c,s,q) - TRADE(c,s,q,q)}};

Write GDPEXPSUM to file SUMMARY header "GESM";

Coefficient (all,d,DST) GDPEXP(d)# Expenditure GDP #;
Formula     (all,d,DST) GDPEXP(d) = sum{i,GDPEXPCAT, GDPEXPSUM(d,i)};

!Expenditure side GDP       !

Set NATGDPEXPCAT (HOU, INV, GOV, STOCKS, EXP, Imports);
Subset NATGDPEXPCAT is subset of GDPEXPCAT;
Subset FINDEM is subset of NATGDPEXPCAT;

Coefficient (all,i,NATGDPEXPCAT) NATGDPEXPSUM(i) # Expend GDP breakdown#;
Formula (all,i,NATGDPEXPCAT) NATGDPEXPSUM(i)=sum{q,DST, GDPEXPSUM(q,i)};
Write NATGDPEXPSUM to file SUMMARY header "NGSM";

Coefficient NATGDPEXP # Expenditure GDP #;
Formula     NATGDPEXP = sum{d,DST, GDPEXP(d)};

Coefficient  RATIONGDPEXP# (Current/initial) real expenditure GDP#;
Formula  (initial) RATIONGDPEXP = 1;


! Investment ratio summary -- identify outlying INV/RENTAL ratios !

Formula (all,r,REGPLUS)(all,i,INDPLUS)(all,z,INVFACT) INVFACTS(r,i,z)=0;
 (all,r,REG)(all,i,IND) INVFACTS(r,i,"Invest")    = INVEST_C(i,r);
```

```
 (all,r,REG)(all,i,IND) INVFACTS(r,i,"CapRental") = CAP(i,r);
 (all,r,REG)(all,i,IND) INVFACTS(r,i,"Output")    = VTOT(i,r);
 (all,i,IND)(all,z,INVFACT) INVFACTS_r(i,z) = sum{r,REG, INVFACTS(r,i,z)};
 (all,i,IND)(all,z,INVFACT) INVFACTS("National",i,z) = INVFACTS_r(i,z);
 (all,r,REGPLUS)(all,z,INVFACT) INVFACTS_I(r,z) = sum{i,IND,
INVFACTS(r,i,z)};
 (all,r,REGPLUS)(all,z,INVFACT) INVFACTS(r,"TotalInds",z) = INVFACTS_I(r,z);
(all,r,REGPLUS)(all,i,INDPLUS) INVFACTS(r,i,"InvOverCap") =
               INVFACTS(r,i,"Invest")/ID01[INVFACTS(r,i,"CapRental")];
(all,r,REGPLUS)(all,i,INDPLUS) INVFACTS(r,i,"InvOverOut") =
               INVFACTS(r,i,"Invest")/ID01[INVFACTS(r,i,"Output")];
 Write INVFACTS to file SUMMARY header "INVF";



! Regional macro reporting variables !


Formula
(all,s,SRC)(all,d,DST) TRADE_CR(s,d) = sum{c,COM, TRADE_R(c,s,d)};
(all,d,DST) IMPUSED_C(d) = TRADE_CR("imp",d);
(all,r,ORG) IMPLANDED_C(r) = sum{c,COM, TRADE_D(c,"imp",r)};

!***************************************************************************!
!Section 5.11 Formulas used for defining national aggeragates               !
!Aggregation of regional to national industry output -- value added weights !
!***************************************************************************!

Formula (all,i,IND) PRIM_D(i) = sum{d,DST, PRIM(i,d)};
Formula (initial)(all,i,IND) RATIOPRIM_D(i) =  1;
Formula
 (all,c,COM) MAKE_IR(c) = sum{r,REG, MAKE_I(c,r)};

!National price and employment variables !

Formula     (all,i,IND) NATVTOT(i) = sum{d,DST,VTOT(i,d)};
            (all,i,IND) LAB_OD(i)  = sum{d,DST,LAB_O(i,d)};
```

```
!***************************************************************************!
!Section 5.12 Check of accounting identities                               !
!***************************************************************************!

Zerodivide default 1;

Coefficient
 EP # Tiny value #;
 (all,i,IND)(all,d,DST) CHECKA(i,d) # Net Output - MAKE_C #;
 (all,i,IND)(all,d,DST) CKRATA(i,d) # Net Output / MAKE_C #;

Formula
EP=0.00001;
(all,i,IND)(all,d,DST) CHECKA(i,d) = [VTOT(i,d) - STOCKS(i,d)] - MAKE_C(i,d);
(all,i,IND)(all,d,DST) CKRATA(i,d) =
                        [EP+VTOT(i,d) - STOCKS(i,d)] / [EP+MAKE_C(i,d)];
Write
 MAKE_C to file SUMMARY header "MAKC";
 STOCKS to file SUMMARY header "STOK";
 CHECKA to file SUMMARY header "CHKA";
 CKRATA to file SUMMARY header "CKRA";

Coefficient
(all,c,COM)(all,s,SRC)(all,d,DST) CHECKB(c,s,d) # USE_U - DELIVRD_R #;
(all,c,COM)(all,s,SRC)(all,d,DST) CKRATB(c,s,d) # USE_U / DELIVRD_R #;

Formula
(all,c,COM)(all,s,SRC)(all,d,DST) CHECKB(c,s,d) = USE_U(c,s,d)-
DELIVRD_R(c,s,d);
(all,c,COM)(all,s,SRC)(all,d,DST) CKRATB(c,s,d) =
                            [EP+USE_U(c,s,d)]/[EP+DELIVRD_R(c,s,d)];
Write
 USE_U  to file SUMMARY header "USEU";
 DELIVRD_R  to file SUMMARY header "DVDR";
 CHECKB to file SUMMARY header "CHKB";
 CKRATB to file SUMMARY header "CKRB";

Coefficient
 (all,c,COM)(all,r,REG) CHECKC(c,r) # MAKE_I - demands #;
 (all,c,COM)(all,r,REG) CKRATC(c,r) # MAKE_I / demands #;
 (all,c,COM)(all,r,REG) DEMANDS(c,r) # Demands #;
Formula
 (all,c,COM)(all,r,REG) DEMANDS(c,r) = TRADE_D(c,"dom",r);
```

```
 (all,m,MAR)(all,r,REG) DEMANDS(m,r) = DEMANDS(m,r) + SUPPMAR_RD(m,r);
 (all,c,COM)(all,r,REG) CHECKC(c,r)  = MAKE_I(c,r) - DEMANDS(c,r);
 (all,c,COM)(all,r,REG) CKRATC(c,r)  = [EP+MAKE_I(c,r)] / [EP+DEMANDS(c,r)];
Write
 CHECKC to file SUMMARY header "CHKC";
 CKRATC to file SUMMARY header "CKRC";
 MAKE_I to file SUMMARY header "MAKI";
 DEMANDS to file SUMMARY header "DMDS";

Coefficient
(all,m,MAR)(all,r,ORG)(all,d,DST) CHECKD(m,r,d) # TRADMAR_CS - SUPPMAR_P #;
(all,m,MAR)(all,r,ORG)(all,d,DST) CKRATD(m,r,d) # TRADMAR_CS / SUPPMAR_P #;
Formula
 (all,m,MAR)(all,r,ORG)(all,d,DST)
   CHECKD(m,r,d) = TRADMAR_CS(m,r,d) - SUPPMAR_P(m,r,d);
 (all,m,MAR)(all,r,ORG)(all,d,DST)
   CKRATD(m,r,d) = [EP+TRADMAR_CS(m,r,d) ]/ [EP+SUPPMAR_P(m,r,d)];
Write
 TRADMAR_CS to file SUMMARY header "TMCS";
 SUPPMAR_P  to file SUMMARY header "SPMP";
 CHECKD     to file SUMMARY header "CHKD";
 CKRATD     to file SUMMARY header "CKRD";

Coefficient
(all,c,COM)(all,d,DST) CHECKE(c,d) # INVEST_I - PUR_S(inv) #;
(all,c,COM)(all,d,DST) CKRATE(c,d) # INVEST_I / PUR_S(inv) #;
Formula (all,c,COM)(all,d,DST) CHECKE(c,d) = INVEST_I(c,d) -
PUR_S(c,"Inv",d);
 (all,c,COM)(all,d,DST) CKRATE(c,d)   =
                      [EP+INVEST_I(c,d)] / [EP+PUR_S(c,"Inv",d)];
Write
 INVEST_I to file SUMMARY header "INVI";
 PUR_S    to file SUMMARY header "PURS";
 CHECKE   to file SUMMARY header "CHKE";
 CKRATE   to file SUMMARY header "CKRE";
Zerodivide off;

Set CONSTRAINTS (A,B,c,d,E);
Coefficient (all,c,CONSTRAINTS) SUMABS(c) # Sum absolute errors #;
Formula
 SUMABS("A") = sum{i,IND, sum{d,DST,            ABS[CHECKA(i,d)]}};
 SUMABS("B") = sum{c,COM, sum{s,SRC, sum{d,DST, ABS[CHECKB(c,s,d)]}}};
 SUMABS("C") = sum{c,COM, sum{r,REG,            ABS[CHECKC(c,r)]}};
 SUMABS("D") = sum{m,MAR, sum{r,ORG, sum{d,DST, ABS[CHECKD(m,r,d)]}}};
```

```
 SUMABS("E") = sum{c,COM, sum{d,DST,          ABS[CHECKE(c,d)]}};
Write SUMABS to file SUMMARY header "SMAB";

Coefficient TOL3;
Formula     TOL3 = 0.01;
!Assertion
# RATA near 1 # (all,i,IND)(all,d,DST)           ABS[1-CKRATA(i,d)]   <
TOL3;
# RATB near 1 # (all,c,COM)(all,s,SRC)(all,d,DST) ABS[1-CKRATB(c,s,d)] <
TOL3;
# RATC near 1 # (all,c,COM)(all,r,REG)           ABS[1-CKRATC(c,r)]   <
TOL3;
# RATD near 1 # (all,m,MAR)(all,r,ORG)(all,d,DST) ABS[1-CKRATD(m,r,d)] <
TOL3;
# RATE near 1 # (all,c,COM)(all,d,DST)           ABS[1-CKRATE(c,d)]   <
TOL3;!
! alternate form, which does not check ratio values where diff is tiny
# RATA near 1 # (all,i,IND)(all,d,DST:ABS[CHECKA(i,d)]>0.1)
            ABS[1-CKRATA(i,d)]   < TOL3;
# RATB near 1 # (all,c,COM)(all,s,SRC)(all,d,DST:ABS[CHECKB(c,s,d)]>0.1)
            ABS[1-CKRATB(c,s,d)] < TOL3;
# RATC near 1 # (all,c,COM)(all,r,REG:ABS[CHECKC(c,r)]>0.1)
            ABS[1-CKRATC(c,r)]   < TOL3;
# RATD near 1 # (all,m,MAR)(all,r,ORG)(all,d,DST:ABS[CHECKD(m,r,d)]>0.1)
            ABS[1-CKRATD(m,r,d)] < TOL3;
# RATE near 1 # (all,c,COM)(all,d,DST:ABS[CHECKE(c,d)]>0.1)
            ABS[1-CKRATE(c,d)]   < TOL3; !


!*****************************************************************************!
!Section 5.13 Formulas used for defining trade flows!
!*****************************************************************************!

!Trade elasticities!

Formula (all,c,COM)(all,e,ELSTRAD) TRADELAST(c,e) = 0;
 (all,c,COM) TRADELAST(c,"EXP_ELAST")    = EXP_ELAST(c);
 (all,c,COM) TRADELAST(c,"SIGMADOMIMP") = SIGMADOMIMP(c);
 (all,c,COM) TRADELAST(c,"SIGMADOMDOM") = SIGMADOMDOM(c);
Write TRADELAST to file SUMMARY header "TREL";

!Foreign trade!

Formula
```

```
(all,c,COM)(all,r,ORG)(all,d,DST) ROWDEM(c,r,d)
   = TRADE(c,"dom",r,d)*USE(c,"dom","Exp",d)/ID01[USE_U(c,"dom",d)];
(all,c,COM)(all,r,ORG) ROWDEM_D(c,r) = sum{d,DST, ROWDEM(c,r,d)};
(all,c,COM)(all,r,REG) EXPSHR(c,r) = ROWDEM_D(c,r)/[0.001+MAKE_I(c,r)];

Write ROWDEM to file SUMMARY header "ROWD";
Write EXPSHR to file SUMMARY header "SHXP";


!domestic trade flows!


Formula
(all,c,COM)(all,r,REG)
VDOMEXP(c,r) = sum{d,DST, TRADE(c,"dom",r,d)} - TRADE(c,"dom",r,r);
(all,c,COM)(all,d,REG)
VDOMIMP(c,d) = sum{r,ORG, TRADE(c,"dom",r,d)} - TRADE(c,"dom",d,d);
(all,r,REG) VDOMEXP_C(r) = sum{c,COM,VDOMEXP(c,r)};
(all,d,REG) VDOMIMP_C(d) = sum{c,COM,VDOMIMP(c,d)};

Write VDOMEXP to file SUMMARY header "VDEX";
Write VDOMIMP to file SUMMARY header "VDIM";


!*****************************************************************************!
!Section 5.14 Summary files!
!*****************************************************************************!

! Local use for summary file !

Coefficient
(all,c,COM)(all,s,SRC)(all,m,MAINUSR)(all,d,DST) VMAINUSE(c,s,m,d)
# Simplified USE matrix #;

Formula
(all,c,COM)(all,s,SRC)(all,d,DST)
   VMAINUSE(c,s,"INT",d)= sum{i,IND, USE(c,s,i,d)};
(all,c,COM)(all,s,SRC)(all,m,FINDEM)(all,d,DST)
   VMAINUSE(c,s,m,d) = USE(c,s,m,d);
Write VMAINUSE to file SUMMARY header "VMNU";

! Local (and other) sales table for summary file !
Coefficient (all,c,COM)(all,s,SRC)(all,r,REG)
    LOCSHR(c,s,r) # Share local-made basic in direct use (delivered) #;
```

```
Formula     (all,c,COM)(all,s,SRC)(all,r,REG)
    LOCSHR(c,s,r) = TRADE(c,s,r,r)/ID01[USE_U(c,s,r)];
Write LOCSHR to file SUMMARY header "LCSR";

Coefficient (all,c,COM)(all,s,SRC)(all,r,REG)
    LOCSHR_R(c,s,r) # Share basic in delivered #;
Formula     (all,c,COM)(all,s,SRC)(all,r,REG)
    LOCSHR_R(c,s,r) = TRADE_R(c,s,r)/ID01[USE_U(c,s,r)];
Write LOCSHR_R to file SUMMARY header "LCRR";

Set  MAINDEST # Main destinations #  (INT, HOU, INV, GOV, MAR, OthReg,
RowEXP);
Set  MAINDESTX # Local direct main destinations #  (INT, HOU, INV, GOV);
Subset  MAINDESTX is subset of MAINDEST;
Subset  MAINDESTX is subset of MAINUSR;

Coefficient
(all,c,COM)(all,r,REG)(all,d,MAINDEST) VMAINDEST(c,r,d)
# Main uses of dom com c made in r #;
Formula
(all,c,COM)(all,r,REG)(all,d,MAINDEST) VMAINDEST(c,r,d) = 42;
(all,c,COM)(all,r,REG)(all,d,MAINDESTX)
   VMAINDEST(c,r,d) = LOCSHR(c,"dom",r)*VMAINUSE(c,"dom",d,r);
(all,c,NONMAR)(all,r,REG)VMAINDEST(c,r,"MAR") = 0;
(all,m,MAR)(all,p,REG)VMAINDEST(m,p,"MAR") =
   sum{r,ORG, sum{d,DST, SUPPMAR(m,r,d,p)}};
(all,c,COM)(all,r,REG)VMAINDEST(c,r,"OthReg") =
   TRADE_D(c,"dom",r)-TRADE(c,"dom",r,r);
(all,c,COM)(all,r,REG)VMAINDEST(c,r,"RowEXP") =
   LOCSHR(c,"dom",r)*USE(c,"dom","Exp",r);
Write VMAINDEST to file SUMMARY header "VMDS";

Coefficient (all,c,COM)(all,d,DST) CHECKDEST(c,d) # Check -- should be tiny
#;
Formula     (all,c,COM)(all,d,DST) CHECKDEST(c,d) = 42;
 (all,c,COM)(all,d,REG)
  CHECKDEST(c,d) = MAKE_I(c,d) -  sum{q,MAINDEST,VMAINDEST(c,d,q)};
Write CHECKDEST to file SUMMARY header "CDST";

Coefficient (all,c,COM)(all,d,MAINDEST)
   NATMAINDEST(c,d) # Main uses of dom com c nationally #;
Formula
(all,c,COM)(all,d,MAINDEST) NATMAINDEST(c,d) = 42; ! should be overwritten !
(all,c,COM)(all,d,MAINDESTX)
```

```
    NATMAINDEST(c,d) = sum{r,REG, LOCSHR_R(c,"dom",r)*VMAINUSE(c,"dom",d,r)};
(all,c,NONMAR) NATMAINDEST(c,"MAR") = 0;
(all,m,MAR) NATMAINDEST(m,"MAR") = sum{p,REG,VMAINDEST(m,p,"MAR")};
(all,c,COM) NATMAINDEST(c,"OthReg") = 0;
(all,c,COM) NATMAINDEST(c,"RowEXP") =
           sum{r,REG, LOCSHR_R(c,"dom",r)*VMAINUSE(c,"dom","EXP",r)};
Write NATMAINDEST to file SUMMARY header "NMDS";

Coefficient (all,c,COM)(all,d,MAINDEST) DIFFDEST(c,d);
Formula     (all,c,COM)(all,d,MAINDEST) DIFFDEST(c,d)
              = sum{r,REG,VMAINDEST(c,r,d)} - NATMAINDEST(c,d);
Write DIFFDEST to file SUMMARY header "DDST";

Coefficient (all,d,DST)(all,f,FINDEM) RATFINPUR(f,d);
Formula     (all,d,DST)(all,f,FINDEM) RATFINPUR(f,d) = PUR_CS(f,d);
Write RATFINPUR to file SUMMARY header "FPUR"
  longname "final demands, pur value";
Formula     (all,d,DST)(all,f,FINDEM) RATFINPUR(f,d) =
RATFINPUR(f,d)/PRIM_I(d);
Write RATFINPUR to file SUMMARY header "RPUR"
  longname "[final demands, pur value]/reg gdp";

! Calculation of Macros for broad regions (optional, but useful for Brazil)
Set AGGREG # Broad regions # read elements from file REGSUPP header "AREG";
Mapping MAPREG from REG to AGGREG;
Read (by_elements) MAPREG from file REGSUPP header "MREG";
Variable (all,m,MAINMACROS)(all,A,AGGREG) ZoneMacro(m,A) # Broad region
macros#;
Equation E_ZoneMacro (all,m,MAINMACROS)(all,A,AGGREG)
sum{q,REG:MAPREG(q)=A, WMAINMACRO(m,q)*[ZoneMacro(m,A)-MainMacro(m,q)]} = 0;!


!**************************************************************************!
!Section 5.15 Formulas for public sectors!
!**************************************************************************!

Formula
(all,d,DST) UN_RATE(d) = [LAB_SUP(d)-EMPLOYS(d)]/LAB_SUP(d);
(all,d,DST) EMP_RATE(d) = EMPLOYS(d)/LAB_SUP(d);
            TINY        = 0.000000000001;

(all,I,IND)(all,d,DST)(all,s,PSEC)
COL_PAYROLLS(i,d,s) = [LAB_O(i,d)/POW_PAYROLL(i,d)]*[POW_PAYROLLS(s,i,d) -1];
(all,d,DST)(all,s,PSEC)
```

```
COL_PAYRTOTS(d,s) = sum{i,IND,COL_PAYROLLS(i,d,s)};



Formula
(all,d,DST)(all,s,PSEC) TAXS_LAB(d,S) =
    TAXS_L_RATE(S,d)*[LAB_IO(d)-sum(p,PSEC,COL_PAYRTOTS(d,p))];
(all,d,DST)(all,s,PSEC) TAXS_CAP(d,S)     = TAXS_K_RATE(S,d)*CAP_I(d);
(all,d,DST)(all,s,PSEC) TAXS_LND(d,S)     = TAXS_K_RATE(s,d)*LND_I(d);
(all,d,DST)(all,s,PSEC) TAXS_AB(d,S)=
                            TAXS_AB_RATE(S,d)*(sum[z,PSEC,AGEBENS(z,d)]);
(all,d,DST)(all,s,PSEC) TAXS_OB(d,S)=
                            TAXS_OB_RATE(S,d)*(sum[z,PSEC,OTHBENS(z,d)]);
(all,d,DST)(all,s,PSEC) TAXS_UB(d,S)=
                            TAXS_UB_RATE(S,d)*(sum[z,PSEC,UNEMPBENS(z,d)]);

Formula
(all,d,DST) TAXINT(d)  = sum{c,COM, sum{s,SRC, sum{i,IND, TAX(c,s,i,d)}}};
(all,d,DST) TAXHOU(d)  = sum{c,COM, sum{s,SRC, TAX(c,s,"hou",d)}};
(all,d,DST) TAXINV(d)   = sum{c,COM, sum{s,SRC, TAX(c,s,"inv",d)}};
(all,d,DST) TAXGOV(d)    = sum{c,COM, sum{s,SRC, TAX(c,s,"gov",d)}};
(all,d,DST) TAXEXP(d)   = sum{c,COM, sum{s,SRC, TAX(c,s,"exp",d)}};
(all,d,DST) TAXPROD(d)   = sum{i,IND, PRODTAX(i,d)};

Formula
(all,d,DST)  TAX_CSI(d) = TAXINT(d) + TAXINV(d) + TAXHOU(d)
            + TAXGOV(d) + TAXEXP(d) + TAXPROD(d);

(all,d,DST)(all,s,PSEC) V0TAX_CSIS(d,S)=0;
(all,d,DST)V0TAX_CSIS(d,"S1311")=TAX_CSI(d);


(all,d,DST)(all,s,PSEC)
INCTAXS(d,S)      = TAXS_LAB(d,S)+TAXS_CAP(d,S)+TAXS_LND(d,S)+TAXS_AB(d,S)
                    +TAXS_OB(d,S) + TAXS_UB(d,s);

(all,d,DST)(all,s,PSEC)
NET_TAXTOTS(d,S) = V0TAX_CSIS(d,S) + INCTAXS(d,S) + COL_PAYRTOTS(d,S);

(initial) LEV_CPI_B    = LEV_CPI;
(initial) LEV_CPI_L_B  = LEV_CPI_L;

INF = LEV_CPI/LEV_CPI_L -1;
```

```
RINT_PSD = RINT;

(all,d,DST)  V1PRIM_I(d)  = LAB_IO(d) + CAP_I(d) + LND_I(d);
(all,d,DST)  V0GDPINC(d)  = V1PRIM_I(d) + TAX_CSI(d);


!Transfers between public sectors to DST to PSEC from PSEC!

Formula
(all,d,DST)(all,s,PSEC)(all,z,PSEC) TRANSPSECS(d,s,z)=0;
(all,d,DST)TRANSPSECS(d,"S1313","S1311")=GOVTOMUN(d);
(all,d,DST)TRANSPSECS(d,"S1314","S1311")=GOVTOSSF(d);
(all,d,DST)TRANSPSECS(d,"S1311","S1313")=MUNTOGOV(d);
(all,d,DST)TRANSPSECS(d,"S1311","S1314")=SSFTOGOV(d);


!Zerodivide default 0.5;!
Zerodivide default 0.00001;

Coefficient (all,d,DST)(all,i,IND) V2SHRSG_S_(d,i);

Formula (all,d,DST)(all,i,IND)

V2SHRSG_S_(d,i)=sum(s,PSEC,V2TOTSG_(d,i,s))/(sum{c,COM,INVEST(c,i,d)});


Coefficient (all,d,DST)(all,i,IND) V2SHRSG_S(d,i);

Formula (all,d,DST)(all,i,IND)

V2SHRSG_S(d,i)=if(V2SHRSG_S_(d,i)<1,V2SHRSG_S_(d,i))+
if(V2SHRSG_S_(d,i)>1,1);


(all,d,DST)(all,i,IND)(all,s,PSEC)

V2SHRSG(d,i,s)=V2TOTSG_(d,i,s)/sum(z,PSEC,V2TOTSG_(d,i,z));

(all,d,DST)(all,i,IND)(all,s,PSEC)

V2SHRSG_(d,i,s)=V2SHRSG(d,i,s)*V2SHRSG_S(d,i);

(all,d,DST)(all,s,PSEC)
V2TOT_G_IS(d,s)=sum{i,IND,V2SHRSG_(d,i,s)*(sum{c,COM,INVEST(c,i,d)})};
```

```
Formula (initial)(all,d,DST)(all,s,PSEC)
OTHCAPGOVS(s,D)=OTHCAPGOVS_(s,D)+sum(i,IND,V2TOTSG_(d,i,s))-V2TOT_G_IS(d,s);

(all,d,DST)(all,c,COM)(all,s,PSEC)
V5SHRS_(D,c,S)=V5TOTS_(D,c,S)/(sum{h,PSEC,V5TOTS_(d,c,h)});

(all,d,DST)(all,c,COM)(all,s,PSEC)
V5TOT(D,c,S)=
V5TOTS_(D,c,S)/(sum{h,PSEC,V5TOTS_(d,c,h)})*(sum{z,SRC,PUR(c,Z,"gov",d)});

(all,d,DST)(all,s,PSEC)
V5TOTS(d,s)=sum{c,COM,V5TOT(d,c,s)};

(all,d,DST)(all,c,COM)(all,r,SRC)(all,s,PSEC)
V5TOTR(D,c,r,S)=SRCSHR(c,r,"gov",d)*V5TOT(d,c,s);

(all,d,DST)(all,c,COM)(all,r,SRC)(all,s,PSEC)
SCRSHR5(d,c,r,s)=V5TOTR(D,c,r,S)/V5TOT(D,c,S);


Zerodivide off;


Read INT_PSD from file EXTRA header "NINT";

Formula
(all,d,DST)(all,s,PSEC)
PSDATTPLUS1(d,s) =
    {PSDATT(s,d)*[1+INT_PSD/2] + V5TOTS(d,s) +
    V2TOT_G_IS(d,s) + OTHCAPGOVS(s,D) - NET_TAXTOTS(d,s)- OTHGOVREVS(s,D)
            +   UNEMPBENS(s,d) + AGEBENS(S,d) + OTHBENS(S,d)
            +   GRANTSS(S,d)
            +(if(s eq "s1313",MUNTOGOV(d)-GOVTOMUN(d)))
            +(if(s eq "s1311",GOVTOMUN(d)-MUNTOGOV(d)+GOVTOSSF(d)-
SSFTOGOV(d)))
            +(if(s eq "s1314",SSFTOGOV(d)-GOVTOSSF(d)))
    }/[1-INT_PSD/2];

(all,d,DST)(all,s,PSEC)
NETINTS_G(s,d)    = INT_PSD*[PSDATT(s,d)+PSDATTPLUS1(d,s)]/2;

Coefficient NETINTS_GSD;
```

```
Formula
NETINTS_GSD    =
INT_PSD*[SUM(s,PSEC,sum(d,DST,PSDATT(s,d)+PSDATTPLUS1(d,s)))]/2;

Coefficient (all,s,PSEC) NETINTS_GD(s);

Formula
(all,s,PSEC)
NETINTS_GD(s)    =
INT_PSD*[sum(d,DST,PSDATT(s,d)+PSDATTPLUS1(d,s))]/2;


Coefficient (all,d,DST) SHRASS_H(d);

Read SHRASS_H from file extra header "SHRA";


Coefficient (all,d,DST) SHRPSD(d);

Formula
(all,d,DST)
SHRPSD(d)=
sum(s,PSEC,PSDATT(s,d))/[SUM(s,PSEC,sum(r,DST,PSDATT(s,r)))];

(all,d,DST)(all,s,PSEC)
NETINTS_HS(d,s)    = SHRASS_H(d)*NETINTS_GD(s);


(initial)(all,d,DST)(all,s,PSEC) PSDATT_B(d,s) = PSDATT(s,d);
(initial)(all,d,DST)(all,s,PSEC) PSDATT_1_B(d,s) = PSDATTPLUS1(d,s);

!TRANSFERS from public sectors to consumer!

(all,d,DST)(all,s,PSEC)
TRANSS_G(d,s) = UNEMPBENS(s,d) + AGEBENS(S,d) + OTHBENS(S,d)
            + GRANTSS(S,d) + NETINTS_G(S,d);

(all,d,DST)(all,s,PSEC)
TRANSS_H(d,s) = UNEMPBENS(s,d) + AGEBENS(S,d) + OTHBENS(S,d)
            + GRANTSS(S,d) + NETINTS_HS(d,s);

Formula

(all,d,DST)(all,s,PSEC)
```

```
GOV_DEFS(d,s) = V5TOTS(d,S) + V2TOT_G_IS(d,s)+ OTHCAPGOVS(s,d)
            - NET_TAXTOTS(d,s) - OTHGOVREVS(s,d) + TRANSS_G(d,s)
            +(if(s eq "s1313",MUNTOGOV(d)))
            -(if(s eq "s1313",GOVTOMUN(d)))
            +(if(s eq "s1311",GOVTOMUN(d)))
            +(if(s eq "s1311",GOVTOSSF(d)))
            -(if(s eq "s1311",MUNTOGOV(d)))
            -(if(s eq "s1311",SSFTOGOV(d)))
            +(if(s eq "s1314",SSFTOGOV(d)))
            -(if(s eq "s1314",GOVTOSSF(d)))
;


Formula
TAX_LAB= sum(d,DST,sum(s,PSEC, TAXS_LAB(d,S)));
TAX_CAP=sum(d,DST,sum(s,PSEC,TAXS_CAP(d,S)));
TAX_K_RATE=TAX_CAP/(sum(d,DST,CAP_I(d)));
(initial)
TAX_L_RATE=(sum(s,PSEC,sum(d,DST,TAXS_LAB(d,s))))/
(sum(d,DST,[LAB_IO(d)-sum(s,PSEC,COL_PAYRTOTS(d,s))]));
(initial)
TAX_L_RATE_O=TAX_L_RATE;
(initial)
TAX_L_RATE_L=TAX_L_RATE;
(initial)
TAX_L_RATE_O=TAX_L_RATE;
(initial)
TAX_L_R_O_L=TAX_L_RATE;

!****************************************************************************!
!Section 5.16 Formulas used for defining
 Income and saving aggregates!
!****************************************************************************!

Coefficient
            C_A0E  # Foreign assets, equity, start of year #;
            C_A1E  # Foreign assets, equity, end of year #;
(parameter) C_A0E_B  # Foreign assets, equity, start of year, base #;
(parameter) C_A1E_B  # Foreign assets, equity, end of year, base #;
            C_A0D  # Foreign assets, debt, start of year #;
            C_A1D  # Foreign assets, debt, end of year #;
(parameter) C_A0D_B   # Foreign assets, debt, start of year, base  #;
(parameter) C_A1D_B  # Foreign assets, debt, end of year, base  #;
            C_FE # Share of national savings devoted to foreign equity assets
```

```
#;
            C_FD # Share of national savings devoted to foreign debt assets
#;
            C_INTAD # Interest on foreign assets, debt #;
            C_INTAE # Interest on foreign assets, equity #;
            C_INTLD # Interest on foreign liabilities, debt #;
            C_INTLE # Interest on foreign liabilities, equity #;
            C_L0E  # Foreign liabilities, equity, start of year #;
            C_L1E  # Foreign liabilities, equity, end of year #;
(parameter) C_L0E_B  # Foreign liabilities, equity, start of year, base #;
(parameter) C_L1E_B  # Foreign liabilities, equity, end of year, base #;
            C_L0D  # Foreign liabilities, debt, start of year #;
            C_L1D  # Foreign liabilities, debt, end of year #;
(parameter) C_L0D_B   # Foreign liabilities, debt, start of year, base  #;
(parameter) C_L1D_B  # Foreign liabilities, debt, end of year, base  #;

            C_NEWLD # New foreign liabilities, debt #;
            C_NEWLE # New foreign liabilities, equity #;

            C_REVAE # Revaluation factor for foreign equity assets #;
            C_REVAD # Revaluation factor for foreign debt assets #;
            C_REVLE # Revaluation factor for foreign equity liabilities #;
            C_REVLD # Revaluation factor for foreign debt liabilities #;

            C_ROIAD # Rate of interest foreign assets, debt #;
            C_ROIAE # Rate of interest foreign assets, equity #;
            C_ROILD # Rate of interest foreign liabilities, debt #;
            C_ROILE # Rate of interest foreign liabilities, equity #;
            NETTRN # Net transfers, contributes to surplus #;


Read C_A0E from file BOPACC header "A0E";
Read C_A1E from file BOPACC header "A1E";
Read C_A0D from file BOPACC header "A0D";
Read C_A1D from file BOPACC header "A1D";
Read C_L0E from file BOPACC header "L0E";
Read C_L1E from file BOPACC header "L1E";
Read C_L0D from file BOPACC header "L0D";
Read C_L1D from file BOPACC header "L1D";
Read C_REVAE from file BOPACC header "RAE";
Read C_REVAD from file BOPACC header "RAD";
Read C_REVLD from file BOPACC header "RLD";
Read C_REVLE from file BOPACC header "RLE";
Read C_FE from file BOPACC header "FE";
```

```
Read C_FD from file BOPACC header "FD";

Read C_INTAD from file BOPACC header "INAD";
Read C_INTAE from file BOPACC header "INAE";
Read C_INTLD from file BOPACC header "INLD";
Read C_INTLE from file BOPACC header "INLE";
Read C_ROIAD from file BOPACC header "RIAD";
Read C_ROIAE from file BOPACC header "RIAE";
Read C_ROILD from file BOPACC header "RILD";
Read C_ROILE from file BOPACC header "RILE";

Read NETTRN from file BOPACC header "NTRN";



Formula
(initial) ADJDUMYEAR1 = 0.0 + if( DUM_YEAR1 lt 0.1, 1.0);
(initial) C_ROIAD = (1-DUM_YEAR1)*C_INTAD/C_A0D +DUM_YEAR1*C_ROIAD;
(initial) C_ROIAE = (1-DUM_YEAR1)*C_INTAE/C_A0E +DUM_YEAR1*C_ROIAE;
(initial) C_ROILD = (1-DUM_YEAR1)*C_INTLD/C_L0D +DUM_YEAR1*C_ROILD;
(initial) C_ROILE = (1-DUM_YEAR1)*C_INTLE/C_L0E +DUM_YEAR1*C_ROILE;

!INTRODUCE NATIONAL POOL HERE!

Formula
(all,d,DST) HOUS_DIS_INC(d) = GDPEXP(d)
+ sum{s,PSEC,TRANSS_H(d,s)
- NET_TAXTOTS(d,s) - OTHGOVREVS(s,d)}
+SHRASS_H(d)*[C_INTAD + C_INTAE - C_INTLD - C_INTLE]
;

(all,d,DST) V3TOT(d)= PUR_CS("Hou",d);

(all,d,DST) HOUS_SAV(d)     = HOUS_DIS_INC(d) - V3TOT(d);

(all,d,DST) AV_PROP_CON(d)  = V3TOT(d)/HOUS_DIS_INC(d);

Coefficient
HOUS_DISINCD;
HOUS_SAV_D;
GOV_SAV_D;
GOV_DEF_D;
TRANS_D;
NET_TAXTOTGD;
```

```
OTHGOVREVD;
OTHCAPGOV_D;
NAT_SAV;
NAT_PROP_CON;
V0GDPEXP;
V2TOT_G_I_D;
OTHCAPGOVD;
V3TOT_D;
V4TOT_D;
V0IMP_C_D;
VCAB;
GNP;


Formula

TRANS_D=sum(d,DST,sum{s,PSEC,TRANSS_G(d,s)});

NET_TAXTOTGD=sum(d,DST,sum{s,PSEC,NET_TAXTOTS(d,s)});

OTHGOVREVD=sum(d,DST,sum{s,PSEC,OTHGOVREVS(s,d)});
OTHCAPGOVD=sum(d,DST,sum{s,PSEC,OTHCAPGOVS(s,d)});

V2TOT_G_I_D=sum(d,DST,sum{s,PSEC,V2TOT_G_IS(d,s)});

GOV_DEF_D=sum(d,DST,sum(s,PSEC,GOV_DEFS(d,s)));

V0GDPEXP=sum(d,DST,GDPEXP(d));

V3TOT_D=sum(d,DST,V3TOT(d));

HOUS_SAV_D=sum(d,DST,HOUS_SAV(d));


(all,d,DST) AV_PROP_CON(d)  = V3TOT(d)/HOUS_DIS_INC(d);

!V4TOT_D=sum(c,COM,sum[s,SRC,sum{d,DST,PUR(c,s,"exp",d)}]);!

V4TOT_D=sum{d,REG,PUR_CS("exp",d)};


!V0IMP_C_D=sum(c,COM,sum[u,USR,sum{d,DST,PUR(c,"imp",u,d)}]);!
V0IMP_C_D=sum{d,REG,IMPLANDED_C(d)};
```

```
Formula
HOUS_DISINCD = V0GDPEXP + TRANS_D - NET_TAXTOTGD- OTHGOVREVD+C_INTAD +
C_INTAE-
C_INTLD - C_INTLE;
HOUS_SAV_D      = HOUS_DISINCD - V3TOT_D;
(initial) NAT_PROP_CON  = V3TOT_D/HOUS_DISINCD;
NAT_SAV       = HOUS_SAV_D - GOV_DEF_D + V2TOT_G_I_D + OTHCAPGOVD;
GOV_SAV_D       = NAT_SAV - HOUS_SAV_D;
NAT_SAV       = HOUS_SAV_D - GOV_DEF_D + V2TOT_G_I_D + OTHCAPGOVD;
VCAB = [V4TOT_D + C_INTAD +C_INTAE] - [V0IMP_C_D +  C_INTLD + C_INTLE ]+
NETTRN;

Coefficient
NAT_PROP_C_B;
(all,d,DST) AV_PROP_C_B(d);

Formula
(initial) NAT_PROP_C_B = NAT_PROP_CON;
(initial)(all,d,DST) AV_PROP_C_B(d) = AV_PROP_CON(d);

Formula
(initial)
C_FE =(1-DUM_YEAR1)*[C_A1E - C_A0E*(1+C_REVAE)]/NAT_SAV +DUM_YEAR1*C_FE;

(initial)
C_FD = (1-DUM_YEAR1)*[C_A1D - C_A0D*(1+C_REVAD)]/NAT_SAV +DUM_YEAR1*C_FD;
(initial)
C_REVLE = (1-DUM_YEAR1)*{(C_L1E - C_L0E
        - C_A1E + C_A0E*(1+C_REVAE)-C_A1D + C_A0D*(1+C_REVAD)
        + C_L1D - C_L0D*(1+C_REVLD)+ VCAB)/C_L0E }
        + DUM_YEAR1*C_REVLE;

Formula
(initial) C_A0E_B = C_A0E;
(initial) C_A1E_B = C_A1E;
(initial) C_A0D_B = C_A0D;
(initial) C_A1D_B = C_A1D;
(initial) C_L0E_B = C_L0E;
(initial) C_L1E_B = C_L1E;
(initial) C_L0D_B = C_L0D;
(initial) C_L1D_B = C_L1D;

!National saving!
```

```
Formula
GNP = V0GDPEXP + C_INTAD +C_INTAE - C_INTLD - C_INTLE + NETTRN;




!****************************************************************************!
!Section 5.17 Formulas used for labour markets!
!****************************************************************************!

Formula
            ALPHA1=1.1;
            ALPHA2=0.4;
            AGGLAB = sum(i,IND,sum{d,DST,LAB_O(i,d)});
            COL_PAYRTOT = sum(d,DST,sum{s,PSEC,COL_PAYRTOTS(d,s)});
(initial)   EMPLOY_B =EMPLOY;
(initial)   EMPLOY_O_B = EMPLOY_OLD;
(initial)   FEMPADJ_B = FEMPADJ;
(initial)   LAB_SUPN = sum(d,DST,LAB_SUP(d));
(initial)    LAB_SUPN_O = sum(d,DST,LAB_SUP(d));
            EMP_RATEN    = EMPLOY/LAB_SUPN;
            UN_RATEN =  [LAB_SUPN-EMPLOY]/LAB_SUPN;
 (initial)   RWAGE_B = RWAGE;
 (initial)   RWAGE_OLD_B = RWAGE_OLD;
             RWAGE_PT = RWAGE*(1 - TAX_L_RATE);
 (initial)   RWAGE_PT_B = RWAGE_PT;
             RWAGE_PT_OLD = RWAGE_OLD*(1 - TAX_L_RATE_O);
 (initial)   RWAGE_PT_O_B = RWAGE_PT_OLD;
 (initial)   RWAGE_PT_L_B    = RWAGE_L_B*(1 - TAX_L_RATE_L);

Formula(initial) RW_PT_O_L_B = RWAGE_O_L_B*(1 - TAX_L_R_O_L);
(initial) FEMPADJ_B = FEMPADJ;

!****************************************************************************!
!Section 5.18 Formulas used for Capital stocks and investment!
!****************************************************************************!

Zerodivide default 0.3333;

Formula
(initial) DIFF = 0.1;
```

```
(all,d,DST)(all,i,IND) V2TOT(d,i)=sum{c,COM,INVEST(c,i,d)};
(all,d,DST)(all,i,IND) QCAPATT(d,i) = VCAP_AT_T(d,i)/PCAP_AT_T(d,i);
(initial)(all,d,DST)(all,i,IND) QCAPATT_B(d,i) =  QCAPATT(d,i);
(all,r,DST)(all,i,IND) QINVEST(r,i) =
sum(d,DST,V2TOT(d,i))*QCAPATT(r,i)/(sum{d,DST,QCAPATT(d,i)})/PCAP_I(r,i);

(initial)(all,d,DST)(all,i,IND) QINV_BASE(d,i) = QINVEST(d,i);
(all,d,DST)(all,i,IND)
 QCAPATTPLUS1(d,i) = QCAPATT(d,i)*[1 - DEP(d,i)] + QINVEST(d,i);
(all,d,DST)(all,i,IND) K_GR(d,i) = [QCAPATTPLUS1(d,i)/QCAPATT(d,i)] - 1;
(initial)(all,d,DST)(all,i,IND) PCAP_AT_T_B(d,i) = PCAP_AT_T(d,i);
(initial)(all,d,DST)(all,i,IND) PCAP_I_B(d,i)    = PCAP_I(d,i);
(initial)(all,d,DST)(all,i,IND) PCAP_I_L_B(d,i)  = PCAP_I_L(d,i);
(initial)(all,d,DST)(all,i,IND) PCAP_AT_T1_B(d,i)= PCAP_AT_T1(d,i);
(initial)(all,d,DST)(all,i,IND) PCAP_AT_T_B(d,i) = PCAP_AT_T(d,i);
(initial)(all,d,DST)(all,i,IND) PCAP_I_L_B(d,i)  = PCAP_I_L(d,i);
(initial)(all,d,DST)(all,i,IND) PCAP_AT_T1_B(d,i)= PCAP_AT_T1(d,i);
(initial)(all,d,DST)(all,i,IND) PCAP_I_B(d,i)    = DUMMY_DEC*PCAP_I(d,i)
                                    + (1-DUMMY_DEC)*PCAP_I_B(d,i);

(all,d,DST)(all,i,IND)
V1CAP(d,i)=CAP(i,d);
V1CAP_I=sum[d,DST,sum{i,IND,V1CAP(d,i)}];
Zerodivide off;

Write K_GR    to file SUMMARY header "KGR";
      QINVEST to file SUMMARY header "QINV";



!Nominal prices and nominal and real interest rates!
Formula
(initial) ADJDUMYEAR1 = 0.0 + if( DUM_YEAR1 lt 0.1, 1.0);
(initial) LEV_CPI_B     = LEV_CPI;
(initial) LEV_CPI_L_B   = LEV_CPI_L;
(initial) LEV_PLAB_B    = LEV_PLAB;
(initial) LEV_PLAB_L_B = LEV_PLAB_L;

INF = LEV_CPI/LEV_CPI_L -1;
INTR = [1 + RINT]*[1 + INF] - 1;
RINT_PT_SE = [1+INTR*[1-TAX_K_RATE]]/[1+ INF] -1;
(all,d,DST)(all,i,IND)
VCAP_AT_TM(d,i) = VCAP_AT_T(d,i)*PCAP_I(d,i)/PCAP_AT_T(d,i);
(all,d,DST)(all,i,IND) ROR_SE(d,i) =
```

```
       (1/[1 + RINT_PT_SE])*{[V1CAP(d,i)*[1 - TAX_K_RATE]]/VCAP_AT_TM(d,i)
       + [1 - DEP(d,i)] + RALPH*TAX_K_RATE*DEP(d,i)} - 1;

(initial)(all,d,DST)(all,i,IND) ROR_SE_BASE(d,i) = ROR_SE(d,i);


!Coefficiencts in investment equations!

Formula
(initial)(all,d,DST)(all,i,IND) K_GR_MIN(d,i) = - DEP(d,i);
(initial)(all,d,DST)(all,i,IND) K_GR_MAX(d,i) = TREND_K(d,i) + DIFF
                                             + if(QCAPATT(d,i) <= 0.00001,
1.0);
(initial)(all,d,DST)(all,i,IND) COEFF_SL(d,i) =SMURF(d,i)*
 [K_GR_MAX(d,i)-K_GR_MIN(d,i)]/[[K_GR_MAX(d,i)-TREND_K(d,i)]*
 [TREND_K(d,i)-K_GR_MIN(d,i)]];
(all,d,DST)(all,i,IND) CHKGR1(d,i)= 0.0 + if(K_GR_MIN(d,i) >= K_GR(d,i),1.0);
(all,d,DST)(all,i,IND) CHKGR2(d,i)= 0.0 + if(K_GR_MAX(d,i) <= K_GR(d,i),1.0);
(all,d,DST)(all,i,IND) K_GR(d,i) = K_GR(d,i)
                + if[K_GR_MIN(d,i) >= K_GR(d,i), K_GR_MIN(d,i)-
K_GR(d,i)+0.005]
                + if[K_GR_MAX(d,i) <= K_GR(d,i),K_GR_MAX(d,i)-K_GR(d,i)-
0.005];

Formula
(initial) YEAR_B = YEAR;
(initial) ITER_NUM_B = ITER_NUM;
(initial) ONE_ITER1 =  if(ITER_NUM_B =1, 1);
(initial) ONE_IT1_REP =  if(ITER_NUM_B =NOFITERS+2, 1);
(initial) ZERO_PYR1 = 1 + if(ITER_NUM_B >= NOFITERS+2 and YEAR_B =YR_POLICY,-
1);
(initial) ITER_ADJUST = 0 + if(YEAR_B = NYEARS , 1);
(initial)(all,t,TIME) DUM_TIME(t) = 0 + if(YEAR_B = COEFF_TIME(t) ,1);
(initial)(all,t,TIME) DUM_TIME_LAG(t) = 0 + if(YEAR_B = COEFF_TIME(t) + 1
,1);
(initial) DUM_IT1 = 0 + if(ITER_NUM_B=1 or ITER_NUM_B = NOFITERS+2, 1);
(initial) COEFF_NYEAR  = 0 + if(YEAR_B < NYEARS,1);
INF_L = LEV_CPI_L/LEV_CPI_2L -1;
INTR_L = (1+INF_L)*(1+RINT_L) -1;
(initial) RINT_B  = RINT;
(initial) RINT_L_B  = RINT_L;
(initial) LEV_CPI_2L_B = LEV_CPI_2L;
```

```
(all,d,DST)(all,i,IND) ROR_ACT_L(d,i) =
        {1/[PCAP_I_L(d,i) *(1+INTR_L*(1-TAX_K_RATE))]}*
      [   (1-TAX_K_RATE)*(V1CAP(d,i)/QCAPATT(d,i))
      + (1-DEP(d,i))*PCAP_I(d,i) + RALPH*TAX_K_RATE*DEP(d,i)*PCAP_I(d,i)] -1;

(initial)(all,d,DST)(all,j,IND) ROR_ACT_L_B(d,j) =ROR_ACT_L(d,j);
(initial)(all,d,DST) (all,j,IND)(all,t,TIME)
    EROR_G_B(d,j,t) =EROR_G(d,j,t)
    + ZERO_PYR1*DUM_IT1*DUM_TIME_LAG(t)*(ROR_ACT_L_B(d,j)-EROR_G(d,j,t));
(initial)(all,d,DST)(all,j,IND)
    EROR_F(d,j) =  sum{t,TIME,DUM_TIME(t)*EROR_G_B(d,j,t)};
(initial)(all,d,DST)(all,j,IND)
    EROR_B(d,j) =  sum{t,TIME,DUM_TIME_LAG(t)*EROR_G_B(d,j,t)};
(all,i,IND)(all,d,DST) EEQROR(d,i)=
    RORN(d,i) +   F_EEQROR(d) + F_EEQROR_I(d,i) + (1/COEFF_SL(d,i))
   *{ [ Loge(K_GR(d,i)-K_GR_MIN(d,i)) - Loge(K_GR_MAX(d,i) - K_GR(d,i))]
  - [Loge(TREND_K(d,i)-K_GR_MIN(d,i)) - Loge(K_GR_MAX(d,i) - TREND_K(d,i))]};
(initial)(all,d,DST)(all,j,IND) DISEQRE_B(d,j) = EROR_B(d,j) - EEQROR(d,j);
(initial)(all,d,DST)(all,j,IND) DISEQSE_B(d,j) = ROR_SE_BASE(d,j) -
EEQROR(d,j);
(initial)(all,d,DST)(all,j,IND)(all,t,TIME)
    EROR_G(d,j,t) = EROR_G_B(d,j,t) +
     ZERO_PYR1*{if[DUM_TIME_LAG(t) ne 0,
    ADJ_RE(d,j)*(ROR_ACT_L_B(d,j)-EROR_G_B(d,j,t))]};


Formula
(initial)(all,d,DST)(all,i,IND) K_GR_MIN(d,i) = - DEP(d,i);
(initial)(all,d,DST)(all,i,IND) K_GR_MAX(d,i) =
    TREND_K(d,i) + DIFF + if(QCAPATT(d,i) <= 0.00001, 1.0);
(initial)(all,d,DST)(all,i,IND)
    COEFF_SL(d,i) =SMURF(d,i)*
    [K_GR_MAX(d,i)-K_GR_MIN(d,i)]/[[K_GR_MAX(d,i)-TREND_K(d,i)]*
    [TREND_K(d,i)-K_GR_MIN(d,i)]];
(all,i,IND)(all,d,DST) CHKGR1(d,i)= 0.0 + if(K_GR_MIN(d,i) >= K_GR(d,i),1.0);
(all,i,IND)(all,d,DST) CHKGR2(d,i)= 0.0 + if(K_GR_MAX(d,i) <= K_GR(d,i),1.0);
(all,i,IND)(all,d,DST) K_GR(d,i) =
        K_GR(d,i)
      + if[K_GR_MIN(d,i) >= K_GR(d,i), K_GR_MIN(d,i) - K_GR(d,i) + 0.005]
      + if[K_GR_MAX(d,i) <= K_GR(d,i),K_GR_MAX(d,i) - K_GR(d,i) - 0.005];

!***************************************************************************!
!Section 5.19 Aggregation of regional macro variables to natl macro variables
!
```

```
!**************************************************************************!

Formula
 (all,q,REG) WMAINMACRO("RealHou",q) = PUR_CS("Hou",q);
 (all,q,REG) WMAINMACRO("RealInv",q) = PUR_CS("Inv",q);
 (all,q,REG) WMAINMACRO("RealGov",q) = PUR_CS("Gov",q);
 (all,q,REG) WMAINMACRO("ExpVol",q) =  PUR_CS("Exp",q);
 (all,q,REG) WMAINMACRO("ImpVolUsed",q) = IMPUSED_C(q);
 (all,q,REG) WMAINMACRO("ImpsLanded",q) = IMPLANDED_C(q);
 (all,q,REG) WMAINMACRO("RealGDP",q) = GDPEXP(q);
 (all,q,REG) WMAINMACRO("AggEmploy",q) = LAB_IO(q);
 (all,q,REG) WMAINMACRO("realwage_io",q) = LAB_IO(q);
 (all,q,REG) WMAINMACRO("p1lab_io",q) = LAB_IO(q);
 (all,q,REG) WMAINMACRO("AggCapStock",q) = CAP_I(q);
 (all,q,REG) WMAINMACRO("CPI",q) = PUR_CS("Hou",q);
 (all,q,REG) WMAINMACRO("GDPPI",q) = GDPEXP(q);
 (all,q,REG) WMAINMACRO("ExportPI",q) = PUR_CS("Exp",q);
 (all,q,REG) WMAINMACRO("ImpsLandedPI",q) = IMPLANDED_C(q);
 (all,q,REG) WMAINMACRO("Population",q) = POP(q);
 (all,q,REG) WMAINMACRO("NomHou",q) = PUR_CS("Hou",q);
 (all,q,REG) WMAINMACRO("NomGDP",q) = GDPEXP(q);
 (all,m,MAINMACROS) WNATMACRO(m) = sum{q,REG, WMAINMACRO(m,q)};
 (initial)(all,m,MAINMACROS) RATIOMMACRO(m) = 1;


!##########################################################################!
!*********************************************************************!
! Section 6: Variable declarations in alphabetical order
!
!*********************************************************************!
!##########################################################################!


Variable
                a0e  # Foreign assets, equity, start of year #;
                a1e  # Foreign assets, equity, end of year #;
                a0d  # Foreign assets, debt, start of year #;
                a1d # Foreign assets, debt, end of year #;
(all,c,COM)(all,i,IND)(all,d,DST)
    a1q(c,i,d)#Shifter for industry structure of c#;
(all,i,IND)(all,d,DST)
    a1qsum(i,d)#Shifter for industry structure of c#;
(all,i,IND)(all,d,DST) a1cap(i,d)   # Capital-augmenting technical change #;
(all,d,DST)(all,s,PSEC) age_bens(d,s) #Age benefits#;
```

```
(all,c,COM)(all,d,DST)(all,h,HOU)
   ahou_s(c,d,h) # Taste change,household imp/dom compsite#;
(all,c,COM)(all,i,IND)(all,d,DST) a1int_s(c,i,d) # Intermediate tech change
#;
(all,i,IND)(all,o,OCC)(all,d,DST) a1lab(i,o,d) #Labour productivity#;
(all,o,OCC)             a1lab_id(o) #Occupation specific labour productivity#;
(all,i,IND)(all,d,DST) a1lab_o(i,d) # Labor-augmenting technical change #;
(all,i,IND) a1lab_od(i) # Labor-augmenting technical change #;
(all,i,IND)(all,d,DST) a1lnd(i,d)   # Land-augmenting technical change #;
(all,i,IND)(all,d,DST) a1primsum(i,d);

(all,c,COM)(all,d,DST)(all,h,HOU)
   alux(c,d,h) # Taste change, supernumerary demands #;
(all,i,IND)(all,d,DST) a1prim(i,d)  # Primary-factor-augmenting tech change
#;
(all,c,COM)(all,d,DST)(all,h,HOU)
   asub(c,d,h) # Taste change, subsistence demands #;
(all,m,MAR)(all,r,ORG)(all,d,DST)(all,p,PRD) asuppmar(m,r,d,p)
  # Tech change, Margin m supplied by p on goods passing from r to d #;
(all,i,IND)(all,d,DST) a1tot(i,d) # All-input-augmenting technical change #;
(all,c,COM)(all,s,SRC)(all,m,MAR)(all,r,ORG)(all,d,DST)
   atradmar(c,s,m,r,d) # Tech change: margin m on good c,s going from r to d
#;
(all,m,MAR)(all,r,ORG)(all,d,DST)
   atradmar_cs(m,r,d) # Tech change: margin m on goods going from r to d #;
(All,c,COM)(all,d,DST) aveqsum(c,d) #All output augmenting tech change#;
(all,i,IND)  bint_scd(i) # Driver: intermediate tech change #;
(all,c,COM)(all,i,IND)(all,d,DST) bint_s(c,i,d) # Intermediate tech change #;
(all,i,IND)(all,d,DST) caplab(i,d);
(all,d,DST)(all,j,IND) ch_kgr1(d,j) # CHKGR1:checks if K_GR_MIN(j) >= K_GR(j)
#;
(all,d,DST)(all,j,IND) ch_kgr2(d,j) # CHKGR2:checks if K_GR_MAX(j) <= K_GR(j)
#;
Variable (change) (all,c,COM)(all,d,DST)(all,h,HOU) contCPI(c,d,h)
 # Contributions by commodity to % regional CPI #;
Variable (change) (all,i,GDPEXPCAT)(all,d,REG) contxgdpexp(i,d)
 # Contributions to % regional real GDP expenditure #;
Variable (change) (all,i,NATGDPEXPCAT) contxnatgdpexp(i)
 # Contributions to % regional real GDP expenditure #;
(change)(all,i,IND)(all,d,DST)
  contnatxtot(i,d) # Regional contributions to national industry output #;
 (change)(all,i,IND)(all,d,DST)
  contxprim_i(i,d) # Sector contributions to regional GDP at factor cost #;
(Change)        d_cab;
```

```
(change)        d_empadj # Determines speed of direct employment adjustment#;
(change)        d_emp_sh # Set at one to zero out shift in E_d_f_empadj #;
(change)        del_f_wage_c # Shift in labour supply, pre-tax #;
(change)        del_f_wage_pt # Shifter in post-tax stick-wage equation #;
(change)        d_ff_empadj # Exogenized to zero out shift in E_d_f_empadj #;
(change)        d_f_empadj # Exogenized to cause direct adjustment of agg
empl#;

(change)(all,d,DST)(all,i,IND)        d_diseq(d,i)
    # Disequilibrium in expected ror, SE #;
(change)        d_dum_year1 # One in year one, zero in later years #;
(change)(all,d,DST)(all,i,IND)   d_eeqror(d,i)
    # Equilibrium expected ror, SE #;
(change)        d_inf_l    # Inflation rate, lagged #;
(change)        d_int_l    # Nominal rate of interest, lagged #;
(change)(all,d,DST)(all,j,IND)    del_ror_se_o(d,j)
    # ror for industry j in forecast, SE #;
(change)(all,d,DST)(all,j,IND) d_f_ror_se_o(d,j)
    # Shift in eqn. that records SE. rate of return #;
(change)        d_f_p1lab_io_l    # Shifter in lagged wage equation #;
(change)        d_f_p3tot_l   # Shifter in lagged CPI equation #;
(change)        d_f_p3tot_2l
    # Turns off 2 lag inflation equ, if init. soln is not from t-1 #;
(change)        d_f_rint_l     # Shifter in real rate of interest, lagged #;
(change)(all,d,DST)(all,i,IND)  d_eror(d,i)
    # Percentage point changes in expected ror #;
(change)(all,d,DST)    d_eror_ave(d) # Average expected rate of return #;
(change)(all,d,DST)(all,i,IND)   d_eror_o(d,i)
    # Expected ror in forecast #;
(change)(all,d,DST)(all,i,IND)    d_f_diseq(d,i)
    # Shifter in d_diseq under SE #;
(change)(all,d,DST)(all,j,IND)   d_f_diseqre(d,j)
    # Shifter in d_diseq under RE #;
(change)(all,d,DST)      d_f_eeqror(d)
    # General capital growth shifter, in yr-to-yr #;
(change)(all,d,DST)(all,i,IND)                d_f_eeqror_i(d,i)
    # Industry-specific cap. growth shifter, yr-to-yr, RE#;
(change)(all,d,DST)(all,j,IND)           d_f_eror_o(d,j)
    # Shift in equation that records forecast of expected ror #;
(change)(all,d,DST)(all,i,IND)                 d_f(d,i)
    # Exog. in all iters with rational expect., endo for static exp #;
(change)(all,d,DST)(all,i,IND)      d_ff(d,i)
    # Exog. simulations with SE, endog for RE #;
(change)(all,d,DST)(all,i,IND)
```

```
   d_f_p2tot_l(d,i)
    # Turns off lag cap. price equ, if initial soln is not from t-1 #;
(change)(all,d,DST)(all,i,IND)
   d_f_ac_p_y(d,i)
   # Shifter in cap. accum. eq'n for year t-1 in yr-to-yr f'cast #;
(change)(all,d,DST)(all,i,IND)
   d_f_pcapatt(d,i)
   # Shifter for pcapatt; Endogenous if initial solution is not from t-1 #;
(change)(all,d,DST)(all,i,IND)
   d_ff_pcapatt1(d,i)
   # Shifter for pcapatt1; Endogenous if initial solution is not from t-1 #;
(change)        d_f_trn;
(change)        d_govsav_nat;
(change)(all,d,DST)(all,j,IND)
   d_k_gr(d,j)  # Capital growth thru forecast year #;
(change)        d_nettrn;
(change)        d_newle # New foreign equity liabilities #;
(change)        d_newld # New foreign debt liabilities #;
(change)        d_revae # Revaluation factor for foreign equity assets #;
(change)        d_revad # Revaluation factor for foreign debt assets #;
(change)        d_revle # Revaluation factor for foreign equity liabilities
#;
(change)        d_revld # Revaluation factor for foreign debt liabilities #;
(change)        d_rint_l   # Real rate of interest, lagged #;
(change)        d_rint_pt_se # Real post-tax interest rate, SE #;
(change)(all,d,DST)(all,i,IND)
                d_ror_act_l(d,i) # Lagged actual rate of return #;
(change)(all,d,DST)(all,j,IND)
                d_ror_se(d,j) # Percentage point changes in ror: static
expect#;
(change)(all,d,DST)(all,s,PSEC) d_col_payrs(d,s) #Collection of payroll taxes
#;
(change)(all,d,DST)(all,s,PSEC) d_gov_defs(d,s) # public sector deficits #;
(change)(all,s,PSEC)d_gov_def(s) # public sector deficits #;
(all,d,DST) d_f_govtomun(d) #Shifter for transfers from govt to muns#;
(all,d,DST) d_f_govtossf(d) #Shifter for transfers from govt to soc sec
funds#;
(all,d,DST) d_f_muntogov(d) #Shifter for transfers from muns to govt#;
(all,d,DST) d_f_ssftogov(d) #Shifter for soc sec funds to govt#;
(change)(all,d,DST)(all,s,PSEC) d_f_othcapgovs(d,s) # shift oth pub cap
expd#;
(change)(all,d,DST)(all,s,PSEC) d_f_psd_ts(d,s);
(change)(all,d,DST)(all,s,PSEC) d_f_psd_t1s(d,s);
(change)        d_f_rint_psd;
```

```
(all,d,DST)(all,c,COM)(all,s,PSEC)  d_f5totr(d,c,s);
(all,d,DST)(all,s,PSEC)             d_f5totrc(d,s);
(all,c,COM)                         d_f5totrds(c);
(all,c,COM)(all,d,DST)              d_f5totrs(d,c);
(all,s,PSEC)                        d_f5totrcd(s);
(all,d,DST)(all,c,COM)(all,s,PSEC)  d_f5reg(d,c,s);
(change) d_gov_def_nat;
(change) d_othcapgov_nat;
(change)       d_inf   # Nominal rate of interest #;
(change)       d_int   # Nominal rate of interest #;
(change)       d_rint   # Nominal rate of interest #;
(change)       d_int_psd   # Nominal rate of interest #;
(change)       d_rint_psd   # real rate of interest #;
(change)(all,d,DST)(all,s,PSEC) d_net_int_gs(d,s);
(Change)(all,s,PSEC)            d_net_int_gd(s);
(Change)(all,d,DST)(all,s,PSEC) d_nettaxtot(d,s);
(Change)(all,p,PSEC) d_nettaxtot_d(p);
(Change) (all,d,DST)(all,s,PSEC) d_othgovrev(d,s);
(Change) (all,s,PSEC) d_othgovrev_d(s);
(change)(all,d,DST)(all,s,PSEC) d_psd_t1s(d,s);
(change)(all,d,DST)(all,s,PSEC) d_psd_ts(d,s);
(Change)(all,s,PSEC) d_psd_ts_d(s);
(Change)(all,s,PSEC) d_psd_t1s_d(s);
(Change) d_psd_ts_nat;
(change)(all,d,DST)(all,s,PSEC) d_transfs(d,s) # public sector transfers #;
(change)(all,d,DST)(all,s,PSEC) d_transfs_h(d,s) ;
(change)(all,d,DST)(all,s,PSEC) d_othcapgovs(d,s) # oth pub sec cap expd#;
(change)       d_unity # Homothopy variable#;
(Change) (all,d,DST)(all,s,PSEC)    d_w5tots(d,s);
(Change) (all,s,PSEC)               d_w5tots_d(s);
(Change)                            d_w5tots_ds;
(change) delB;
(change) (all,i,GDPEXPCAT)(all,d,DST)
   delPGDPEXP(d,i)# Ordinary change in price expenditure GDP component #;
(change) (all,i,GDPEXPCAT)(all,d,DST)
   delXGDPEXP(d,i)# Ordinary change in quantity expenditure GDP component #;
(change) (all,i,GDPINCCAT)(all,d,DST)
   delGDPINC(d,i)# Ordinary change in nominal income GDP component #;
(change) (all,i,NATGDPEXPCAT)
   delNatXGDPEXP(i)# Ordinary change in quantity expenditure GDP component #;
(change) (all,i,NATGDPEXPCAT)
   delNATPGDPEXP(i)# Ordinary change in price expenditure GDP component #;
(change)(all,i,IND)(all,d,DST)
   delPRIM(i,d)# Ordinary change in cost of primary factors #;
```

```
(change)(all,i,IND)(all,d,DST)
 delPTX(i,d) # Ordinary change in production tax revenue #;
 (change)(all,i,IND)(all,d,DST)
   delPTXRATE(i,d) # Change in rate of production tax #;
(change)(all,c,COM)(all,s,SRC)(all,d,DST)(all,i,IND)
 delTAXint(c,s,i,d)# Ordinary change in intermediate commodity tax revenue #;
(change)(all,c,COM)(all,s,SRC)(all,d,DST)
 delTAXhou(c,s,d)# Ordinary change in household commodity tax revenue #;
(change)(all,c,COM)(all,s,SRC)(all,d,DST)
 delTAXinv(c,s,d)# Ordinary change in investment commodity tax revenue #;
(change)(all,c,COM)(all,s,SRC)(all,d,DST)
 delTAXgov(c,s,d)# Ordinary change in government commodity tax revenue #;
(change)(all,c,COM)(all,s,SRC)(all,d,DST)
 delTAXexp(c,s,d)# Ordinary change in export commodity tax revenue #;
(change)(all,d,DST)(all,s,PSEC) delV0tax_csis(d,s);
(change)(all,d,DST)             delV0tax_csi(d);
(change)(all,d,DST) d_UR(d) #Ordinary change in unemployment rate#;
(change) d_URN #Ordinary change in unemployment rate, national#;
    employ_i   # Aggregate employment: wage bill weights #;
    employ_i_o # Aggregate employment in forecast simulation #;
(all,i,IND) employed(i)# Aggregate employment in forecast simulation #;
    f_aprim_id #General primary factor productivity shifter#;
(all,i,IND) f_atot_d(i) #Intermediate improving tech change#;
(all,i,IND)(all,d,DST) f_atot(i,d) #Intermediate improving tech change#;
(all,c,COM)(all,i,IND)(all,d,DST) f_a1q(c,i,d)#Shifter for ind structure of
c#;
(all,i,IND) f_a1q_dc(i) #Shifter for ind structure of c#;
    f_age_ben_nat;
(all,d,DST)(all,s,PSEC) f_age_bens(d,s) #shifter for agebens#;
(all,i,IND)(all,d,DST) f_aprim(i,d) #Primary factor augmenting tech change#;
(all,i,IND) f_aprim_d(i) #Industry specific prim fact tech change #;
(all,d,DST) f_aprim_i(d)#Region specific tech change#;
    f_emp_o # Shift for the value of emp_hours_o #;
    f_labsup_o;
    f_rwage_o # Shift for the value of real_wage_c_o #;
    f_rwage_pt_o # Shift for the value of real_wage_pt_o #;
(all,i,IND)(all,o,OCC)(all,d,DST) f_x1lab(i,o,d);
(all,i,IND)(all,o,OCC)(all,d,DST) ff_x1lab(i,o,d);
    ftax_l_rds_o;
(all,d,DST) f_govtomun(d);
(all,d,DST) f_govtossf(d);
    f_grants_nat;
(all,d,DST)(all,s,PSEC) f_grants_s(d,s);
(all,d,DST) f_labsup(d)#Labour supply shifter#;
```

```
   f_labsupn #Labour supply shifter, national#;
(all,d,DST) f_muntogov(d);
(all,d,DST)(all,s,PSEC) f_oth_bens(d,s);
                        f_oth_ben_nat;
(all,d,DST)(all,s,PSEC) f_othcapgovs(d,s);
(all,d,DST)(all,s,PSEC) f_othcapgov(d,s);
(all,d,DST)(all,s,PSEC) f_oth_grevs(d,s) #shifter for oth pub sec revenue#;
(all,d,DST) f_ssftogov(d);
(all,c,com) f_pimp_d(c)#foreign currency import price shifter#;
            f_pimp_dc;
(all,d,DST)(all,s,PSEC) f_tax_k_r(d,s);
(all,d,DST)(all,s,PSEC) f_tax_ab_r(d,s);
(all,d,DST)(all,s,PSEC) f_tax_l_r(d,s);
(all,s,PSEC) f_tax_l_rd(s);
(all,d,DST)(all,s,PSEC) f_tax_ob_r(d,s);
(all,d,DST)(all,s,PSEC) f_tax_ub_r(d,s);
(all,d,DST)(all,s,PSEC) f_unempben_rats(d,s);
(all,d,DST)(all,s,PSEC) f_unemp_bens(d,s) #shifter for unempbens#;
(all,c,COM)(all,d,DST) f_x0com(c,d) #Shifter for turning off sum-up of
x0com#;
(all,i,IND) f_x2tot_d(i);
    f_x2tot_id;
                fd # Shr of national savings devoted to foreign debt assets#;
                fe # Shr of national savings devoted to foreign equity
assets#;
(all,c,COM)(all,s,SRC)(all,d,DST) fgov(c,s,d) # Government demand shifter #;
(all,c,COM)(all,d,DST)            fgov_s(c,d) # Government demand shifter #;
(all,d,DST)                        fgovtot(d) # Government demand shifter #;
     fgovgen   # Economy-wide govt demand shift#;
(all,i,IND)(all,o,OCC)(all,d,DST) flab(i,o,d) # Wage shifter #;
(all,o,OCC) flab_id(o) # Wage shifter #;
(all,d,DST) flab_io(d) # Wage shifter #;
(all,i,IND) (all,d,DST) flab_o(i,d) #Wage shifter#;
(all,i,IND) flab_od(i) #Industry soecific wage shifter#;
    flab_iod #Wage shifter, national#;
(all,c,COM)(all,s,SRC) fqexp(c,s)  # Export quantity shift variable #;
(all,c,COM)(all,s,SRC) fpexp(c,s)  # Export price shift variable #;
(all,c,COM)            f_pexp_d(c);
                       f_pexp_dc;
(all,c,COM)(all,d,DST) f_qexp(c,d);
(all,d,DST)            f_qexp_c(d);
(all,c,COM)            f_qexp_d(c);
(all,c,COM)            ff_qexp_d(c);
(all,i,IND)            ff_qexp_i(i);
```

```
                      f_qexp_dc;
(all,i,IND)(all,d,DST) f_stocks(i,d);
(all,i,IND)            f_stocks_d(i);
(all,d,DST)            f_f3tot(d);
                       f_f3tot_d;
(all,d,DST) f3tot(d);
            f3tot_d;
(all,d,DST)(all,c,COM)(all,s,PSEC)f5totr(d,c,s);
               gnpnom # Nominal gnp #;
(all,d,DST)(all,s,PSEC) grants_s(d,s);
(all,d,DST) hdispinc(d);
            hdispinc_d;
(all,d,DST) housav(d);
            housav_d # Household savings #;
            intad # %-change in Interest on foreign assets, debt #;
            intae # %- change in Interest foreign assets, equity #;
            intld # %- change in Interest foreign liabilities, debt #;
            intle # %- change in Interest foreign liabilities, equity #;
            l0e  # Foreign liabilities, equity, start of year #;
            l1e  # Foreign liabilities, equity, end of year #;
            l0d  # Foreign liabilities, debt, start of year #;
            l1d # Foreign liabilities, debt, end of year #;
(all,d,DST)             labsup(d);
  labsupn;
  labsup_o # Labour supply in forecast simulation #;
(change)(all,d,DST)(all,i,IND)
   lev_eror(d,i) # Levels of expected rors in year t #;
(change)(all,d,DST)(all,i,IND)
   lev_eror_l(d,i)
   # Lagged levels of expect. rors, usually expect. rors for t-1 #;
(change)(all,d,DST)(all,i,IND)
   lev_ror_act_l(d,i) # Level of actual ror in year t-1 #;
(all,m,MAINMACROS)(all,q,REG) MainMacro(m,q) # Convenient macros for
reporting#;
(all,m,MAINMACROS) NatMacro(m) # National macros for reporting #;
(change)(all,m,MAINMACROS)(all,q,REG)
   contMainMacro(m,q) # Regional contributions to national macro results #;
(all,i,IND) natemploy(i) # National employment average #;
(all,h,HOU) natp3tot(h) # National CPI by household #;
(all,i,IND) natptot(i) # National output price average #;
            natsav # National savings #;
(all,c,COM) natxcom(c) # National commodity outputs #;
(all,d,DST)(all,s,PSEC) nettaxtots(d,s) # net tax revenue#;
(all,h,HOU) natx3tot(h) # National real consumption by household #;
```

```
(all,c,COM) natximp(c) # National imports #;
(all,i,IND) natxtot(i) # National industry output -- value added weights #;
(all,i,IND) natwprim(i) # National nominal factor payments #;
(all,d,DST) nhou(d) # Number of households #;
(all,d,DST)(all,h,HOU)           nhouh(d,h) # Number of households #;
(all,d,DST)(all,s,PSEC) oth_bens(d,s)# oth benefits#;
(all,d,DST)(all,s,PSEC) oth_govrevs(d,s) # oth pub sector revenue#;
                        p0gdpexp;
                        p0imp_c;
                        p0realdev;
                        p0toft;
   p3tot;
   p1lab_io_l   # Lagged nominal wage #;
(all,d,DST)(all,i,IND)
   p2tot(d,i)        # Cost of unit of capital #;
(all,d,DST)(all,i,IND)
   p2tot_l(d,i)        # Cost of unit of capital, lagged #;
   p3tot_l   # Consumer price index lagged #;
   p3tot_2l  # Consumer price index lagged #;
(all,i,IND) p2tot_d(i);
(all,d,DST) p2tot_i(d);
            p4tot;
(all,d,DST)(all,s,PSEC) p5tots(d,s);
(all,c,COM)(all,s,SRC)(all,r,ORG) pbasic(c,s,r) # Basic prices #;
(all,d,DST)(all,i,IND)
   pcapatt(d,i) #Asset price of capital by ind, start of year #;
(all,d,DST)(all,i,IND)
   pcapatt1(d,i) # Asset price of capital by ind, end of year #;
(all,i,IND)(all,d,DST) p1cap(i,d) # Rental price of capital #;
(all,i,IND)(all,d,DST)  pcst(i,d)   # Ex-tax cost of production #;
(all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
         pdelivrd(c,s,r,d) # All-user delivered price of good c,s from r to d
#;
(all,c,COM)(all,r,ORG)
   pdom(c,r) # Output prices = basic prices of domestic goods #;
(all,c,COM)(all,r,ORG) pfimp(c,r) # Import prices, foreign currency #;
(all,u,FINDEM)(all,d,DST) pfin(u,d) # Final user price indices #;
(all,d,DST) pgdpexp(d) # Price index expenditure GDP #;
  phi        # Exchange rate, local currency/$world #;
(all,d,DST)(all,h,HOU)          p3tot_c(d,h) # CPI #;
(all,c,COM)(all,d,DST) p3(c,d) # Household price of composites #;
(all,c,COM)(all,r,ORG) pimp(c,r)  # Import prices, local currency #;
(all,d,DST) pimpused(d) # Price index, imports used in d #;
(all,r,ORG) pimplanded(r) # Price index, imports landed in d #;
```

```
(all,i,IND)(all,d,DST) pint(i,d) # Intermediate effective price indices #;
(all,c,COM)(all,d,DST)
   pinvest(c,d) # Purchaser's price of good c for investment in d #;
(all,i,IND)(all,d,DST) pinvitot(i,d) # Investment price index by industry #;
(all,i,IND)(all,o,OCC)(all,d,DST) p1lab(i,o,d) # Wage rates #;
(all,o,OCC)(all,d,DST) p1lab_i(o,d)     # Average wage by occ and reg #;
(all,o,OCC) p1lab_id(o)     # Average wage by occ and reg #;
(all,d,DST) p1lab_io(d) # Ave wage by region #;
(all,i,IND)(all,d,DST) p1lab_o(i,d) # Price of labour composite #;
(all,i,IND)(all,d,DST) plnd(i,d) # Rental price of land #;
(all,c,COM)(all,i,IND)(all,d,REG)
   pmake(c,i,d) # Price received by industries #;
   pnatgdpexp # Price index expenditure national GDP #;
   popagednat # Population over 65#;
   popnat # Population, national#;
(all,d,DST)              pops(d);
(all,d,DST)              popaged(d);
(all,d,DST)(all,i,IND)  powpayroll(d,i);
(all,d,DST)(all,i,IND)(all,s,PSEC) powpayrolls(d,i,s)#pow of payroll taxes#;
(all,i,IND)(all,d,DST)
 p1prim(i,d)  # Effective price of primary factor composite #;
(all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST)
   ppur(c,s,u,d) # User (purchasers) prices, inc margins and taxes #;
(all,c,COM)(all,u,USR)(all,d,DST) ppur_s(c,u,d) # User prices, average over
s#;
(all,c,COM)(all,s,SRC)(all,d,DST)
    puse(c,s,d) # Delivered price of regional composite good c,s to d #;
(all,i,IND)(all,d,DST) ptot(i,d) # Industry output prices #;
(all,i,IND)(all,d,DST)
    pvar(i,d)   # Shortrun variable cost of production #;
    ratio_led # Ratio, fgn equity liab. to fgn debt liab.,endofyr #;
(all,d,DST)(all,i,IND)
    r_inv_cap_i(d,i) # Investment capital ratio by  industry #;
    r_inv_cap_u # Uniform shifter in investment capital ratio #;
(all,d,DST) r_inv_cap(d);
(all,i,IND) r_inv_cap_d(i);
(all,i,IND)(all,o,OCC)(all,d,DST) realwage(i,o,d) # Wages deflated by CPI #;
  real_wage_c # Real wage for consumers #;
  real_wage_c_o # Pre-Tax real wage in forecast simulation #;
(all,o,OCC)(all,d,DST) realwage_i(o,d) # Average real wages #;
(all,o,OCC) realwage_id(o) # Average real wages #;
(all,d,DST) realwage_io(d) # Average real wage by region #;
  realwage_iod;
  real_wage_pt # Economy-wide CPI-deflated wage rate, post tax #;
```

```
  real_wage_pt_o # Post-Tax real wage in forecast simulation #;
(all,o,OCC)(all,d,DST) rlab_i(o,d)     # Total real wage bill #;
(all,o,OCC) rlab_id(o)     # Total real wage bill #;
(all,d,DST) rlab_io(d) # Total real wage bill by region #;
              roiad # %- change in Rate of interest fgn assets, debt #;
              roiae # %- change in Rate of interest fgn assets, debt #;
              roild # %- change in Rate of interest fgn liabilities, debt
#;
              roile # %- change in Rate of interest fgn liabilities, equity
#;
(all,i,IND)            s2gov(i);
(all,d,DST)(all,s,PSEC) tax_ab_r(d,s);
(all,d,DST)(all,s,PSEC) tax_k_r(d,s);
    tax_k_rr;
(all,d,DST)(all,s,PSEC) tax_l_r(d,s);
    tax_l_rds;
    tax_l_rds_o;
 (all,d,DST)(all,s,PSEC) tax_ob_r(d,s);
(all,d,DST)(all,s,PSEC) tax_ub_r(d,s);
(all,d,DST)(all,s,PSEC) taxrev_incs(d,s) #Value of income tax revenue #;

(all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST)
   tuser(c,s,u,d) # Powers of commodity taxes #;
(all,c,COM)(all,u,USR) tuser_sd(c,u);
(all,c,COM)(all,s,SRC) tuser_ud(c,s) # Tax shifter by commodity #;
(all,i,COM) (all,s,SRC)(all,k,ORG) twistsrc(i,s,k)
   # Sourcing twist towards origin (k) for good (i,s) #;
(all,i,IND) twistlk(i)# Twist towards labour away from capital # ;
(all,d,DST)(all,s,PSEC) unemp_bens(d,s);
(all,d,DST)(all,s,PSEC) unempben_rats(d,s);
    x0gdpexp;
    x0imp_c;
    x1cap_id    # Aggregate capital stock, rental weights #;
(all,d,DST)(all,i,IND)
    x1cap_tplus1(d,i) # Capital stock at t+1 (end of forecast year) #;
(all,d,DST)(all,i,IND)
   x2tot(i,d)        # Investment by using industry #;
(all,i,IND) x2tot_d(i);
(all,d,DST) x2tot_i(d);
    x4tot;
(all,c,COM)(all,s,SRC)(all,d,DST)(all,z,PSEC) x5totr(c,s,d,z);
(all,d,DST)(all,s,PSEC) x5tots(d,s);
(all,i,IND)(all,d,DST) x1cap(i,d) # Capital usage #;
(all,d,DST) x1cap_i(d)  # Aggregate capital, rental-weighted #;
```

```
(all,c,COM)(all,d,DST) xcom(c,d) # Total output of commodities #;
(all,c,COM)(all,r,REG) xdomexp(c,r)
  # Amount good c made in r sent to other domestic regions (non-margin) #;
(all,c,COM)(all,d,REG) xdomimp(c,d)
# Amount domestic good c used in d made in other domestic regions (non-
margin)#;
(all,r,REG) xdomexp_c(r)
  # Amount goods made in r sent to other domestic regions (non-margin) #;
(all,d,REG) xdomimp_c(d)
# Amount domestic goods used in d made in other domestic regions (non-
margin)#;
(all,c,COM)(all,r,REG) xdomloc(c,r) # Amount good c made in r and used in r
#;
(all,c,COM)(all,s,SRC)(all,d,DST)
   xexp(c,s,d) # Export of all-region composite leaving port at d #;
(all,c,COM)(all,d,DST) xexp_s(c,d) # Export demands, dom+imp #;
(all,u,FINDEM)(all,d,DST) xfin(u,d) # Final user quantity indices #;
(all,d,DST) xgdpexp(d) # Real expenditure GDP #;
(all,c,COM)(all,s,SRC)(all,d,DST)
   xgov(c,s,d) # Government demands for all-region composite #;
(all,c,COM)(all,d,DST) xgov_s(c,d) # Government demands, dom+imp #;
(all,c,COM)(all,s,SRC)(all,d,DST)
   x3(c,s,d) # Household demands for all-region composite #;
(all,c,COM)(all,d,DST)(all,h,HOU) x3_s(c,d,h) # Household demands #;
(all,c,COM)(all,d,DST) x3_sh(c,d) # Household demands for dom/imp composite
#;
(all,d,DST)(all,h,HOU) x3tot(d,h) # Total real household consumption #;
(all,d,DST) xhoutot(d) # Total real household consumption #;
(all,c,COM)(all,d,DST) ximps(c,d) # Volume of imports used in d #;
(all,d,DST) ximpused(d) # Volume of imports used in d #;
(all,r,ORG) ximplanded(r) # Volume of imports landed in d #;
(all,c,COM)(all,s,SRC)(all,d,DST)
   x2(c,s,d) # Investment demands for all-region composite #;
(all,c,COM)(all,i,IND)(all,d,DST)
   x2i(c,i,d) # Amount of good c for investment, industry i in d #;
(all,c,COM)(all,d,DST) x2_s(c,d) # Investment demands for dom/imp composite
#;
(all,c,COM)(all,s,SRC)(all,i,IND)(all,d,DST)
   x1(c,s,i,d) # Intermediate demands for all-region composite #;
(all,c,COM)(all,s,SRC)(all,d,DST)
   x1_i(c,s,d) # Total intermediate demand for regional composite c,s in d #;
 (all,c,COM)(all,i,IND)(all,d,DST)
   x1_s(c,i,d) # Industry demands for dom/imp composite #;
(all,i,IND)(all,o,OCC)(all,d,DST) x1lab(i,o,d) # Labour demands #;
```

```
(all,o,OCC)(all,d,DST) x1lab_i(o,d)     # Aggregate labour, wage-weighted #;
(all,o,OCC) x1lab_id(o)     # Aggregate labour, wage-weighted #;
(all,d,DST) x1lab_io(d) # Labour by region, wage-weighted #;
(all,i,IND)(all,d,DST) x1lab_o(i,d) # Effective labour input #;
(all,i,IND)(all,d,DST) x1lnd(i,d) # Land usage #;
(all,d,DST) x1lnd_i(d)  # Aggregate land, rental-weighted #;
(all,c,COM)(all,s,SRC)(all,d,DST)
   xlocuse(c,s,d) # Total non-export demand for regional composite c,s in d#;
(all,c,COM)(all,d,DST)
   xlocuse_s(c,d) # Total non-export demand for good c in d #;
(all,c,COM)
   xlocuse_sd(c) # National non-export demand for good c #;
 (all,c,COM)(all,d,DST)(all,h,HOU)
   xlux(c,d,h) # Household - supernumerary demands #;
(all,c,COM)(all,i,IND)(all,d,REG)
   xmake(c,i,d) # Output of good c by industry i in d #;
(all,i,NATGDPEXPCAT)
   xNatXGDPEXP(i) # Percent change in quantity expenditure GDP component #;
   xnatgdpexp # Real expenditure GDP #;
(all,i,IND)(all,d,DST) x1prim(i,d)  # Primary factor composite #;
(all,d,DST) xprim_i(d)   # Regional GDP at factor cost (% change) #;
(all,c,COM)(all,r,ORG)(all,d,DST) xrowdem(c,r,d) # Eventually exported goods
#;
(all,c,COM)(all,r,ORG) xrowdem_d(c,r) # Eventually exported goods made in r
#;
(all,i,IND)(all,d,DST) xstocks(i,d) # Inventories #;
(all,c,COM)(all,d,DST)(all,h,HOU)
   xsub(c,d,h) # Household - subsistence demands #;
(all,m,MAR)(all,r,ORG)(all,p,PRD)
   xsuppmar_d(m,r,p) # Total margins on goods from r, produced in p #;
(all,m,MAR)(all,r,ORG)(all,d,DST)(all,p,PRD)
   xsuppmar(m,r,d,p) # Demand for margin m (made in p) on goods from r to d
#;
(all,m,MAR)(all,r,ORG)(all,d,DST)
   psuppmar_p(m,r,d) # Price of composite margin m on goods from r to d #;
(all,m,MAR)(all,r,ORG)(all,d,DST)
   xsuppmar_p(m,r,d) # Quantity of composite margin m on goods from r to d #;
(all,m,MAR)(all,p,PRD)
   xsuppmar_rd(m,p) # Total demand for margins produced in p #;
(all,i,IND)(all,d,DST) x1tot(i,d) # Industry outputs #;
(all,c,COM)(all,s,SRC)(all,d,DST)
   xtrad_r(c,s,d) # Total demand for regional composite c,s in d #;
(all,c,COM)(all,s,SRC)(all,m,MAR)(all,r,ORG)(all,d,DST)
   xtradmar(c,s,m,r,d) # Margin m on good c,s going from r to d #;
```

```
(all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
   xtrad(c,s,r,d) # Quantity of good c,s from r to d #;
(all,c,COM)(all,s,SRC)(all,r,ORG) xtrad_d(c,s,r)
  # Total direct demands for goods produced(dom) or landed(imp) in r #;
(all,c,COM)(all,s,SRC) xtrad_rd(c,s)  # National direct use of goods #;
(all,d,DST)(all,i,GDPEXPCAT)
     xXGDPEXP(d,i)# Percent change in quantity expenditure GDP component #;
     w0gdpexp;
     w0imp_c;
(all,d,DST)(all,s,PSEC) w0tax_csis(d,s) #value of indirect tax collection#;
(all,d,DST)(all,s,PSEC) w2totg_is(d,s);
                        w2totg_is_ds;
                        w3tot_d;
(all,d,DST)(all,s,PSEC) w5tots(d,s);
(all,d,DST) wcap_i(d)  # Total rentals to capital #;
(all,u,FINDEM)(all,d,DST) wfin(u,d) # Final user expenditures #;
(all,d,DST) wgdpexp(d) # Nominal expenditure GDP #;
(all,d,DST) wgdpdiff(d) # SHOULD=0: nominal (income - expend) GDP #;
(all,d,DST) wgdpinc(d) # Nominal income GDP #;
(all,d,DST)(all,h,HOU) w3tot(d,h) # Total nominal household consumption #;
    w4tot;
(all,o,OCC)(all,d,DST) wlab_i(o,d)     # Wage bill #;
(all,o,OCC) wlab_id(o)     # Total wage bill by occupation#;
(all,d,DST) wlab_io(d) # Total wage bill by region #;
(all,i,IND)(all,d,DST) wlab_o(i,d) # Wage bills by occupation#;
   wlabnat#Wage bill at national level#;
(all,d,DST) wlnd_i(d)  # Total rentals to land #;
(all,d,DST)(all,h,HOU)
   wlux(d,h) # Total nominal supernumerary household expenditure #;
   wnatgdpexp # Nominal expenditure GDP #;
   wnatgdpinc # Nominal income GDP #;
   wnatgdpdiff  # SHOULD=0: nominal (income - expend) GDP #;
(all,d,DST) wprim_i(d) # Total factor payments #;
(all,i,IND)(all,d,DST) wprim(i,d)  # Primary factor payments #;


!##########################################################################!
!**************************************************************************!
! Section 7: Updates in alphabetical order                                 !
!**************************************************************************!
!##########################################################################!

Update
```

```
(all,d,DST)(all,s,PSEC)  AGEBENS(s,d)=age_bens(d,s);
(all,i,IND)(all,d,DST) CAP(i,d) = p1cap(i,d)*x1cap(i,d);
                    C_A0E = a0e;
                    C_A1E = a1e;
                    C_A0D = a0d;
                    C_A1D = a1d;
                    C_L0E = l0e;
                    C_L1E = l1e;
                    C_L0D = l0d;
                    C_L1D = l1d;
                    C_INTAD =intad;
                    C_INTAE =intae;
                    C_INTLD =intld;
                    C_INTLE =intle;
                    C_ROIAD =roiad;
                    C_ROIAE =roiae;
                    C_ROILD =roild;
                    C_ROILE =roile;
(change)            C_REVAE =d_revae;
(change)            C_REVAD =d_revad;
(change)            C_REVLD =d_revld;
(change)            C_REVLE =d_revle;
                    C_FE =fe;
                    C_FD =fd;

    EMPLOY = employ_i;
    EMPLOY_OLD = employ_i_o;
(all,d,DST) EMPLOYS(d)=x1lab_io(d);

(change)(all,c,COM)(all,d,DST)(all,h,HOU)
    EPSH(c,d,h) = EPSH(c,d,h)
    *[xlux(c,d,h)- x3_s(c,d,h)+w3tot(d,h)-wlux(d,h)]/100.0;
!
multiple households only:
(all,c,COM)(all,d,DST)(all,h,HOU) HOUPUR(c,d,h) = phou(c,d)*xhouh_s(c,d,h);
!

(change)            FEMPADJ = d_f_empadj;
(change)(all,d,DST) F_EEQROR(d) = d_f_eeqror(d);
(change)(all,d,DST)(all,i,IND)
    F_EEQROR_I(d,i) = d_f_eeqror_i(d,i);

(change)(all,d,DST)(all,h,HOU)
    FRISCHH(d,h) = FRISCHH(d,h)*[w3tot(d,h)-wlux(d,h)]/100.0;
```

```
(all,d,DST) GOVTOMUN(d)=f_govtomun(d);
(all,d,DST) GOVTOSSF(d)=f_govtossf(d);
(all,d,DST)(all,s,PSEC)            GRANTSS(s,d) = grants_s(d,s);
(all,c,COM)(all,i,IND)(all,d,DST)
    INVEST(c,i,d) = pinvest(c,d)*x2i(c,i,d);
(all,i,IND)(all,o,OCC)(all,d,DST) LAB(i,o,d) = p1lab(i,o,d)*x1lab(i,o,d);
(all,d,DST) LAB_SUP(d)=labsup(d);
    LAB_SUPN = labsupn;
    LAB_SUPN_O = labsup_o;
    LEV_CPI = p3tot;
    LEV_CPI_L = p3tot_l;
    LEV_CPI_2L = p3tot_2l;
(all,i,IND)(all,d,DST) LND(i,d) = plnd(i,d)*x1lnd(i,d);
(all,c,COM)(all,i,IND)(all,d,REG)
    MAKE(c,i,d) = xmake(c,i,d)*pmake(c,i,d);
(all,d,DST) MUNTOGOV(d)=f_muntogov(d);
                         NAT_PROP_CON=f_f3tot_d;
Update (change)     NETTRN = d_nettrn;
(all,d,DST)(all,s,PSEC)  OTHBENS(s,d)=oth_bens(d,s);
(all,d,DST)(all,s,PSEC)            OTHGOVREVS(s,d) = oth_govrevs(d,s);
(change) (all,d,DST)(all,s,PSEC)   OTHCAPGOVS_(s,d) = d_othcapgovs(d,s);
(change) (all,d,DST)(all,s,PSEC)   OTHCAPGOVS(s,d) = d_othcapgovs(d,s);
(all,d,DST)(all,i,IND)         PCAP_AT_T(d,i) = pcapatt(d,i);
(all,d,DST)(all,i,IND)         PCAP_I(d,i) = p2tot(d,i);
(all,d,DST)(all,i,IND)         PCAP_I_L(d,i) = p2tot_l(d,i);
(all,d,DST)(all,i,IND)         PCAP_AT_T1(d,i) = pcapatt1(d,i);
(all,d,DST)(all,i,IND)             POW_PAYROLL(i,d) = powpayroll(d,i);
(all,d,DST)(all,i,IND)(all,s,PSEC) POW_PAYROLLS(s,i,d) = powpayrolls(d,i,s);
(change)(all,i,IND)(all,d,DST) PRODTAX(i,d) = delPTX(i,d);
(change)(all,d,DST)(all,s,PSEC)    PSDATT(s,d) = d_psd_ts(d,s);
(all,d,DST)(all,h,HOU) RATIOCPI(d,h) = p3tot_c(d,h);
(all,d,DST) RATIOPRIM_I(d)= xprim_i(d);
(all,m,MAINMACROS) RATIOMMACRO(m) = NatMacro(m);
    RATIONGDPEXP = xNATgdpexp;
(all,i,IND) RATIOPRIM_D(i) =  natxtot(i);
(change)            RINT_L = d_rint_l;
(change)            RINT = d_rint;
(change)            INT_PSD = d_int_psd;
    RWAGE = real_wage_c;
    RWAGE_OLD = real_wage_c_o;
(all,d,DST) SSFTOGOV(d)=f_ssftogov(d);
(all,i,IND)(all,d,DST) STOCKS(i,d) = ptot(i,d)*xstocks(i,d);
(change)(all,c,COM)(all,s,SRC)(all,i,IND)(all,d,DST)
   TAX(c,s,i,d) = delTAXint(c,s,i,d);
```

```
(change)(all,c,COM)(all,s,SRC)(all,d,DST)
   TAX(c,s,"hou",d) = delTAXhou(c,s,d);
(change)(all,c,COM)(all,s,SRC)(all,d,DST)
   TAX(c,s,"inv",d) = delTAXinv(c,s,d);
(change)(all,c,COM)(all,s,SRC)(all,d,DST)
   TAX(c,s,"gov",d) = delTAXgov(c,s,d);
(change)(all,c,COM)(all,s,SRC)(all,d,DST)
   TAX(c,s,"exp",d) = delTAXexp(c,s,d);
(all,d,DST)(all,s,PSEC) TAXS_AB_RATE(s,d)=tax_ab_r(d,s);
(all,d,DST)(all,s,PSEC) TAXS_K_RATE(s,d)=tax_k_r(d,s);
(all,d,DST)(all,s,PSEC) TAXS_L_RATE(s,d)=tax_l_r(d,s);
   TAX_L_RATE = tax_l_rds;
   TAX_L_RATE_O = tax_l_rds_o;
(all,d,DST)(all,s,PSEC) TAXS_OB_RATE(s,d)=tax_ob_r(d,s);
(all,d,DST)(all,s,PSEC) TAXS_UB_RATE(s,d)=tax_ub_r(d,s);
(all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
   TRADE(c,s,r,d) = pbasic(c,s,r)*xtrad(c,s,r,d);
(all,m,MAR)(all,r,ORG)(all,d,DST)(all,p,PRD)
   SUPPMAR(m,r,d,p) = xsuppmar(m,r,d,p)*pdom(m,p);
(all,c,COM)(all,s,SRC)(all,m,MAR)(all,r,ORG)(all,d,DST)
   TRADMAR(c,s,m,r,d) =  xtradmar(c,s,m,r,d)*psuppmar_p(m,r,d);
(all,d,DST)(all,s,PSEC) UNEMPBENS(s,d)=unemp_bens(d,s);
(all,c,COM)(all,s,SRC)(all,i,IND)(all,d,DST)
   USE(c,s,i,d) = x1(c,s,i,d)*puse(c,s,d);
(all,c,COM)(all,s,SRC)(all,d,DST)
   USE(c,s,"hou",d) = x3(c,s,d)*puse(c,s,d);
(all,c,COM)(all,s,SRC)(all,d,DST)
   USE(c,s,"inv",d) = x2(c,s,d)*puse(c,s,d);
(all,c,COM)(all,s,SRC)(all,d,DST)
   USE(c,s,"gov",d) = xgov(c,s,d)*puse(c,s,d);
(all,c,COM)(all,s,SRC)(all,d,DST)
   USE(c,s,"exp",d) = xexp(c,s,d)*puse(c,s,d);
(all,d,DST)(all,i,IND)
   VCAP_AT_T(d,i) = x1cap(i,d)*pcapatt(d,i);
(change)           YEAR = d_unity;
```

```
!##########################################################################!
!**************************************************************************!
! Section 8: Equations in thematic order
!
!**************************************************************************!
!##########################################################################!
!**************************************************************************!
! Section 8.1. Demand for goods and primary factors
!
!**************************************************************************!
!**************************************************************************!
! Section 8.1.1 Intermediate demands
!
!**************************************************************************!

set
FUEL # Energy commodities #
    read elements from file INFILE header "fuel";  ! f !

Subset FUEL is subset of COM;
SET COMLFUEL # non-energy commodities # = COM - FUEL;

Coefficient
(all,i,IND)     SIGMAFUELS(i) # CES substitution btw. energy composites #;
(all,i,IND)     SIGMAGREEN(i) # CES subst.el. btw. primary factors and energy
#;


Read sigmagreen from file INFILE header "SIGR";
Read sigmafuels from file INFILE header "SIFU";




Equation
 E_x1 #Dom/imp substitution in intermediate demands#
(all,c,COM)(all,s,SRC)(all,i,IND)(all,d,DST)
x1(c,s,i,d) = x1_s(c,i,d) - SIGMADOMIMP(c)*[ppur(c,s,i,d)-ppur_s(c,i,d)];

Equation
 E_a1int_s #Intermediate-augmenting technical change#
(all,c,COM)(all,i,IND)(all,d,DST)
   a1int_s(c,i,d) = bint_scd(i) + bint_s(c,i,d);
```

```
Equation E_x1_i # Total intermediate demand for regional composite c,s in d #
(all,c,COM)(all,s,SRC)(all,d,DST)
  ID01[USE_I(c,s,d)]*x1_i(c,s,d) = sum{i,IND, USE(c,s,i,d)*x1(c,s,i,d)};

 E_pint #Price index for composite intermediates #  (all,i,IND)(all,d,DST)
   ID01(PUR_CS(i,d))*pint(i,d)
  = sum{c,COM,PUR_S(c,i,d)*[ppur_s(c,i,d)+a1int_s(c,i,d)]};


![[!
!***************************************************************************
!
! Section 8.1.2. Primary factor demand
!
!***************************************************************************
!

formula TINY=0.00000001;

Equation E_a1primsum  # Primary factor saving tech change at the K-L-Land
nest #
(all,i,IND)(all,d,DST)
[PRIM(i,d)+TINY]*a1primsum(i,d) =
                [LAB_O(i,d)+TINY]*[a1lab_o(i,d)+a1lab_od(i)]
              + [CAP(i,d)+TINY]*[a1cap(i,d)]
              + [LND(i,d)+TINY]*[a1lnd(i,d)];

! Occupational composition of labour demand !



Equation E_x1lab   # Demand for labour by industry and skill group #
  (all,i,IND)(all,o,OCC)(all,d,DST)
   x1lab(i,o,d) = x1lab_o(i,d) - SIGMa1lab(i)*[p1lab(i,o,d) - p1lab_o(i,d)]
    +f_x1lab(i,o,d);


Equation E_ff_x1lab   # Demand for labour by industry and skill group #
  (all,i,IND)(all,o,OCC)(all,d,DST)
   x1lab(i,o,d) = x1lab_o(i,d)+ ff_x1lab(i,o,d);

Equation E_p1lab_o # Price to each industry of labour composite #
```

```
  (all,i,IND)(all,d,DST)
 ID01(LAB_O(i,d))*p1lab_o(i,d) = sum{o,OCC, LAB(i,o,d)*p1lab(i,o,d)};

Equation E_wlab_o # Wage bills #
  (all,i,IND)(all,d,DST)
ID01(LAB_O(i,d))*wlab_o(i,d)=
sum{o,OCC,LAB(i,o,d)*[p1lab(i,o,d)+x1lab(i,o,d)]};

Equation E_x1lab_o  # Industry demands for effective labour #
  (all,i,IND)(all,d,DST) x1lab_o(i,d) - a1lab_o(i,d)-a1lab_od(i) =
xprim(i,d)-SIGMAPRIM(i)*[p1lab_o(i,d) + a1lab_o(i,d)+a1lab_od(i) -
pprim(i,d)]
-SIGMAPRIM(i)*(a1lab_o(i,d)+a1lab_od(i)-a1primsum(i,d))
            +SHRLK(i,d,"capital")*twistlk(i);

Equation E_a1lab #Labour productivity change#
 (all,i,IND)(all,o,OCC)(all,d,DST) a1lab(i,o,d)=a1lab_id(o);

Equation E_p1cap  # Industry demands for capital #
  (all,i,IND)(all,d,DST)  x1cap(i,d) - a1cap(i,d) =
   xprim(i,d) - SIGMAPRIM(i)*[p1cap(i,d) + a1cap(i,d) - pprim(i,d)]
-SIGMAPRIM(i)*(a1cap(i,d)-a1primsum(i,d))
            -SHRLK(i,d,"labour")*twistlk(i);


Equation E_caplab  # Equation for cap lab ratio #
  (all,i,IND)(all,d,DST)  x1cap(i,d) - x1lab_o(i,d) = caplab(i,d);


Equation E_plnd  # Industry demands for land #
  (all,i,IND)(all,d,DST)  x1lnd(i,d) - a1lnd(i,d) =
   xprim(i,d) - SIGMAPRIM(i)*[plnd(i,d) + a1lnd(i,d) - pprim(i,d)]
-SIGMAPRIM(i)*(a1lnd(i,d)-a1primsum(i,d));


Equation E_pprim  # Effective price term for factor demand equations #
  (all,i,IND)(all,d,DST)
ID01(PRIM(i,d))*pprim(i,d) =
 LAB_O(i,d)*[p1lab_o(i,d) + a1lab_o(i,d)+a1lab_od(i)]
  + CAP(i,d)*[p1cap(i,d) + a1cap(i,d)] + LND(i,d)*[plnd(i,d) + a1lnd(i,d)];

Equation E_xprim  # Use of composite primary factor #
  (all,i,IND)(all,d,DST) xprim(i,d) = xtot(i,d)+atot(i,d)+aprim(i,d);
```

```
Equation E_f_atot (all,i,IND)(all,d,DST)

atot(i,d)=f_atot(i,d)+f_atot_d(i);

Equation E_aprim #Total primary factor productivity#
(all,i,IND)(all,d,DST)
aprim(i,d)=f_aprim(i,d)+f_aprim_d(i)+f_aprim_i(d)+f_aprim_id;


Equation E_wprim  # Primary factor payments #
  (all,i,IND)(all,d,DST)
ID01[PRIM(i,d)]*wprim(i,d) = CAP(i,d) *[p1cap(i,d) + x1cap(i,d)]
                              +  LND(i,d) *[plnd(i,d) + x1lnd(i,d)]
                    + sum{o,OCC, LAB(i,o,d)* [p1lab(i,o,d) + x1lab(i,o,d)]};

Equation E_delPRIM  # Ordinary change in total cost of primary factors #
  (all,i,IND)(all,d,DST)
100*delPRIM(i,d) = PRIM(i,d)*wprim(i,d);

!]]!

!***************************************************************************!
! Section 8.1.2. Primary factor demand
!
!***************************************************************************!

formula TINY=0.00000001;


Equation E_a1primsum  # Primary factor saving tech change at the K-L-Land
nest #
(all,i,IND)(all,d,DST)
[PRIM(i,d)+TINY]*a1primsum(i,d) =
                [LAB_O(i,d)+TINY]*[a1lab_o(i,d)+a1lab_od(i)]
              + [CAP(i,d)+TINY]*[a1cap(i,d)]
              + [LND(i,d)+TINY]*[a1lnd(i,d)];

! Occupational composition of labour demand !


Equation E_x1lab   # Demand for labour by industry and skill group #
  (all,i,IND)(all,o,OCC)(all,d,DST)
   x1lab(i,o,d) = x1lab_o(i,d) - SIGMa1lab(i)*[p1lab(i,o,d) - p1lab_o(i,d)]
```

```
    +f_x1lab(i,o,d);


Equation E_ff_x1lab   # Demand for labour by industry and skill group #
  (all,i,IND)(all,o,OCC)(all,d,DST)
   x1lab(i,o,d) = x1lab_o(i,d)+ ff_x1lab(i,o,d);

Equation E_p1lab_o # Price to each industry of labour composite #
  (all,i,IND)(all,d,DST)
 ID01(LAB_O(i,d))*p1lab_o(i,d) = sum{o,OCC, LAB(i,o,d)*p1lab(i,o,d)};

Equation E_wlab_o # Wage bills #
  (all,i,IND)(all,d,DST)
ID01(LAB_O(i,d))*wlab_o(i,d)=
sum{o,OCC,LAB(i,o,d)*[p1lab(i,o,d)+x1lab(i,o,d)]};

Equation E_x1lab_o  # Industry demands for effective labour #
  (all,i,IND)(all,d,DST) x1lab_o(i,d) - a1lab_o(i,d)-a1lab_od(i) =
x1prim(i,d)-SIGMAPRIM(i)*[p1lab_o(i,d) + a1lab_o(i,d)+a1lab_od(i) -
p1prim(i,d)]
-SIGMAPRIM(i)*(a1lab_o(i,d)+a1lab_od(i)-a1primsum(i,d))
            +SHRLK(i,d,"capital")*twistlk(i);

Equation E_a1lab #Labour productivity change#
 (all,i,IND)(all,o,OCC)(all,d,DST) a1lab(i,o,d)=a1lab_id(o);

Equation E_p1cap  # Industry demands for capital #
  (all,i,IND)(all,d,DST)  x1cap(i,d) - a1cap(i,d) =
   x1prim(i,d) - SIGMAPRIM(i)*[p1cap(i,d) + a1cap(i,d) - p1prim(i,d)]
-SIGMAPRIM(i)*(a1cap(i,d)-a1primsum(i,d))
            -SHRLK(i,d,"labour")*twistlk(i);


Equation E_caplab  # Equation for cap lab ratio #
  (all,i,IND)(all,d,DST)  x1cap(i,d) - x1lab_o(i,d) = caplab(i,d);


Equation E_plnd  # Industry demands for land #
  (all,i,IND)(all,d,DST)  x1lnd(i,d) - a1lnd(i,d) =
   x1prim(i,d) - SIGMAPRIM(i)*[plnd(i,d) + a1lnd(i,d) - p1prim(i,d)]
-SIGMAPRIM(i)*(a1lnd(i,d)-a1primsum(i,d));


Equation E_pprim  # Effective price term for factor demand equations #
```

```
  (all,i,IND)(all,d,DST)
ID01(PRIM(i,d))*p1prim(i,d) =
 LAB_O(i,d)*[p1lab_o(i,d) + a1lab_o(i,d)+a1lab_od(i)]
  + CAP(i,d)*[p1cap(i,d) + a1cap(i,d)] + LND(i,d)*[plnd(i,d) + a1lnd(i,d)];


Equation E_f_atot (all,i,IND)(all,d,DST)

a1tot(i,d)=f_atot(i,d)+f_atot_d(i);

Equation E_aprim #Total primary factor productivity#
(all,i,IND)(all,d,DST)
a1prim(i,d)=f_aprim(i,d)+f_aprim_d(i)+f_aprim_i(d)+f_aprim_id;


Equation E_wprim  # Primary factor payments #
  (all,i,IND)(all,d,DST)
ID01[PRIM(i,d)]*wprim(i,d) = CAP(i,d) *[p1cap(i,d) + x1cap(i,d)]
                             +  LND(i,d) *[plnd(i,d) + x1lnd(i,d)]
                   + sum{o,OCC, LAB(i,o,d)* [p1lab(i,o,d) + x1lab(i,o,d)]};

Equation E_delPRIM  # Ordinary change in total cost of primary factors #
  (all,i,IND)(all,d,DST)
100*delPRIM(i,d) = PRIM(i,d)*wprim(i,d);

!****************************************************************************!
!subsection 8.1.3. demands for energy carriers!
!****************************************************************************!

!$  Problem: for each industry i, minimize cost of energy           !
!$          sum{c,FUEL, P1_S(c,i)*X1_S(c,i)}                        !
!$  such that $  X1FUEL(i) = CES(All,c,FUEL: X1_S(c,i)/A1_s(c,i) )  !

Coefficient
(all,i,IND)(all,d,DST)   V1FUEL(i,d) # Aggregate use of alternative fuels #;
(all,i,IND)(all,d,DST)   V1PRIM_F(i,d)# Total primary factor and energy
inputs#;

Formula
(all,i,IND)(all,d,DST) V1FUEL(i,d)= sum{f,FUEL,sum{s,SRC,PUR(f,s,i,d)}};
(all,i,IND)(all,d,DST) V1PRIM_F(i,d)= PRIM(i,d)+V1FUEL(i,d);

Variable
(all,i,IND)(all,d,DST) x1fuel(i,d);
```

```
(all,i,IND)(all,d,DST) p1fuel(i,d);
(all,i,IND)(all,d,DST) a1_sfuel(i,d);
(all,i,IND)(all,d,DST) a1prim_f(i,d);


Equation E_x1_s_f  # Demands for fuels composites #
 (All,c,FUEL)(all,i,IND)(all,d,DST)
x1_s(c,i,d) - [a1int_s(c,i,d)] = x1fuel(i,d)
                              -  SIGMAFUELS(i)*{ppur_s(c,i,d)-p1fuel(i,d)}
                              -  SIGMAFUELS(i)*{a1int_s(c,i,d)-
a1_sfuel(i,d)};

 E_p1fuel # Price of fuel composite #
(all,i,IND)(all,d,DST) [V1FUEL(i,d)+TINY]*p1fuel(i,d)
 = sum{c,FUEL,sum(s,SRC, PUR(c,s,i,d))*ppur_s(c,i,d)};

E_a1_sfuel # Fuel saving tech change at the fuel nest #
(all,i,IND)(all,d,DST) [V1FUEL(i,d)+TINY]*a1_sfuel(i,d)
 = sum{c,FUEL,sum{s,SRC, PUR(c,s,i,d)}*a1int_s(c,i,d)};

!**********************************************************************!
!subsection 8.1.4. demand for primary-factor and energy composites!
!**********************************************************************!

!$  Problem: for each industry i, minimize cost of primary factors and energy
!
!$         P1PRIM(i)*X1PRIM(i)+P1FUEL(i)*X1FUEL(i) such that
!
!$  X1PRIM_F(i) = CES(X1PRIM(i)/A1PRIM(i),X1FUEL(i)/A1_FUEL(i))
!

Variable
(all,i,IND)(all,d,DST) x1prim_f(i,d);
(all,i,IND)(all,d,DST) p1prim_f(i,d);
(all,i,IND)(all,d,DST) a1prim_fsum(i,d);
 (all,i,IND)(all,d,DST) a1_fuel(i,d);

Equation E_x1prim  # Demands for primary factor composite #
 (all,i,IND)(all,d,DST)  x1prim(i,d) - [a1prim(i,d) ] = x1prim_f(i,d)
                -  SIGMAGREEN(i)*(p1prim(i,d)-p1prim_f(i,d))
                -  SIGMAGREEN(i)*(a1prim(i,d)-a1prim_fsum(i,d));

Equation E_x1fuel  # Demands for energy composite #
 (all,i,IND)(all,d,DST)  x1fuel(i,d) - [a1_fuel(i,d) ] = x1prim_f(i,d)
```

```
                           -SIGMAGREEN(i)*{p1fuel(i,d)-p1prim_f(i,d)}
                           -SIGMAGREEN(i)*{a1_fuel(i,d)-a1prim_fsum(i,d)};
 E_p1prim_f # Price of energy-primary composite #
  (all,i,IND)(all,d,DST)
  V1PRIM_F(i,d)*p1prim_f(i,d) =
PRIM(i,d)*p1prim(i,d)+V1FUEL(i,d)*p1fuel(i,d);

E_a1prim_fsum # Energy-primary-saving tech change at energy primary nest #
 (all,i,IND)(all,d,DST)
   V1PRIM_F(i,d)*a1prim_fsum(i,d)
                 = PRIM(i,d)*a1prim(i,d)+V1FUEL(i,d)*a1_fuel(i,d);

!****************************************************************************!
! section 8.1.5. Demand for intermediate, primary-energy
and other cost composites!
!****************************************************************************!


!$  Problem: for each industry i, minimize cost
!
!$  sum(c,COM,P1_s(c,i)*x1_s(c,i))+P1PRIM_F(i)*X1PRIM_F(i)+P1OCT(i)*X1OCT(i)
!
!$  such that
!
!   X1TOT(i) = MIN( All,c,COM: X1_S(c,i)/[A1_S(c,s,i)*A1TOT(i)],
!
!$                             X1PRIM_F(i)/[A1PRIM_F(i)*A1TOT(i)],
!
!$                             X1OCT(i)/[A1OCT(i)*A1TOT(i)] )
!

Equation E_x1prim_f
(All,i,IND)(all,d,DST) x1prim_f(i,d) -a1prim_f(i,d) -a1tot(i,d) = x1tot(i,d);


 E_x1_s #Demand for composite intermediate goods#
(all,c,COMLFUEL)(all,i,IND)(all,d,DST)
   x1_s(c,i,d) = a1tot(i,d) + a1int_s(c,i,d) + x1tot(i,d)
            -0.15*{ppur_s(c,i,d) + a1int_s(c,i,d) - pint(i,d)};


! Output prices !

Equation E_pvar #Variable cost#
```

```
(all,i,IND)(all,d,DST)
ID01(VARCST(i,d))*[pvar(i,d)-a1tot(i,d)] =
 LAB_O(i,d)*[p1lab_o(i,d) + a1lab_o(i,d)+a1lab_od(i)] +
PUR_CS(i,d)*pint(i,d);

Equation E_pcst #Total cost of production excluding taxes#
(all,i,IND)(all,d,DST)
ID01(VCST(i,d))*[pcst(i,d)-a1tot(i,d)] =
 PRIM(i,d)*[a1prim(i,d)+p1prim(i,d)] +  PUR_CS(i,d)*pint(i,d);

Equation E_delPTX #Change in production taxes#
(all,i,IND)(all,d,DST)
delPTX(i,d) =
  0.01*PRODTAX(i,d)*[x1tot(i,d)+pcst(i,d)]    +    VCST(i,d)*delPTXRATE(i,d);

Equation E_ptot #Total cost# (all,i,IND)(all,d,DST)
 ID01(VTOT(i,d))*[ptot(i,d) + x1tot(i,d)] =
 VCST(i,d)*[pcst(i,d) + x1tot(i,d)] + 100*delPTX(i,d);


!zero profit!
Equation E_xtotA # Average price received by multi-product industries #
  (all,i, MIND)(all,d,REG)
    ptot(i,d) = sum{c,MINDCOM, MAKESHR1(c,i,d)*pmake(c,i,d)};

Equation E_xtotB # Price received by single-product industries #
  (all,i,SIND)(all,d,REG) ptot(i,d) = pdom(SIND2COM(i),d);


! Next equation says that
 users of good c do not regard c made different industries as perfect
 substitutes. Rather they CES (sigma=1/0.05) between the different industry
 products. This would be needed, for example, if 2 industries produced only
 electricity, with all inputs available elastically. The division of output
 between the two industries would not be well defined. Replace 0.05 by 0 to
 restore perfect substitute idea. !

Equation E_pmake  # Demands for commodities from industries #
 (all,c,COM)(all,i,IND)(all,d,REG)
!   xmake(c,i,d) = xcom(c,d) - 20*[pout(c,i,d) - pdom(c,d)];         !
!    pmake(c,i,d) = pdom(c,d) - 0.05*[xmake(c,i,d) - xcom(c,d)];     !
    pmake(c,i,d) = pdom(c,d) - 0.5*[xmake(c,i,d) - xcom(c,d)];
```

```
!***************************************************************************!
! Section 8.1.6. Commodity mix of output
!
!***************************************************************************!


Equation
 E_xmake  # Supplies of commodities by industries #
 (all,c,COM)(all,i,IND)(all,d,REG)
  xmake(c,i,d)+a1q(c,i,d) = x1tot(i,d) + SIGMAOUT(i)*[pmake(c,i,d)-
ptot(i,d)]
             + SIGMAOUT(i)*[ -a1q(c,i,d) + a1qsum(i,d)];

Equation E_a1qsum #All output-augmenting technical change#
(All,i,IND)(all,d,DST)
 Sum(c,COM, MAKE(c,i,d))*a1qsum(i,d) = Sum(c,COM, MAKE(c,i,d)*a1q(c,i,d));

Equation E_aveqsum #Average output-augmenting technical change#
(All,c,COM)(all,d,DST)
 Sum(i,IND, MAKE(c,i,d))*aveqsum(c,d) = Sum(i,IND, MAKE(c,i,d)*a1q(c,i,d));

Equation E_f_a1q #Fixes industry structure of producion of c #
  (all,c,COM)(all,i,IND)(all,d,REG)
xmake(c,i,d)=xcom(c,d)+f_a1q(c,i,d)+f_a1q_dc(i);


Equation
E_xcom # Total output of commodities made by several industries #
(all,c,COM)(all,d,REG)
xcom(c,d) = sum{i,IND, MAKESHR2(c,i,d)*xmake(c,i,d)}+f_x0com(c,d);




!***************************************************************************!
! Section 8.2.   Investment demands                                          !
!***************************************************************************!


Equation E_x2 # Dom/imp substitution # (all,c,COM)(all,s,SRC)(all,d,DST)
    x2(c,s,d) = x2_s(c,d) - SIGMADOMIMP(c)*[ppur(c,s,"inv",d)-pinvest(c,d)];
```

```
Equation E_x2i # Leontief technology for new capital creation #
  (all,c,COM)(all,i,IND)(all,d,DST) x2i(c,i,d) = x2tot(i,d);

Equation E_pinvest # Alias # (all,c,COM)(all,d,DST)
    pinvest(c,d) = ppur_s(c,"Inv",d);

Equation E_pinvitot # Price index of investment goods #
(all,i,IND)(all,d,DST)
    ID01(INVEST_C(i,d))*pinvitot(i,d) =
sum{c,COM,INVEST(c,i,d)*pinvest(c,d)};

Equation E_x2_s # Add up industry demands for investment goods #
  (all,c,COM)(all,d,DST)
    ID01(INVEST_I(c,d))*x2_s(c,d)= sum{i,IND,INVEST(c,i,d)*x2i(c,i,d)};



!***************************************************************************!
! Section 8.3. Household demands                                           !
! follows LES/Stone-Geary/Klein-Rubin scheme                               !
!***************************************************************************!

! Multiple households version
Set HOU # Households # read elements from file REGSETS header "HOU";
Coefficient (all,c,COM)(all,d,DST)(all,h,HOU) HOUPUR(c,d,h) #Household
demands#;
Read HOUPUR from file INFILE header "3PUR";  !
! end single households section !
! scale EPS so they average to 1: else they tend to drift off in recursive
sims!

! Dom/imp substitution !

Equation E_x3 # Dom/imp substitution in consumption#
(all,c,COM)(all,s,SRC)(all,d,DST)
x3(c,s,d) = x3_sh(c,d) - SIGMADOMIMP(c)*[ppur(c,s,"hou",d)-p3(c,d)];

Equation E_p3 # Alias for consumer prices #
  (all,c,COM)(all,d,DST) p3(c,d) = ppur_s(c,"hou",d);

Equation E_xsub # Subsistence demand for composite commodities #
  (all,c,COM)(all,d,DST)(all,h,HOU) xsub(c,d,h) = nhouh(d,h) + asub(c,d,h);

Equation E_xlux # Luxury demand for composite commodities #
```

```
  (all,c,COM)(all,d,DST)(all,h,HOU)
   xlux(c,d,h) + p3(c,d) = wlux(d,h) + alux(c,d,h);

Equation E_x3_s  # Total household demand for composite commodities #
  (all,c,COM)(all,d,DST)(all,h,HOU)
   x3_s(c,d,h) = BLUX(c,d,h)*xlux(c,d,h) + [1-BLUX(c,d,h)]*xsub(c,d,h);

Equation E_alux # Default setting for luxury taste shifter #
  (all,c,COM)(all,d,DST)(all,h,HOU)
   alux(c,d,h) = asub(c,d,h) - sum{k,COM, SLUX(k,d,h)*asub(k,d,h)};

Equation E_asub # Default setting for subsistence taste shifter #
  (all,c,COM)(all,d,DST)(all,h,HOU)
   asub(c,d,h) = ahou_s(c,d,h) - sum{k,COM, BUDGSHR(k,d,h)*ahou_s(k,d,h)};

Equation E_wlux #Regional household demands # (all,d,DST)(all,h,HOU)
   x3tot(d,h)= sum{c,COM,BUDGSHR(c,d,h)*x3_s(c,d,h)};

Equation E_p3tot_c #Regional household price baskets# (all,d,DST)(all,h,HOU)
   p3tot_c(d,h)= sum{c,COM,BUDGSHR(c,d,h)*p3(c,d)};

Equation E_w3tot #Value of regional household consumption#
(all,d,DST)(all,h,HOU)
   w3tot(d,h)= p3tot_c(d,h) + x3tot(d,h);

Equation E_contCPI #Household contributions to CPI #
(all,c,COM)(all,d,DST)(all,h,HOU)
  contCPI(c,d,h)= RATIOCPI(d,h)*BUDGSHR(c,d,h)*p3(c,d);

Equation E_x3_sh  # All-household demand for composite commodities #
  (all,c,COM)(all,d,DST) x3_sh(c,d)= sum{h,HOU, HOUSHR(c,d,h)*x3_s(c,d,h)};

Equation E_xhoutot  # All-household demand #
(all,d,DST) sum{h,HOU, HOUPUR_C(d,h)*[xhoutot(d)-x3tot(d,h)]} = 0;

Equation E_nhouh # Number of households #
 (all,d,DST)(all,h,HOU) nhouh(d,h) = nhou(d);

! national addup of HOU-length variables !
Equation E_natx3tot # National real consumption by household #
(all,h,HOU) sum{d,DST, HOUPUR_C(d,h)*[natx3tot(h)-x3tot(d,h)]} = 0;

Equation E_natp3tot # National CPI by household #
(all,h,HOU) sum{d,DST, HOUPUR_C(d,h)*[natp3tot(h)-p3tot_c(d,h)]} = 0;
```

```
!*************************************************************************!
! Section 8.4.  Export and inventory demands                   !
!*************************************************************************!


Equation E_ff_qexp_d #Links export demands to industry outputs# (all,c,COM)

f_qexp_d(c)=ff_qexp_d(c)+sum(i,IND,ISMADE(c,i)*MAKESHR_i(c,i)*ff_qexp_i(i));

Equation E_xexp #Export demands # (all,c,COM)(all,s,SRC)(all,d,DST)
   xexp(c,s,d) = f_qexp_dc + f_qexp_d(c)+f_qexp_c(d)+f_qexp(c,d)+fqexp(c,s)
- ABS[EXP_ELAST(c)]*[ppur(c,s,"Exp",d)-f_pexp_dc -fpexp(c,s)-f_pexp_d(c)-
phi];



Equation E_xexp_s #Export demands for composite commodities #
(all,c,COM)(all,d,DST)
   xexp_s(c,d) = sum{s,SRC, SRCSHR(c,s,"exp",d)*xexp(c,s,d)};

! Inventories !

Equation E_xstocks #Inventory demands #(all,i,IND)(all,d,DST)
        xstocks(i,d) = x1tot(i,d) + f_stocks(i,d)+ f_stocks_d(i);


!*************************************************************************!
! Section 8.5. Demand for margins                                !
!*************************************************************************!



![[! for optional road-rail substitution!
Set ROADRAIL (TransRodo, TransFerro); ! edit these to match your names !
Subset ROADRAIL is subset of MAR;
Coefficient (all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
 VRoadRail(c,s,r,d) # Road + rail cost #;
Formula     (all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
 VRoadRail(c,s,r,d) = sum{m,ROADRAIL,TRADMAR(c,s,m,r,d)};
Coefficient (all,c,COM)(all,s,SRC)(all,m,ROADRAIL)(all,r,ORG)(all,d,DST)
 RRshr(c,s,m,r,d) # Road/rail share #;
```

```
Zerodivide default 0.5;
Formula     (all,c,COM)(all,s,SRC)(all,m,ROADRAIL)(all,r,ORG)(all,d,DST)
 RRshr(c,s,m,r,d) = TRADMAR(c,s,m,r,d)/VRoadRail(c,s,r,d);
Zerodivide off;
Coefficient (parameter) (all,c,COM) SIGROADRAIL(c) # Road/rail subst elast #;
!Formula     (all,c,COM) SIGROADRAIL(c) = 0.2;!
Read SIGROADRAIL from file REGSUPP header "SGRR";
Coefficient (all,m,MAR) IsRoadRail(m) # Binary dummy #;
Formula     (all,m,MAR) IsRoadRail(m) = 0;
       (all,m,ROADRAIL) IsRoadRail(m) = 1;
Variable    (all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
 pRoadRail(c,s,r,d) # Ave efective road/rail price on good c,s going r to d
#;
Equation E_pRoadRail
(all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
 pRoadRail(c,s,r,d) = sum{m,ROADRAIL,RRshr(c,s,m,r,d)*
                        [psuppmar_p(m,r,d)+atradmar(c,s,m,r,d)]};
Substitute pRoadRail using E_pRoadRail; !]]!

!Demand for margins!

Equation E_xtradmar # Leontief demand for margins #
 (all,c,COM)(all,s,SRC)(all,m,MAR)(all,r,ORG)(all,d,DST)
  xtradmar(c,s,m,r,d) =  xtrad(c,s,r,d) + atradmar(c,s,m,r,d)
! only for optional road-rail substitution:
   - IsRoadRail(m)*SIGROADRAIL(c)*
          [psuppmar_p(m,r,d)+atradmar(c,s,m,r,d) - pRoadRail(c,s,r,d)] !;

Equation E_atradmar # Driver for margin tech change #
 (all,c,COM)(all,s,SRC)(all,m,MAR)(all,r,ORG)(all,d,DST)
  atradmar(c,s,m,r,d) =  atradmar_cs(m,r,d);
Substitute atradmar using E_atradmar;

Equation E_pdelivrd # Delivered price of good c,s from r to d #
(all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST) pdelivrd(c,s,r,d) =
BASSHR(c,s,r,d)*pbasic(c,s,r)
 + sum{m,MAR, MARSHR(c,s,m,r,d)*[psuppmar_p(m,r,d)+atradmar(c,s,m,r,d)]};

! Excerpt 18 of TABLO input file: !
! For each good and destination region, an average user chooses region
sourcing!
! based on delivered (margin-paid, but ex-tax) prices and values !

Equation E_puse # Delivered price of regional composite good c,s to d #
```

```
 (all,c,COM)(all,s,SRC)(all,d,DST)
  ID01(DELIVRD_R(c,s,d))*puse(c,s,d) ! CES price index !
                     = sum{r,ORG,DELIVRD(c,s,r,d)*pdelivrd(c,s,r,d)};

!Equation E_xtrad # CES between goods from different regions #
 (all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
  xtrad(c,s,r,d) = xtrad_r(c,s,d)
                    - SIGMADOMDOM(c)*[pdelivrd(c,s,r,d)-puse(c,s,d)];!

Equation E_xtrad # CES between goods from different regions #
(all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
xtrad(c,s,r,d)=xtrad_r(c,s,d)-SIGMADOMDOM(c)*[pdelivrd(c,s,r,d)-puse(c,s,d)]
 +  sum{k,ORG,  [if(k eq r,1) -
    DELIVRD(c,s,k,d)/{DELIVRD_R(c,s,d)+0.000000000001}]*twistsrc(c,s,k) };

!Supply of margins!

Equation E_xsuppmar_p # Composite margin m on goods passing from r to d #
 (all,m,MAR)(all,r,ORG)(all,d,DST) ! add up demands !
  ID01(TRADMAR_CS(m,r,d))*xsuppmar_p(m,r,d) =
   sum{c,COM, sum{s,SRC, TRADMAR(c,s,m,r,d)*xtradmar(c,s,m,r,d)}};


Equation E_psuppmar_p # Price of composite margin m on goods from r to d #
 (all,m,MAR)(all,r,ORG)(all,d,DST) ! CES price index !
   ID01(SUPPMAR_P(m,r,d))*psuppmar_p(m,r,d) =
      sum{p,PRD, SUPPMAR(m,r,d,p)*[pdom(m,p)+asuppmar(m,r,d,p)]};

Equation E_xsuppmar # Margin m supplied by p on goods passing from r to d #
 (all,m,MAR)(all,r,ORG)(all,d,DST)(all,p,PRD) ! CES demands !
  xsuppmar(m,r,d,p) = xsuppmar_p(m,r,d) + asuppmar(m,r,d,p)
           - SIGMAMAR(m)*[pdom(m,p)+asuppmar(m,r,d,p)-psuppmar_p(m,r,d)];

Equation E_xsuppmar_d # Total margins on goods from r, produced in p #
(all,m,MAR)(all,r,ORG)(all,p,PRD)
ID01(SUPPMAR_D(m,r,p))*xsuppmar_d(m,r,p) =
 sum{d,DST, SUPPMAR(m,r,d,p)*xsuppmar(m,r,d,p)};

Equation E_xsuppmar_rd # Total demand for margins produced in p #
(all,m,MAR)(all,p,PRD) ID01(SUPPMAR_RD(m,p))*xsuppmar_rd(m,p) =
 sum{r,ORG, SUPPMAR_D(m,r,p)*xsuppmar_d(m,r,p)};
```

```
!***************************************************************************!
! Section 8.6. Total regional demand for delivered goods                    !
!***************************************************************************!


Equation E_xtrad_r # Total demand for regional composite c,s in d #
(all,c,COM)(all,s,SRC)(all,d,DST)
  ID01[USE_U(c,s,d)]*xtrad_r(c,s,d) =
               USE_I(c,s,d)  *x1_i(c,s,d)
             + USE(c,s,"hou",d)*x3(c,s,d)
             + USE(c,s,"inv",d)*x2(c,s,d)
             + USE(c,s,"gov",d)*xgov(c,s,d)
             + USE(c,s,"exp",d)*xexp(c,s,d);

Equation E_xlocuse # Total non-export demand for regional composite c,s in d#
(all,c,COM)(all,s,SRC)(all,d,DST)   ID01[USE_U(c,s,d)]*xtrad_r(c,s,d) =
   ID01[LOCUSE(c,s,d)]*xlocuse(c,s,d) + USE(c,s,"exp",d)*xexp(c,s,d);
Backsolve xlocuse using E_xlocuse;

Equation E_xlocuse_s # Total non-export demand for good c in d #
(all,c,COM)(all,d,DST)
ID01[LOCUSE_S(c,d)]*xlocuse_s(c,d) = sum{s,SRC,
LOCUSE(c,s,d)*xlocuse(c,s,d)};
Backsolve xlocuse_s using E_xlocuse_s;

Equation E_xlocuse_sd # National non-export demand for good c #
(all,c,COM)
ID01[LOCUSE_SD(c)]*xlocuse_sd(c) = sum{d,DST, LOCUSE_S(c,d)*xlocuse_s(c,d)};


!***************************************************************************!
! Section 8.7. Definitions of purchasers prices and                         !
! final demand price and quantity indices                                   !
!***************************************************************************!

!              Definition purchasers prices                !

Equation E_pimp (all,c,COM)(all,r,ORG) pimp(c,r)         =
    pfimp(c,r) +f_pimp_d(c)+ f_pimp_dc+phi;
      E_pbasicA (all,c,COM)(all,r,ORG) pbasic(c,"dom",r) = pdom(c,r);
      E_pbasicB (all,c,COM)(all,r,ORG) pbasic(c,"imp",r) = pimp(c,r);

Equation E_ppur # Purchasers prices #
```

```
(all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST)
 ppur(c,s,u,d) = puse(c,s,d) + tuser(c,s,u,d);

Equation E_tuser # Usr x Dst taxes driven by commodity specific shifter #
(all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST)
tuser(c,s,u,d)=tuser_ud(c,s)+tuser_sd(c,u);

Substitute tuser using E_tuser;

Equation E_ppur_s
(all,c,COM)(all,u,USR)(all,d,DST)
          ppur_s(c,u,d) = sum{s,SRC,SRCSHR(c,s,u,d)*ppur(c,s,u,d)};

!Definition of final demand price and quantity indices                  !

Equation
E_pfin (all,u,FINDEM)(all,d,DST)
   ID01(PUR_CS(u,d))*pfin(u,d)=sum{c,COM,PUR_S(c,u,d)*ppur_s(c,u,d)};

E_xfina (all,d,DST)
   xfin("hou",d) = xhoutot(d);

E_xfinb (all,d,DST)
   ID01(PUR_CS("inv",d))*xfin("inv",d)=
           sum{c,COM, PUR_S(c,"inv",d)*x2_s(c,d)};

E_xfinc (all,d,DST)
   ID01(PUR_CS("gov",d))*xfin("gov",d)=
           sum{c,COM, PUR_S(c,"gov",d)*xgov_s(c,d)};

E_xfind (all,d,DST)
   ID01(PUR_CS("exp",d))*xfin("exp",d)=
           sum{c,COM,sum{s,SRC, PUR(c,s,"exp",d)*xexp(c,s,d)}};

E_wfin (all,u,FINDEM)(all,d,DST)  wfin(u,d) = xfin(u,d)+ pfin(u,d);


!***************************************************************************!
! Section 8.8. Market clearing equations for goods                          !
! Total demand for commodity c produced in r = supply commodity c produced in
r!
!***************************************************************************!
```

```
Equation E_xtrad_d
(all,c,COM)(all,s,SRC)(all,r,ORG) ID01(TRADE_D(c,s,r))*xtrad_d(c,s,r) =
                              sum{d,DST, TRADE(c,s,r,d)*xtrad(c,s,r,d)};

Equation E_pdomA # Demand = supply for non-margins #
 (all,c,NONMAR)(all,r,REG)
   xcom(c,r) = xtrad_d(c,"dom",r);

Equation E_pdomB # Demand = supply for margins #
 (all,m,MAR)(all,p,REG)
   MAKE_I(m,p)*xcom(m,p) = TRADE_D(m,"dom",p)*xtrad_d(m,"dom",p)
                          + SUPPMAR_RD(m,p)*xsuppmar_rd(m,p);


!***************************************************************************!
! Section 8.9. Indirect tax revenues !
!***************************************************************************!



Equation
 E_delTAXint (all,c,COM)(all,s,SRC)(all,i,IND)(all,d,DST)
  delTAXint(c,s,i,d) = 0.01*TAX(c,s,i,d)*[x1(c,s,i,d)+puse(c,s,d)]
                + 0.01*PUR(c,s,i,d)*tuser(c,s,i,d);
 E_delTAXhou (all,c,COM)(all,s,SRC)(all,d,DST)
  delTAXhou(c,s,d)    = 0.01*TAX(c,s,"hou",d)*[x3(c,s,d)+puse(c,s,d)]
                + 0.01*PUR(c,s,"hou",d)*tuser(c,s,"hou",d);
 E_delTAXinv (all,c,COM)(all,s,SRC)(all,d,DST)
  delTAXinv(c,s,d)    = 0.01*TAX(c,s,"inv",d)*[x2(c,s,d)+puse(c,s,d)]
                + 0.01*PUR(c,s,"inv",d)*tuser(c,s,"inv",d);
 E_delTAXgov (all,c,COM)(all,s,SRC)(all,d,DST)
  delTAXgov(c,s,d)    = 0.01*TAX(c,s,"gov",d)*[xgov(c,s,d)+puse(c,s,d)]
                + 0.01*PUR(c,s,"gov",d)*tuser(c,s,"gov",d);
 E_delTAXexp (all,c,COM)(all,s,SRC)(all,d,DST)
  delTAXexp(c,s,d)    = 0.01*TAX(c,s,"exp",d)*[xexp(c,s,d)+puse(c,s,d)]
                + 0.01*PUR(c,s,"exp",d)*tuser(c,s,"exp",d);
!****************************************************************************
!
! Section 8.10. Nominal Income-side GDP !
!****************************************************************************
!

!Primary factor aggregates!
```

```
Equation
E_wlnd_i (all,d,DST)
  LND_I(d)*wlnd_i(d) = sum{i,IND,LND(i,d)*[plnd(i,d)+x1lnd(i,d)]};
E_wcap_i (all,d,DST)
  CAP_I(d)*wcap_i(d) = sum{i,IND,CAP(i,d)*[p1cap(i,d)+x1cap(i,d)]};
E_wprim_i (all,d,DST)
  PRIM_I(d)*wprim_i(d) = sum{i,IND,PRIM(i,d)*wprim(i,d)};
E_x1lnd_i (all,d,DST)
  LND_I(d)*x1lnd_i(d) = sum{i,IND,LND(i,d)*x1lnd(i,d)};
E_x1cap_i (all,d,DST)
  CAP_I(d)*x1cap_i(d) = sum{i,IND,CAP(i,d)*x1cap(i,d)};

Equation
 E_delGDPINCa (all,d,DST) delGDPINC(d,"Land")
    = 0.01*LND_I(d)*wlnd_i(d);
 E_delGDPINCb (all,d,DST) delGDPINC(d,"Capital")
    = 0.01*CAP_I(d)*wcap_i(d);
 E_delGDPINCc (all,d,DST) delGDPINC(d,"Labour")
    = 0.01*LAB_IO(d)*wlab_io(d);
 E_delGDPINCd (all,d,DST) delGDPINC(d,"ProdTax") = sum{i,IND,delPTX(i,d)};
 E_delGDPINCe (all,d,DST) delGDPINC(d,"ComTax") = sum{c,COM,sum{s,SRC,
 sum{i,IND,delTAXint(c,s,i,d)}+
  delTAXhou(c,s,d)+delTAXinv(c,s,d)+delTAXgov(c,s,d)+delTAXexp(c,s,d)}};

Equation E_wgdpinc
 (all,d,DST) GDPINC(d)*wgdpinc(d) = 100*sum{i,GDPINCCAT, delGDPINC(d,i)};


!***************************************************************************!
! Section 8.11. Real and nominal expenditure side GDP !
!***************************************************************************!

Equation
 E_delXGDPEXPa (all,d,DST)(all,u,FINDEM) delXGDPEXP(d,u)
    = 0.01*PUR_CS(u,d)*xfin(u,d);
 E_delXGDPEXPb (all,d,DST) delXGDPEXP(d,"Stocks")
    = 0.01*sum{i,IND,STOCKS(i,d)*xstocks(i,d)};
 E_delXGDPEXPc (all,q,REG) delXGDPEXP(q,"Imports")
    =-0.01*sum{c,COM, TRADE_D(c,"imp",q)*xtrad_d(c,"imp",q)};
E_delXGDPEXPd (all,q,REG) delXGDPEXP(q,"NetMar")
    = 0.01*sum{m,MAR, sum{r,ORG,
        sum{d,DST, SUPPMAR(m,r,d,q)*xsuppmar(m,r,d,q)}
      - sum{p,PRD, SUPPMAR(m,r,q,p)*xsuppmar(m,r,q,p)}  }};
```

```
E_delXGDPEXPe (all,q,REG) delXGDPEXP(q,"Rexports")
    = 0.01*sum{c,COM, sum{s,SRC,
        TRADE_D(c,s,q)*xtrad_d(c,s,q)
                 - TRADE(c,s,q,q)*xtrad(c,s,q,q)  }};
E_delXGDPEXPf (all,q,REG) delXGDPEXP(q,"Rimports")
    =-0.01*sum{c,COM, sum{s,SRC,
          TRADE_R(c,s,q)*xtrad_r(c,s,q)
                 - TRADE(c,s,q,q)*xtrad(c,s,q,q)  }};

Equation E_xXGDPEXP (all,d,DST)(all,i,GDPEXPCAT)
  ID01[GDPEXPSUM(d,i)]*xXGDPEXP(d,i)= 100*delXGDPEXP(d,i);

Equation E_xgdpexp
 (all,d,DST) GDPEXP(d)*xgdpexp(d) = 100*sum{i,GDPEXPCAT, delXGDPEXP(d,i)};

Equation
 E_delPGDPEXPa (all,d,DST)(all,u,FINDEM) delPGDPEXP(d,u)
    = 0.01*PUR_CS(u,d)*pfin(u,d);
 E_delPGDPEXPb (all,d,DST) delPGDPEXP(d,"Stocks")
    = 0.01*sum{i,IND, STOCKS(i,d)*ptot(i,d)};
 E_delPGDPEXPc (all,q,REG) delPGDPEXP(q,"Imports")
    =-0.01*sum{c,COM, TRADE_D(c,"imp",q)*pimp(c,q)};
E_delPGDPEXPd (all,q,REG) delPGDPEXP(q,"NetMar")
    = 0.01*sum{m,MAR, SUPPMAR_RD(m,q)*pdom(m,q)
        -  sum{p,PRD, SUPPMAR_R(m,q,p)*pdom(m,p)}};
E_delPGDPEXPe (all,q,REG) delPGDPEXP(q,"Rexports") =
0.01*sum{c,COM,sum{s,SRC,
    [TRADE_D(c,s,q) - TRADE(c,s,q,q)]*pbasic(c,s,q)}};
E_delPGDPEXPf (all,q,REG) delPGDPEXP(q,"Rimports") =-
0.01*sum{c,COM,sum{s,SRC,
    sum{r,ORG, TRADE(c,s,r,q)*pbasic(c,s,r)} -
TRADE(c,s,q,q)*pbasic(c,s,q)}};

Equation E_pgdpexp
 (all,d,DST) GDPEXP(d)*pgdpexp(d) = 100*sum{i,GDPEXPCAT, delPGDPEXP(d,i)};

Equation E_wgdpexp   (all,d,DST) wgdpexp(d) = xgdpexp(d)+pgdpexp(d);

Equation E_wgdpdiff (all,d,DST) wgdpdiff(d) = wgdpinc(d)- wgdpexp(d);

Coefficient (all,d,REG) RATIOGDPEXP(d)# (Current/initial) real expenditure
GDP#;
Formula  (initial)(all,d,REG) RATIOGDPEXP(d) = 1;
Update            (all,d,REG) RATIOGDPEXP(d) = xgdpexp(d);
```

```
Equation E_contxgdpexp (all,i,GDPEXPCAT)(all,d,REG)
   GDPEXP(d)*contxgdpexp(i,d) = 100*RATIOGDPEXP(d)*delXGDPEXP(d,i);

! Indices and components of national expenditure side GDP !


Equation E_delNatXGDPEXP  (all,i,NATGDPEXPCAT)
    delNatXGDPEXP(i) = sum{d,DST, delXGDPEXP(d,i)};

Equation E_xNATXGDPEXP (all,i,NATGDPEXPCAT)
  ID01[NATGDPEXPSUM(i)]*xNATXGDPEXP(i)= 100*delNATXGDPEXP(i);

Equation E_xnatgdpexp
  NATGDPEXP*xnatgdpexp  = 100*sum{i,NATGDPEXPCAT, delNatXGDPEXP(i)};

Equation E_delNatPGDPEXP  (all,i,NATGDPEXPCAT)
      delNatPGDPEXP(i) = sum{d,DST, delPGDPEXP(d,i)};

Equation E_pnatgdpexp
  NATGDPEXP*pnatgdpexp = 100*sum{i,NATGDPEXPCAT, delNATPGDPEXP(i)};

Equation E_wnatgdpexp   wnatgdpexp = xnatgdpexp+pnatgdpexp;

Equation E_wnatgdpinc sum{d,DST, GDPINC(d)*[wgdpinc(d)-wnatgdpinc]} = 0;

Equation E_wnatgdpdiff  wnatgdpdiff  = wnatgdpinc - wnatgdpexp;

Equation E_contxnatgdpexp (all,i,NATGDPEXPCAT)
   NATGDPEXP*contxnatgdpexp(i) = 100*RATIONGDPEXP*delNATXGDPEXP(i);


!Regional macro reporting variables !


Equation E_ximps (all,c,COM)(all,d,DST) ximps(c,d) =xtrad_r(c,"imp",d);

Equation E_ximpused (all,d,DST) IMPUSED_C(d)*ximpused(d) =
              sum{c,COM,TRADE_R(c,"imp",d)*ximps(c,d)};
Equation E_pimpused (all,d,DST) IMPUSED_C(d)*pimpused(d) =
              sum{c,COM, sum{r,ORG,TRADE(c,"imp",r,d)*pimp(c,r)}};

Equation E_ximplanded (all,r,ORG) ID01[IMPLANDED_C(r)]*ximplanded(r) =
```

```
              sum{c,COM,TRADE_D(c,"imp",r)*xtrad_d(c,"imp",r)};
Equation E_pimplanded (all,r,ORG) ID01[IMPLANDED_C(r)]*pimplanded(r) =
              sum{c,COM,TRADE_D(c,"imp",r)*pimp(c,r)};




!**************************************************************************!
! Section 8.12. National price and quantity indices  !
!*************************************************************************!


Equation E_natxtot  (all,i,IND)
   PRIM_D(i)*natxtot(i) =  sum{d,DST,PRIM(i,d)*x1tot(i,d)};

Equation E_contnatxtot (all,i,IND)(all,d,DST)
   PRIM_D(i) *contnatxtot(i,d) = RATIOPRIM_D(i)*PRIM(i,d)*x1tot(i,d);

Equation E_xtrad_rd
 (all,c,COM)(all,s,SRC) ID01[TRADE_RD(c,s)]*xtrad_rd(c,s) =
   sum{d,DST, TRADE_R(c,s,d)*xtrad_r(c,s,d)};

Equation E_natximp
 (all,c,COM) natximp(c) = xtrad_rd(c,"imp");

Equation E_natxcom
 (all,c,COM) natxcom(c) = sum{r,REG, [MAKE_I(c,r)/MAKE_IR(c)]*xcom(c,r)};

Equation
E_natptot (all,i,IND)
  ID01[NATVTOT(i)]*natptot(i) = sum{d,DST,VTOT(i,d)*ptot(i,d)};
E_natemploy (all,i,IND)
  ID01[LAB_OD(i)]*natemploy(i) = sum{d,DST,LAB_O(i,d)*x1lab_o(i,d)};
E_natwprim (all,i,IND)
  ID01[PRIM_D(i)]*natwprim(i) = sum{d,DST,PRIM(i,d)*wprim(i,d)};
!Backsolve natptot   using E_natptot;
Backsolve natemploy using E_natemploy;!
Backsolve natwprim  using E_natwprim;



!************************************************************************!
!Section 8.13. TRADE FLOWS                                               !
```

```
!*************************************************************************!


! Exports to ROW by region of production !
! Strictly speaking, the concept of "Exports by region of production" is not
  part of the TERM model. However, we can estimate them by making additional
  common-sense assumptions [source share for exports is same as for overall
  demand !

! Exports to ROW by region of production !

Equation E_xrowdem (all,c,COM)(all,r,ORG)(all,d,DST)
  xrowdem(c,r,d) = xexp(c,"dom",d) - xtrad_r(c,"dom",d) + xtrad(c,"dom",r,d);

Equation E_xrowdem_d (all,c,COM)(all,r,ORG)
 ID01[ROWDEM_D(c,r)]*xrowdem_d(c,r) = sum{d,DST,
ROWDEM(c,r,d)*xrowdem(c,r,d)};

!   Domestic trade flows    !

Equation E_xdomexp
(all,c,COM)(all,r,REG!ORG!) ID01[VDOMEXP(c,r)]*xdomexp(c,r) =
 TRADE_D(c,"dom",r)*xtrad_d(c,"dom",r) -
TRADE(c,"dom",r,r)*xtrad(c,"dom",r,r);

Equation E_xdomimp
(all,c,COM)(all,d,REG) ID01[VDOMIMP(c,d)]*xdomimp(c,d) =
 TRADE_R(c,"dom",d)*xtrad_r(c,"dom",d) -
TRADE(c,"dom",d,d)*xtrad(c,"dom",d,d);

Equation E_xdomexp_c
(all,r,REG) ID01[VDOMEXP_C(r)]*xdomexp_c(r) =
  sum{c,COM,VDOMEXP(c,r)*xdomexp(c,r)};

Equation E_xdomimp_c
(all,d,REG) ID01[VDOMIMP_C(d)]*xdomimp_c(d) =
  sum{c,COM,VDOMIMP(c,d)*xdomimp(c,d)};

Equation E_xdomloc
(all,c,COM)(all,r,REG) xdomloc(c,r) = xtrad(c,"dom",r,r);


!**************************************************************************!
!* Section 8.14.       Public sector accounts                             *!
```

```
!**************************************************************************!

!Subsection 8.14.1 Government, municipal, and social security fund      !
!expenditures, revenues and deficits                                     !

Equation E_d_gov_defs1
  # Public sector deficit, or public sector financing transactions #
(all,d,DST)(all,s,GSEC)
  100*d_gov_defs(d,s) =
                V5TOTS(d,s)*w5tots(d,s) + V2TOT_G_IS(d,s)*w2totg_is(d,s)
                +      100*d_othcapgovs(d,s)
                    - NET_TAXTOTS(d,S)*nettaxtots(d,s)
                    - OTHGOVREVS(s,d)*oth_govrevs(d,s) + 100*d_transfs(d,s)
                +GOVTOMUN(d)*f_govtomun(d)
                +GOVTOSSF(d)*f_govtossf(d)
                -MUNTOGOV(d)*f_muntogov(d)
                -SSFTOGOV(d)*f_ssftogov(d)
;

Equation E_d_gov_defs2
  # Public sector deficit, or public sector financing transactions #
(all,d,DST)(all,s,LSEC)
  100*d_gov_defs(d,s) =
                V5TOTS(d,s)*w5tots(d,s) + V2TOT_G_IS(d,s)*w2totg_is(d,s)
                +      100*d_othcapgovs(d,s)
                - NET_TAXTOTS(d,S)*nettaxtots(d,s)
                - OTHGOVREVS(s,d)*oth_govrevs(d,s) + 100*d_transfs(d,s)
                +MUNTOGOV(d)*f_muntogov(d)
                -GOVTOMUN(d)*f_govtomun(d);

Equation E_d_gov_defs3
  # Public sector deficit, or public sector financing transactions #
(all,d,DST)(all,s,FSEC)
  100*d_gov_defs(d,s) =
                V5TOTS(d,s)*w5tots(d,s) + V2TOT_G_IS(d,s)*w2totg_is(d,s)
                +      100*d_othcapgovs(d,s)
                - NET_TAXTOTS(d,S)*nettaxtots(d,s)
                - OTHGOVREVS(s,d)*oth_govrevs(d,s) + 100*d_transfs(d,s)
                +SSFTOGOV(d)*f_ssftogov(d)
                -GOVTOSSF(d)*f_govtossf(d);
```

```
Equation E_d_gov_def
  # Public sector deficits, or public sector financing transactions #
(all,s,PSEC)
d_gov_def(s) = sum(d,DST,d_gov_defs(d,s));

Equation E_d_gov_def_nat
  # Combined public sector deficit #
d_gov_def_nat = sum(s,PSEC,d_gov_def(s));

Equation E_d_othcapgov_nat
  # Combined public sector deficit #
d_othcapgov_nat = sum[d,DST,sum(s,PSEC,d_othcapgovs(d,s))];


Equation E_xgov #Public sector demands #(all,c,COM)(all,s,SRC)(all,d,DST)
       xgov(c,s,d) =  fgovtot(d) + fgov(c,s,d) + fgov_s(c,d) + fgovgen;

Equation E_xgov_s # Public demand for composite goods #(all,c,COM)(all,d,DST)
       xgov_s(c,d) = sum{s,SRC, SRCSHR(c,s,"Gov",d)*xgov(c,s,d)};


Equation E_fgov_s #shifter for govt consumption#
(all,c,COM)(all,d,DST)
fgov_s(c,d)=sum(s,PSEC,V5SHRS_(d,c,s)*f5totr(d,c,s));


Equation E_f5totr #shifter for govt consumption#
(all,d,DST)(all,c,COM)(all,s,PSEC)

f5totr(d,c,s)=d_f5totr(d,c,s)+d_f5totrc(d,s)+d_f5totrcd(s)+d_f5totrds(c)
+d_f5reg(d,c,s)+d_f5totrs(d,c);

Equation E_x5totr #shifter for govt consumption#
(all,c,COM)(all,s,SRC)(all,d,DST)(all,z,PSEC)
x5totr(c,s,d,z)=SCRSHR5(d,c,s,z)*f5totr(d,c,z);

Equation E_x5tots (all,d,DST)(all,s,PSEC)
V5TOTS(d,s)*x5tots(d,s)=sum(c,COM,V5TOT(d,c,s)*xgov_s(c,d));

Equation E_p5tots (all,d,DST)(all,s,PSEC)
V5TOTS(d,s)*p5tots(d,s)=sum(c,COM,V5TOT(d,c,s)*ppur_s(c,"gov",d));

Equation E_w5tots (all,d,DST)(all,s,PSEC)
w5tots(d,s) = x5tots(d,s) + p5tots(d,s);
```

```
Equation E_d_w5tots (all,d,DST)(all,s,PSEC)
100*d_w5tots(d,s) = V5TOTS(d,s)*w5tots(d,s);

Equation E_d_w5tots_d (all,s,PSEC)
d_w5tots_d(s) = sum(d,DST,d_w5tots(d,s));

Equation E_d_w5tots_ds
d_w5tots_ds = sum(s,PSEC,d_w5tots_d(s));

Equation E_w2totg_is # Value of public sector investments #
(all,d,DST)(all,s,PSEC)
V2TOT_G_IS(d,s)*w2totg_is(d,s) =
!sum{i,IND,V2TOTSG_(d,i,s)*w2totg_s(d,i,s)};!
sum{i,IND,V2SHRSG_(d,i,s)*INVEST_C(i,d)*(s2gov(i)+pinvitot(i,d)+x2tot(i,d))};


Equation E_w2totg_is_ds # Value of public sector investments #
V2TOT_G_I_D*w2totg_is_ds = SUM[d,DST,SUM(s,PSEC,
V2TOT_G_IS(d,s)*w2totg_is(d,s))];

Equation E_delV0tax_csi
(all,d,DST) delV0tax_csi(d) =

sum{c,COM, sum{s,SRC, sum{i,IND, delTAXint(c,s,i,d)}}}
+
 sum{c,COM, sum{s,SRC, delTAXhou(c,s,d)}}
+
 sum{c,COM, sum{s,SRC, delTAXinv(c,s,d)}}
+
 sum{c,COM, sum{s,SRC, delTAXgov(c,s,d)}}
+
 sum{c,COM, sum{s,SRC, delTAXexp(c,s,d)}}
+
 sum{i,IND, delPTX(i,d)};

Equation E_w0tax_csis #Indeirec tax revenue#
(all,d,DST)(all,s,PSEC)
[TINY+V0TAX_CSIS(d,s)]*w0tax_csis(d,s) = 100*delV0tax_csis(d,s);

Equation E_delV0tax_csis #Indirec tax collection#
(all,d,DST)(all,s,GSEC) delV0tax_csis(d,s)=delV0tax_csi(d);
```

```
Equation E_d_othcapgovs (all,d,DST)(all,s,PSEC)
100*d_othcapgovs(d,s)=OTHCAPGOVS_(s,D)*f_othcapgovs(d,s)+d_f_othcapgovs(d,s);

Equation E_f_othcapgovs (all,d,DST)(all,s,PSEC)
f_othcapgovs(d,s)=wgdpexp(d)+f_othcapgov(d,s);


Equation E_d_col_payrs # Total collection of payroll taxes #
(all,d,DST)(all,s,PSEC) 100*d_col_payrs(d,s)
   = sum{i,IND,[[POW_PAYROLLS(s,i,d)-
1]/POW_PAYROLL(i,d)]*LAB_O(i,d)*wlab_o(i,d)
    + [LAB_O(i,d)/POW_PAYROLL(i,d)]*powpayrolls(d,i,s)};

Equation E_nettaxtots
  # Net collection of genuine indirect taxes and income taxes #
(all,d,DST)(all,s,PSEC)  NET_TAXTOTS(d,S)*nettaxtots(d,s)
        = V0TAX_CSIS(d,s)*w0tax_csis(d,s)  + INCTAXS(d,s)*taxrev_incs(d,s)
        + 100*d_col_payrs(d,s) ;


Equation E_d_nettaxtots (all,d,DST)(all,s,PSEC)
100*d_nettaxtot(d,s)=NET_TAXTOTS(d,s)*nettaxtots(d,s);



Equation E_d_nettaxtot_d (all,s,PSEC)
100*d_nettaxtot_d(s)=sum(d,DST,d_nettaxtot(d,s));


Equation E_oth_govrevs
  # Other government revenue, e.g. income from public enterprises #
(all,d,DST)(all,s,GSEC)  oth_govrevs(d,s) =
!wgdpexp(d)+!wnatgdpexp+f_oth_grevs(d,s);

Equation E_oth_govrevs2
  # Other government revenue, e.g. income from public enterprises #
(all,d,DST)(all,s,LSEC)  oth_govrevs(d,s) =
wgdpexp(d)!+wnatgdpexp!+f_oth_grevs(d,s);

Equation E_oth_govrevs3
  # Other government revenue, e.g. income from public enterprises #
(all,d,DST)(all,s,FSEC)  oth_govrevs(d,s) =
```

```
!wgdpexp(d)+!wnatgdpexp+f_oth_grevs(d,s);


Equation E_d_othgovrev (all,d,DST)(all,s,PSEC)
100*d_othgovrev(d,s)=OTHGOVREVS(s,d)*oth_govrevs(d,s);

Equation E_d_othgovrev_d (all,s,PSEC)
d_othgovrev_d(s)=sum(d,DST,d_othgovrev(d,s));

Equation E_d_transfs # Transfers from the government #
(all,d,DST)(all,s,PSEC) d_transfs(d,s) =

0.01*UNEMPBENS(s,d)*unemp_bens(d,s)+0.01*AGEBENS(s,d)*age_bens(d,s)
            + 0.01*OTHBENS(s,d)*oth_bens(d,s)+0.01*GRANTSS(s,d)*grants_s(d,s)
            + d_net_int_gs(d,s);

Equation E_taxrev_incs # Income tax revenue #
(all,d,DST)(all,s,PSEC) [TINY+INCTAXS(d,s)]*taxrev_incs(d,s) =
            TAXS_LAB(d,s)*[wlab_io(d) + tax_l_r(d,s) ]+
          + TAXS_CAP(d,s)*[wcap_i(d) + tax_k_r(d,s)]
          + TAXS_LND(d,s)*[wlnd_i(d) + tax_k_r(d,s)]
          + TAXS_AB(d,s)*[age_bens(d,s)  + tax_ab_r(d,s)]
          + TAXS_OB(d,s)*[oth_bens(d,s) + tax_ob_r(d,s)]
          + TAXS_UB(d,s)*[unemp_bens(d,s) + tax_ub_r(d,s)];

Equation E_unemp_bens # Unemployment benefits  #
(all,d,DST)(all,s,PSEC)  unemp_bens(d,s) = !eligsh +! unempben_rats(d,s)
 +[1/[LAB_SUP(d) - EMPLOYS(d)]]*[LAB_SUP(d)*labsup(d)-
EMPLOYS(d)*x1lab_io(d)]
 +f_unemp_bens(d,s);

Equation E_realwage_iod #National real wage change#
(sum{d,DST, LAB_IO(d)})*realwage_iod = sum{d,DST,LAB_IO(d)*realwage_io(d)};

Equation E_p3tot #Alias for national consumer price index#


p3tot=natp3tot("Allhou");


Equation E_unempben_rats # Unemployment benefit rate #
(all,d,DST)(all,s,PSEC)  unempben_rats(d,s)=
p3tot+realwage_iod + f_unempben_rats(d,s);
```

```
Equation E_age_bens # Old age benefits paid by government #
(all,d,DST)(all,s,PSEC) age_bens(d,s) =
!popaged(d)+p3tot+realwage_iod+f_age_bens(d,s)+f_age_ben_nat;!
popaged(d)-popagednat+f_age_bens(d,s)+f_age_ben_nat;

Equation E_oth_bens # Other personal benefits paid by government #
(all,d,DST)(all,s,PSEC) oth_bens(d,s) =
!pops(d) + p3tot +realwage_iod + f_oth_bens(d,s)+f_oth_ben_nat;!
pops(d)-popnat+f_oth_bens(d,s)+f_oth_ben_nat;

Equation E_grants_s # Grants & transf. oth than unemp age & oth bens #
(all,d,DST)(all,s,PSEC)  grants_s(d,s) =
!wnatgdpexp + f_grants_s(d,s);!
pops(d)-popnat+f_grants_s(d,s)+f_grants_nat;

Equation E_d_f_govtossf # Other personal benefits paid by government #
(all,d,DST) f_govtossf(d) = age_bens(d,"S1311")+ d_f_govtossf(d);
!popaged(d) + p3tot +realwage_iod + d_f_govtossf(d);!

Equation E_d_f_muntogov # Other personal benefits paid by government #
(all,d,DST) f_muntogov(d) = nettaxtots(d,"S1314") + d_f_muntogov(d);
!w5tots(d,"S1313")+d_f_muntogov(d);!

Equation E_d_f_ssftogov # Other personal benefits paid by government #
(all,d,DST) f_ssftogov(d) = age_bens(d,"S1314")+d_f_ssftogov(d);

Equation E_d_int_psd  # Nominal rate of interest on public sector debt #
  d_int_psd = [1+INF]*d_rint_psd + [1+RINT_PSD]*d_inf;

Equation E_d_rint_psd
  # Link between real rates of interest on PSD and business borrowing #
  d_rint_psd = d_rint + d_f_rint_psd;

Equation E_d_net_int_gs # Net interest payments from government #
(all,d,DST)(all,s,PSEC) d_net_int_gs(d,s) =
        [[PSDATT(s,d)+PSDATTPLUS1(d,s)]/2]*d_int_psd
             + 0.5*INT_PSD*[d_psd_ts(d,s) + d_psd_t1s(d,s)];

Equation E_d_net_int_gd # Net interest payments from government #
(all,s,PSEC)
d_net_int_gd(s) =sum{d,DST,
        [[PSDATT(s,d)+PSDATTPLUS1(d,s)]/2]*d_int_psd
             + 0.5*INT_PSD*[d_psd_ts(d,s) + d_psd_t1s(d,s)]};
```

```
Equation E_d_transfs_h # Transfers from the government #

(all,d,DST)(all,s,PSEC) d_transfs_h(d,s) =

0.01*UNEMPBENS(s,d)*unemp_bens(d,s)+0.01*AGEBENS(s,d)*age_bens(d,s)
            + 0.01*OTHBENS(s,d)*oth_bens(d,s)+0.01*GRANTSS(s,d)*grants_s(d,s)
            + SHRASS_H(d)*d_net_int_gd(s);

Equation E_d_psd_t1s # Public sector debt, end of year #
(all,d,DST)(all,s,PSEC)
d_psd_t1s(d,s) = d_psd_ts(d,s) + d_gov_defs(d,s)+d_f_psd_t1s(d,s);

Equation E_d_f_psd_ts
  # Gives shock to start-of-year public sector debt, yr-to-yr sims #
(all,d,DST)(all,s,PSEC)
d_psd_ts(d,s) = [PSDATT_1_B(d,s) - PSDATT_B(d,s)]*d_unity + d_f_psd_ts(d,s);

Equation E_d_psd_ts_d
  # Change in aggregate start-of-year public sector debt #
(all,s,PSEC)
d_psd_ts_d(s) = sum(d,DST,d_psd_ts(d,s));

Equation E_d_psd_t1s_d
  # Change in aggregate start-of-year public sector debt #
(all,s,PSEC)
d_psd_t1s_d(s) = sum(d,DST,d_psd_t1s(d,s));

Equation E_d_psd_ts_nat
  # Change in aggregate start-of-year public sector debt #
d_psd_ts_nat = sum(s,PSEC,d_psd_ts_d(s));


!Public sector saving!

!reporting variable for overall regional public sector demand!

Variable (all,d,REG) fgovtot2(d) # Government demand shifter #;

Equation E_fgovtot2
 (all,d,REG) fgovtot(d) = fgovtot2(d) +  MainMacro("RealHou",d);

! with fgovtot2 exogenous, fgovtot endogenous, reg. real gov spending demand
  follows regional real household demand !
```

```
!***************************************************************************!
!Section 8.15. Income and saving aggregates!
!***************************************************************************!

!
NAT_SAV      = HOUS_SAV - GOV_DEF + V2TOT_G_I + OTHCAPGOV;
NAT_SAV      = HOUS_SAV - GOV_DEF + V2TOT_G_I + OTHCAPGOV;
GOV_SAV      = NAT_SAV - HOUS_SAV;
VCAB = [V4TOT + C_INTAD +C_INTAE] - [V0IMP_C +  C_INTLD + C_INTLE ] + NETTRN;
!

! Consumption function !

Variable natfhou # National ratio, nominal household consumption to GDP #;
Equation E_natfhou   NatMacro("NomHou") = natfhou + NatMacro("NomGDP");

Equation E_hdispinc # Household disposable income #
(all,d,DST) HOUS_DIS_INC(d)*hdispinc(d) =
    GDPEXP(d)*wgdpexp(d)
+ (sum{s,PSEC,100*d_transfs_h(d,s)
   - {NET_TAXTOTS(d,s)*nettaxtots(d,s) + OTHGOVREVS(s,d)*oth_govrevs(d,s)}})
   +SHRASS_H(d)*[C_INTAD*intad+C_INTAE*intae -C_INTLD*intld - C_INTLE*intle];

Equation E_hdispinc_d # Household disposable income #
(sum(d,DST,HOUS_DIS_INC(d)))*hdispinc_d=sum(d,DST,HOUS_DIS_INC(d)*hdispinc(d)
);

Equation E_housav # Household saving #
(all,d,DST) HOUS_SAV(d)*housav(d) =
    HOUS_DIS_INC(d)*hdispinc(d) - V3TOT(d)*wfin("Hou",d);

Equation E_housav_d # National saving #
sum(d,DST, HOUS_SAV(d))*housav_d = sum(d,DST, HOUS_SAV(d)*housav(d));


Equation E_w3tot_d
w3tot_d=natx3tot("Allhou")+natp3tot("Allhou");


!
Equation E_f_f3tot_d # Consumption related to disposable income #
  V3TOT_D*w3tot_d = NAT_PROP_CON*HOUS_DISINCD*[f_f3tot_d + hdispinc_d];
!
```

```
Coefficient (parameter) adjprop;

Read adjprop from file EXTRA header "adjp";

Equation E_f_f3tot
(all,d,DST)
(AV_PROP_CON(d)/NAT_PROP_CON)*(f3tot(d)-f_f3tot_d)=
100*ADJPROP*(1-AV_PROP_C_B(d)/NAT_PROP_C_B)*d_unity+f_f3tot(d)+f3tot_d;

Equation E_f3tot # Consumption related to disposable income #

(all,d,DST)(all,h,HOU)  HOUPUR_C(d,h)*w3tot(d,h) =
    AV_PROP_CON(d)*HOUS_DIS_INC(d)*[f3tot(d) + hdispinc(d)];


!***************************************************************************!
!Traces implications of f3tot !
Variable
 (all,d,DST)(all,h,HOU) f3(d,h)
 # Regional propensity to consume from labour income #;

Equation E_fhou (all,d,DST)(all,h,HOU)
   w3tot(d,h)  =  wlab_io(d) + f3(d,h);



!Section trade balance and balance of payments!

!Trade balance!



Equation
 E_x4tot     V4TOT_D*x4tot = sum{d,DST,PUR_CS("exp",d)*xfin("exp",d)};
 E_p4tot     V4TOT_D*p4tot = sum{d,DST,PUR_CS("exp",d)*pfin("exp",d)};
 E_w4tot     w4tot = x4tot + p4tot;

E_x0imp_c  V0IMP_C_D*x0imp_c = sum{d,DST,IMPUSED_C(d)*ximpused(d)};
E_p0imp_c  V0IMP_C_D*p0imp_c = sum{d,DST,IMPUSED_C(d)*pimpused(d)};
E_w0imp_c          w0imp_c = x0imp_c + p0imp_c;
E_p0toft            p0toft = p4tot - p0imp_c;
E_p0realdev      p0realdev =  p0imp_c - p0gdpexp;
```

```
E_delB 100*V0GDPEXP*delB=
V4TOT_D*w4tot-V0IMP_C_D*w0imp_c-[V4TOT_D-V0IMP_C_D]*w0gdpexp;

E_x0gdpexp x0gdpexp = xnatgdpexp;
E_p0gdpexp p0gdpexp = pnatgdpexp;
E_w0gdpexp w0gdpexp = x0gdpexp + p0gdpexp;


Equation E_gnpnom
GNP*gnpnom  = V0GDPEXP*w0gdpexp + C_INTAD*intad +C_INTAE*intae
               - C_INTLD*intld - C_INTLE*intle + 100*d_nettrn;


Equation E_natsav
  # National saving: household saving plus public sector surplus #
  NAT_SAV*natsav = HOUS_SAV_d*housav_d +100*d_govsav_nat;

Equation E_d_govsav_nat # Government saving #
  d_govsav_nat = - d_gov_def_nat
                    + 0.01*V2TOT_G_I_D*w2totg_is_ds
                    + d_othcapgov_nat;


!Accumulation of foreign assets!

Equation E_d_dum_year1 # One in year one, zero in later years #
d_dum_year1 = ADJDUMYEAR1*d_unity;



Equation E_d_nettrn # Evolution of net transfers #
d_nettrn = V0GDPEXP*d_f_trn +0.01*NETTRN*w0gdpexp;


Equation E_a1e
C_A1E*a1e = C_A0E*a0e  + (C_FE*NAT_SAV)*(fe + natsav) + C_A0E*C_REVAE*a0e
           + 100*C_A0E*d_revae;

Equation E_a1d
C_A1D*a1d = C_A0D*a0d  + (C_FD*NAT_SAV)*(fd + natsav) + C_A0D*C_REVAD*a0d
           + 100*C_A0D*d_revad;

Equation E_l1e
C_L1E*l1e = C_L0E*l0e  + 100*d_newle + C_L0E*C_REVLE*l0e
```

```
           + 100*C_L0E*d_revle;

Equation E_l1d
C_L1D*l1d = C_L0D*l0d  + 100*d_newld + C_L0D*C_REVLD*l0d
           + 100*C_L0D*d_revld;

Equation E_d_newld
  ratio_led = l1e - l1d;

Equation E_a0e
C_A0E*a0e = 100*(C_A1E_B - C_A0E_B)*d_unity;

Equation E_a0d
C_A0D*a0d = 100*(C_A1D_B - C_A0D_B)*d_unity;

Equation E_l0e
C_L0E*l0e = 100*(C_L1E_B - C_L0E_B)*d_unity;

Equation E_l0d
C_L0D*l0d = 100*(C_A1D_B - C_L0D_B)*d_unity;


Equation E_d_newle
100*d_cab = C_FE*NAT_SAV*(fe+natsav) + C_FD*NAT_SAV*(fd+natsav)
                      - 100*d_newle - 100*d_newld;
Equation E_d_cab
100*d_cab =
  [V4TOT_D*w4tot + C_INTAD*intad +C_INTAE*intae]
   - [V0IMP_C_D*w0imp_c +  C_INTLD*intld + C_INTLE*intle ] +100*d_nettrn;

Equation E_intad
   intad = roiad + a0d;
Equation E_intae
   intae = roiae + a0e;
Equation E_intld
   intld = roild + l0d;
Equation E_intle
   intle = roile + l0e;


![[!
!
Equation E_hdispinc # Household disposable income #
(all,d,DST) HOUS_DIS_INC(d)*hdispinc(d) =
```

```
    GDPEXP(d)*wgdpexp(d)
+ (sum{s,PSEC,100*d_transfs(d,s)
   - {NET_TAXTOTS(d,s)*nettaxtots(d,s) + OTHGOVREVS(s,d)*oth_govrevs(d,s)}});

Equation E_delS

100*V0GDPEXP*delS=HOUS_SAV*housav_d-HOUS_SAV*w0gdpexp;

Equation E_housav_r # Real household saving #
  housav_r = housav_d - p2tot_i;

Equation E_natsav
  # National saving: household saving plus public sector surplus #
  NAT_SAV*natsav = HOUS_SAV*housav_d +100*d_govsav;

Equation E_d_govsav # Government saving #
  d_govsav = - d_gov_def
                  + 0.01*V2TOT_G_I*w2tot_g_i + d_othcapgov;

Equation E_natsav_r # Real national saving #
  natsav_r = natsav - p2tot_i;

Equation E_d_govsav_r # Real government saving #
  d_govsav_r = d_govsav - 0.01*GOV_SAV*p2tot_i;
!
!]]!


!***************************************************************************!
! section 8.16. Investment over time!
!***************************************************************************!



! Subsection 8.16.1.  Capital stocks and investment          !



! Capital investment accumulation relationship                              !
! The underlying levels equation is:                                        !
! QCAPATT(i) = QCAPATT_B(i)*(1-DEP(i)) + QINV_BASE(i)                       !

!Equation E_d_dum_year1 # One in year one, zero in later years #
```

```
d_dum_year1 = ADJDUMYEAR1*d_unity;!

Equation E_d_f_ac_p_y
  # Gives shock in yr-to-yr forecasting to capital at beginning of year t #
(all,d,DST)(all,i,IND) [QCAPATT(d,i) + TINY]*x1cap(i,d)!x1cap(d,i)!=
     100*{QINV_BASE(d,i) -
DEP(d,i)*QCAPATT_B(d,i)}*d_unity+100*d_f_ac_p_y(d,i);

Equation E_x1cap_tplus1
  # Capital accum thru the fcst year (t) related to investment in the year #
(all,d,DST)(all,i,IND) [QCAPATTPLUS1(d,i) + TINY]*x1cap_tplus1(d,i)
           = [1-DEP(d,i)]*QCAPATT(d,i)*x1cap(i,d) + QINVEST(d,i)*x2tot(i,d);

! this equation defines the q'ty of investments needed to satisfy the
  growth rate of capital stock in period (t+1) spesified by inverse
  logistic relatioship !

Equation E_x1cap_id
V1CAP_I*x1cap_id   =
sum[d,DST,sum{i,IND, V1CAP(d,i)*x1cap(i,d)}];

Equation E_x2tot # Investment/capital ratios by industry #
(all,d,DST)(all,i,IND) x2tot(i,d) = !x1cap(d,i)!
      x1cap(i,d) + r_inv_cap_i(d,i) + r_inv_cap_d(i)+r_inv_cap(d) +
r_inv_cap_u;

!Equation E_xinvitot (all,i,IND)(all,d,DST) xinvitot(i,d) = x2tot(d,i);!

Equation E_p2tot #Alias p2tot price of investment# (all,d,DST)(all,i,IND)
 p2tot(d,i)=pinvitot(i,d);

Equation E_x2tot_i (all,d,DST)
(sum{i,IND,INVEST_C(i,d)})*x2tot_i(d)= sum{i,IND,INVEST_C(i,d)*x2tot(i,d)};

Equation E_p2tot_i (all,d,DST)
(sum{i,IND,INVEST_C(i,d)})*p2tot_i(d)= sum{i,IND,INVEST_C(i,d)*p2tot(d,i)};

Equation E_x2tot_d (all,i,IND)
(sum{d,DST,INVEST_C(i,d)})*x2tot_d(i)= sum{d,DST,INVEST_C(i,d)*x2tot(i,d)};

Equation E_p2tot_d (all,i,IND)
(sum{d,DST,INVEST_C(i,d)})*p2tot_d(i)= sum{d,DST,INVEST_C(i,d)*p2tot(d,i)};

Equation E_f_x2tot_d
```

```
(all,i,IND) x2tot_d(i) = f_x2tot_d(i)+f_x2tot_id;

Equation E_pcapatt
  # Gives shock to start-of-year asset prices in year-to-year simulations #
(all,d,DST)(all,i,IND) PCAP_AT_T(d,i)*pcapatt(d,i)
     = 100*[PCAP_AT_T1_B(d,i) - PCAP_AT_T_B(d,i)]*d_unity
+100*d_f_pcapatt(d,i);

Equation E_pcapatt1 # End-of-year asset prices in year-to-year sims #
(all,d,DST)(all,i,IND) PCAP_AT_T1(d,i)*pcapatt1(d,i)=
  [1 + 0.5]*[PCAP_I(d,i)*[PCAP_I(d,i)/PCAP_I_B(d,i)]^(0.5)]*p2tot(d,i)
+ 100*[PCAP_I_B(d,i) - PCAP_AT_T1_B(d,i)]*d_unity + 100*d_ff_pcapatt1(d,i);


! section 8.16.2. The inverse logistic relationships between expected rates
of
  return and rates of capital growth !

Variable
(Change)(all,i,IND) d_f_eeqror_d(i);


Equation E_d_f_eeqror
   # Change in equilibrium expected rate of return in forecast year #
(all,d,DST)(all,i,IND) d_eeqror(d,i) = [1/COEFF_SL(d,i)]*
    [1/[K_GR(d,i)-K_GR_MIN(d,i)]+1/[K_GR_MAX(d,i)-K_GR(d,i)]]*d_k_gr(d,i)
                   + d_f_eeqror_i(d,i) + d_f_eeqror(d)+d_f_eeqror_d(i);

Equation E_d_k_gr # Capital growth thru forecast year #
(all,d,DST)(all,i,IND)
   d_k_gr(d,i) = [{QCAPATTPLUS1(d,i)/QCAPATT(d,i)}/100]*
                 [x1cap_tplus1(d,i) - !x1cap(d,i)!x1cap(i,d)];

Equation E_ch_kgr1
  # Provides convenient method for viewing the values of CHKGR1 #
(all,d,DST)(all,j,IND) ch_kgr1(d,j) = CHKGR1(d,j)*d_unity;

Equation E_ch_kgr2
  # Provides convenient method for viewing the values of CHKGR2 #
(all,d,DST)(all,j,IND) ch_kgr2(d,j) = CHKGR2(d,j)*d_unity;

! section 8.16.3 Expected and actual rates of return, and the algorithm for
  imposing forward-looking expectations  parameter!
```

```
Equation E_d_ror_se
       # Changes in expected rors by industry: static exp. #
(all,d,DST)(all,i,IND) 100*d_ror_se(d,i) = [1/[1 + RINT_PT_SE]]*
      {[V1CAP(d,i)*[1 - TAX_K_RATE]/VCAP_AT_TM(d,i)]*[p1cap(i,d) -
p2tot(d,i)]
       - TAX_K_RATE*{[V1CAP(d,i)/VCAP_AT_TM(d,i)] - RALPH*DEP(d,i)}*tax_k_rr
    -[V1CAP(d,i)*[1-TAX_K_RATE]/VCAP_AT_TM(d,i)
    +1-DEP(d,i)+RALPH*TAX_K_RATE*DEP(d,i)]*
            (1/[[1 + RINT_PT_SE]])*100*d_rint_pt_se };

Equation E_d_rint_pt_se
      # Changes in real post-tax rate of interest, static expectations #
100*d_rint_pt_se = [1/[1+INF]]*{100*[1-TAX_K_RATE]*d_int
      -INTR*TAX_K_RATE*tax_k_rr-100*[1/[1+INF]]*[1+INTR*[1-TAX_K_RATE]]*d_inf
};

Equation E_p3tot_l
# Lagged value of the CPI if initial sol for year t is sol for year t-1 #
  LEV_CPI_L*p3tot_l = 100*[LEV_CPI_B - LEV_CPI_L_B]*d_unity +
100*d_f_p3tot_l;

Equation E_p1lab_io_l
# Lagged value of the CPI if initial sol for year t is sol for year t-1 #
  LEV_PLAB_L*p1lab_io_l = 100*[LEV_PLAB_B - LEV_PLAB_L_B]*d_unity
                          + 100*d_f_p1lab_io_l;

Equation E_d_int # Nominal rate of interest #
  d_int = [1+INF]*d_rint + [1+RINT]*d_inf;

Equation E_d_inf # Rate of inflation #
  100*d_inf =[1+INF]*[p3tot - p3tot_l];

Equation E_d_eeqror
  # Expected ror equals equil. expec. ror plus disequilibrium in expec. ror #
(all,d,DST)(all,i,IND) d_eror(d,i) = d_eeqror(d,i) + d_diseq(d,i);

Equation E_d_diseq
  # Gives shock to disequil. in s.e. rors, moves them towards zero #
(all,d,DST)(all,i,IND) d_diseq(d,i) =
    - ADJ_COEFF(d,i)*DISEQSE_B(d,i)*d_unity + d_f_diseq(d,i);

Equation E_d_f_diseqre
  # Gives shock to disequil. in r.e. rors, moves them towards zero #
(all,d,DST)(all,j,IND) d_diseq(d,j) = - ADJ_COEFF(d,j)*DISEQRE_B(d,j)*d_unity
```

```
                                        + d_f_diseqre(d,j);

Equation E_d_eror # Expected rate of return, static expectations #
(all,d,DST)(all,i,IND) d_eror(d,i) =  d_ror_se(d,i) + d_ff(d,i);

!Equalization of rates of return!

Variable del_r_tot;

Variable (all,i,IND)(all,d,DST)
d_ff_eror(d,i);

Equation E_d_ff_eror # Allows for equalization of changes in rates of return
#
 (all,d,DST)(All,i,IND) d_eror(d,i) = del_r_tot +  d_ff_eror(d,i);

Coefficient QCAPATT_DI;
Formula
QCAPATT_DI=sum(d,DST,sum[i,IND,QCAPATT(d,i)]);

Variable capatt_di;

Equation E_capatt_di
QCAPATT_DI*capatt_di=sum(d,DST,sum[i,IND,QCAPATT(d,i)*x1cap(i,d)]);


Coefficient (all,d,DST) QCAPATT_I(d);
Formula
(all,d,DST) QCAPATT_i(d)=sum[i,IND,QCAPATT(d,i)];

Variable (all,d,DST) capatt_i(d);

Equation E_capatt_i
(all,d,DST) QCAPATT_i(d)*capatt_i(d)=sum[i,IND,QCAPATT(d,i)*x1cap(i,d)];

Equation E_d_eror_ave # Average expected rate of return #
(all,d,DST) d_eror_ave(d)=
        (1/{sum{i,IND,V1CAP(d,i)}+TINY})*sum{i,IND,V1CAP(d,i)*d_eror(d,i)};

Equation E_del_ror_se_o
# Used in 1st rat. expect. policy iteration to introduce forecast s.e. rors #
(all,d,DST)(all,i,IND) del_ror_se_o(d,i) = d_ror_se(d,i) + d_f_ror_se_o(d,i);

Equation E_d_eror_o
```

```
# Used in 1st rat. expect. policy iteration to introduce forecast r.e. rors #
(all,d,DST)(all,i,IND) d_eror_o(d,i) = d_eror(d,i) + d_f_eror_o(d,i);

Equation E_d_f # Expected rate of return, rational expectations #
(all,d,DST)(all,i,IND) d_eror(d,i) =  ONE_ITER1*d_ror_se(d,i)
  + ONE_IT1_REP*COEFF_NYEAR*( d_eror_o(d,i) + d_ror_se(d,i)-
del_ror_se_o(d,i))
+(1-ONE_IT1_REP)*(1-ONE_ITER1)*COEFF_NYEAR*{EROR_F(d,i) -
EROR_B(d,i)}*d_unity
                                + d_f(d,i);

Equation  E_d_f_pi_l
  # Lag value of cap. asset price if init. sol for year t is sol for year t-1
#
(all,d,DST)(all,i,IND)
    PCAP_I_L(d,i)*p2tot_l(d,i) = 100*(PCAP_I_B(d,i) -
PCAP_I_L_B(d,i))*d_unity
                                      + 100*d_f_p2tot_l(d,i);

Equation E_d_int_l # Lagged nominal interest rate #
  d_int_l =  (1+RINT_L)*d_inf_l +(1+INF_L)*d_rint_l;

Equation E_d_inf_l # Lagged rate of inflation #
  100*d_inf_l =(1+INF_L)*(p3tot_l - p3tot_2l);

Equation E_d_f_rint_l
  # Lagged real interest rate, if init. sol for year t is sol for year t-1 #
  d_rint_l =(RINT_B - RINT_L_B)*d_unity + d_f_rint_l;

 Equation  E_d_f_p3tot_2l
  # Double lagged value of the CPI if init. sol for year t is sol for year t-
1 #
  LEV_CPI_2L*p3tot_2l = 100*(LEV_CPI_L_B - LEV_CPI_2L_B)*d_unity
                                      + 100*d_f_p3tot_2l;

Equation E_lev_eror # Level of the expected ror in year t #
(all,d,DST)(all,i,IND) lev_eror(d,i) = EROR_B(d,i)*d_unity + d_eror(d,i);

Equation E_lev_eror_l # Level of the expected ror in year t-1 #
(all,d,DST)(all,i,IND) lev_eror_l(d,i) = EROR_B(d,i)*d_unity;

Equation E_lev_ror_act_l # Actual ror in year t-1 #
(all,d,DST)(all,i,IND)
    lev_ror_act_l(d,i) = ROR_ACT_L_B(d,i)*d_unity +d_ror_act_l(d,i);
```

```
Equation E_d_ror_act_l # Actual rate of return for year t-1 #
(all,d,DST)(all,j,IND) 100*d_ror_act_l(d,j) =
        (1/{[1 + INTR_L*(1-TAX_K_RATE)]*PCAP_I_L(d,j)})*
        {  - (V1CAP(d,j)/QCAPATT(d,j))*TAX_K_RATE*tax_k_rr
      + (1-TAX_K_RATE)*(V1CAP(d,j)/QCAPATT(d,j))*(p1cap(j,d) - p2tot_l(d,j))
      + (1-DEP(d,j))*PCAP_I(d,j)*(p2tot(d,j) - p2tot_l(d,j))
  + RALPH*TAX_K_RATE*DEP(d,j)*PCAP_I(d,j)*(tax_k_rr + p2tot(d,j)-
p2tot_l(d,j))}
       - {(1+ROR_ACT_L(d,j))/[1 + INTR_L*(1-TAX_K_RATE)]}*
               {    100*(1-TAX_K_RATE)*d_int_l
                  - INTR_L*TAX_K_RATE*tax_k_rr                     };


!****************************************************************************!
! section 8.17. Labour markets !
!****************************************************************************!

! definition of real wages !

Equation
E_p1lab (all,i,IND)(all,o,OCC)(all,d,DST)
  realwage(i,o,d) = p1lab(i,o,d) - pfin("Hou",d);

!####### sum-over-industry labour aggregates ##########!

Equation
E_p1lab_i (all,o,OCC)(all,d,DST)
  p1lab_i(o,d) = sum{i,IND, SLAB_I(i,o,d)*p1lab(i,o,d)};

E_realwage_i (all,o,OCC)(all,d,DST)
  realwage_i(o,d) = sum{i,IND, SLAB_I(i,o,d)*realwage(i,o,d)};

E_x1lab_i (all,o,OCC)(all,d,DST)
  x1lab_i(o,d) = sum{i,IND, SLAB_I(i,o,d)*x1lab(i,o,d)};

E_wlab_i  (all,o,OCC)(all,d,DST)
  wlab_i(o,d) = x1lab_i(o,d) + p1lab_i(o,d);

E_rlab_i  (all,o,OCC)(all,d,DST)
  rlab_i(o,d) = x1lab_i(o,d) + realwage_i(o,d);
```

```
!####### sum-over-industry-and-region labour aggregates ##########!
Equation
E_p1lab_id (all,o,OCC)
  p1lab_id(o) = sum{d,DST, SLAB_ID(o,d)*p1lab_i(o,d)};

E_realwage_id (all,o,OCC)
  realwage_id(o) = sum{d,DST, SLAB_ID(o,d)*realwage_i(o,d)};

E_x1lab_id (all,o,OCC)
  x1lab_id(o) = sum{d,DST, SLAB_ID(o,d)*x1lab_i(o,d)};

E_wlab_id  (all,o,OCC)
  wlab_id(o) = x1lab_id(o) + p1lab_id(o);

E_rlab_id  (all,o,OCC)
  rlab_id(o) = x1lab_id(o) + realwage_id(o);

!####### sum-over-occupation-and-industry labour aggregates ##########!
Equation
E_p1lab_io  (all,d,DST)
  LAB_IO(d)*p1lab_io(d) = sum{o,OCC,LAB_I(o,d)*p1lab_i(o,d)};
E_x1lab_io  (all,d,DST)
  LAB_IO(d)*x1lab_io(d) = sum{o,OCC,LAB_I(o,d)*x1lab_i(o,d)};
E_realwage_io  (all,d,DST)
  LAB_IO(d)*realwage_io(d) = sum{o,OCC,LAB_I(o,d)*realwage_i(o,d)};
E_wlab_io  (all,d,DST)
  wlab_io(d) = x1lab_io(d) + p1lab_io(d);
E_rlab_io  (all,d,DST)
  rlab_io(d) = x1lab_io(d) + realwage_io(d);


!############# wage dynamics ##########################################!

Equation E_del_f_wage_c
# Relates deviation in CPI-deflated pre-tax wage to deviation in employment #
(RWAGE/RWAGE_OLD)*(real_wage_iod - real_wage_c_o)
    =    100*((RWAGE_B/RWAGE_OLD_B)
    -   (RWAGE_L_B/RWAGE_O_L_B))*d_unity
    +    ALPHA1*(EMPLOY/EMPLOY_OLD)*(employ_i - employ_i_o)
    -    100*ALPHA1*((RWAGE_B/RWAGE_OLD_B)^ALPHA2
    -   (RWAGE_L_B/RWAGE_O_L_B)^ALPHA2)*d_unity
    +    del_f_wage_c;
```

```
Equation E_del_f_wage_pt
# Relates deviation in CPI-deflated post-tax wage to deviat. in employment #
(RWAGE_PT/RWAGE_PT_OLD)*(real_wage_pt - real_wage_pt_o)
    =    100 * ((RWAGE_PT_B/RWAGE_PT_O_B)
    -   (RWAGE_PT_L_B/RW_PT_O_L_B))*d_unity
    +    ALPHA1*(EMPLOY/EMPLOY_OLD)*(employ_i - employ_i_o)
    -    100*ALPHA1*((RWAGE_PT_B/RWAGE_PT_O_B)^ALPHA2
    -   (RWAGE_PT_L_B/RW_PT_O_L_B)^ALPHA2)*d_unity
    +    del_f_wage_pt;

Equation E_real_wage_pt # Economy-wide CPI-deflated wage rate, post tax #
real_wage_pt
    = real_wage_iod
    - TAX_L_RATE/(1 - TAX_L_RATE)*tax_l_rds;

Equation E_d_f_empadj
  # Direct adjustment of employment back to basecase forecast #
(EMPLOY/EMPLOY_OLD)*(employ_i - employ_i_o)
    =    100*{(EMPLOY_B/EMPLOY_O_B)
    -   (EMPLOY_L_B/EMPLOY_O_L_B)}*d_empadj
    +    d_f_empadj;

Equation E_d_ff_empadj
  # Equation for moving level of shift variable in E_d_f_empadj back to zero
#
d_f_empadj
    =   {-FEMPADJ_B+ FEMPADJ_O}*d_emp_sh
    +    d_ff_empadj;

Equation E_real_wage_c # Economy-wide real wage rate for consumers #
[AGGLAB-COL_PAYRTOT]*[real_wage_c + p3tot]
    =   sum{d,DST, sum{i,IND,[LAB_O(i,d)/POW_PAYROLL(i,d)]*p1lab_o(i,d)}};

Equation E_real_wage_c_o
  # Introduces forecast CPI-deflated pre-tax wage into policy simulation #
  real_wage_c_o = realwage_iod + f_rwage_o;

Equation E_employ_i_o # Introduces forecast employment into policy simulation
#
  employ_i_o  = employ_i + f_emp_o;

Equation E_real_wage_pt_o
  # Forecast post-tax CPI-deflated wage used in policy simulations #
  real_wage_pt_o = real_wage_pt + f_rwage_pt_o;
```

```
Equation E_tax_l_rds_o # Forecast tax rate on wages used in policy
simulations #
  tax_l_rds_o = tax_l_rds + ftax_l_rds_o;

Equation E_wlabnat
sum{d,DST,LAB_IO(d)}*wlabnat = sum{d,DST,LAB_IO(d)*wlab_io(d)};

Equation E_tax_l_rds # Forecast tax rate on wages used in policy simulations
#
sum(d,DST,sum(s,PSEC,TAXS_LAB(d,s)))*(wlabnat+tax_l_rds)
= sum(d,DST,sum(s,PSEC,TAXS_LAB(d,s)*(wlab_io(d)+tax_l_r(d,s))));

Equation E_f_tax_l_rd # Forecast tax rate on wages used in policy simulations
#
(all,d,DST)(all,s,PSEC)
tax_l_r(d,s)=f_tax_l_r(d,s)+f_tax_l_rd(s);

Equation E_tax_k_r
# Rate of tax on capital income related to rate of tax on labour income #
(all,d,DST)(all,s,PSEC)
  tax_k_r(d,s) = tax_l_r(d,s) + f_tax_k_r(d,s);

Equation E_tax_l_ub
# Rate of tax on capital income related to rate of tax on labour income #
(all,d,DST)(all,s,PSEC)
  tax_ub_r(d,s) = tax_l_r(d,s) + f_tax_ub_r(d,s);

Equation E_tax_l_ab
# Rate of tax on capital income related to rate of tax on labour income #
(all,d,DST)(all,s,PSEC)
  tax_ab_r(d,s) = tax_l_r(d,s) + f_tax_ab_r(d,s);

Equation E_tax_l_ob
# Rate of tax on capital income related to rate of tax on labour income #
(all,d,DST)(all,s,PSEC)
  tax_ob_r(d,s) = tax_l_r(d,s) + f_tax_ob_r(d,s);

Equation E_employed # Industry level employment #
(all,i,IND) ID01[LAB_OD(i)]*employed(i) = sum{d,DST,LAB_O(i,d)*x1lab_o(i,d)};

Equation E_employ_i # Employment at national level #
sum(d,DST,LAB_IO(d))*employ_i = sum{d,DST,LAB_IO(d)*x1lab_io(d)};
```

```
Equation E_f_labsup # regional labour supply  #
(all,d,DST)  labsup(d) = x1lab_io(d)+(100/EMP_RATE(d))*d_UR(d)
+f_labsup(d);

Equation E_f_labsupn # National labour supply #
  LAB_SUPN*labsupn = sum(d,DST,LAB_SUP(d)*labsup(d))+f_labsupn;

Equation E_d_URN # National labour supply #
labsupn=employ_i+[100/EMP_RATEN]*d_URN;

Equation E_labsup_o # Forecast labour supply used in policy simulations #
  labsup_o = labsupn + f_labsup_o;

Equation E_flab   # Flexible setting of money wages #
(all,i,IND)(all,o,OCC)(all,d,DST)
p1lab(i,o,d)=
p3tot+flab(i,o,d)+flab_id(o)+flab_io(d)+flab_o(i,d)+flab_iod+flab_od(i);


!#############################################################################!

!           Section 9 Reporting variables                                     !
!#############################################################################!
!*****************************************************************************!
! Section 9.1.                                                                !
! Variables for reporting purposes: attack problem of % change in nothing     !
! Below variables MUST be endogenous and should be backsolved                 !
!*****************************************************************************!


Coefficient (parameter) SIGNIF # Threshold reporting value #;
Formula     (initial)   SIGNIF = 1;

Variable (all,c,COM)(all,d,DST) xexpSHO(c,d)
   # Export demands for domestic all-region composite leaving port at d #;
Equation E_xexpSHO (all,c,COM)(all,d,DST)
  xexpSHO(c,d) = if(USE(c,"dom","exp",d)>SIGNIF, xexp(c,"dom",d));

Variable (all,i,IND)(all,d,DST) x1capSHO(i,d) # Capital usage #;
Equation E_x1capSHO (all,i,IND)(all,d,DST)
  x1capSHO(i,d) = if(CAP(i,d)>SIGNIF, x1cap(i,d));

Variable (all,i,IND)(all,d,DST) pcapSHO(i,d) # Rental price of capital #;
Equation E_pcapSHO (all,i,IND)(all,d,DST)
  pcapSHO(i,d) = if(CAP(i,d)>SIGNIF, p1cap(i,d));
```

```
Variable (all,i,IND)(all,d,DST) x1lab_oSHO(i,d) # Effective labour input #;
Equation E_x1lab_oSHO (all,i,IND)(all,d,DST)
  x1lab_oSHO(i,d) = if(LAB_O(i,d)>SIGNIF, x1lab_o(i,d));

Variable (all,i,IND)(all,d,DST) x2totSHO(i,d) # Investment by industry #;
Equation E_x2totSHO (all,i,IND)(all,d,DST)
  x2totSHO(i,d) = if(INVEST_C(i,d)>SIGNIF, x2tot(i,d));


!***************************************************************************!
! Section 9.2.                                                              !
! Sectoral contributions to regional GDP at factor cost                     !
!***************************************************************************!

Equation
 E_contxprim_i (all,i,IND)(all,d,DST)
   contxprim_i(i,d) = RATIOPRIM_I(d)*PRIMSHR(i,d)*x1tot(i,d);

 E_xprim_i (all,d,DST) xprim_i(d) = sum{i,IND, PRIMSHR(i,d)*x1tot(i,d)};
!above could also be: RATIOPRIM_I(d)*xPRIM_I(d)=sum{i,IND,contxPRIM_I(i,d)}!

! Contributions of endowments, tech change and tax to national real GDP !
! Keller decomposition
These are matrices (best viewed by ViewHAR from the SOL file) which add
up to the percent change in real GDP
!
Set CONTINC # Contributors to real income gdp change #
(Land, Labour, Capital,                             ! factors !
 a1lnd, a1cap, a1lab_o, aprim, atot, a1int_s, atradmar,   ! tech change !
 PRODTAX, ComTax);                                  ! indirect taxes !

Coefficient NATGDPINC # National income-side GDP #;
            RATIOXGDP # (Current/initial) ratio, national real gdp #;
Formula     NATGDPINC = sum{d,DST, GDPINC(d)};
  (initial) RATIOXGDP = 1;
Update      RATIOXGDP = NatMacro("RealGDP");

Variable
 (change)(all,i,IND)(all,d,DST)(all,c,CONTINC)
  contincind(i,d,c) # Industry contribution terms to national real income gdp
#;
 (change)(all,i,IND)(all,c,CONTINC)
  contincind_d(i,c) # Industry contribution terms to national real income gdp
```

```
#;
Equation E_contincind
 (all,i,IND)(all,d,DST)(all,c,CONTINC) contincind(i,d,c) =
[RATIOXGDP/NATGDPINC]*{
  if[c="Land",    LND(i,d)*x1lnd(i,d)]
+ if[c="Labour",  sum{o,OCC, LAB(i,o,d)*x1lab(i,o,d)}]
+ if[c="Capital", CAP(i,d)*x1cap(i,d)]
- if[c="a1lnd",    LND(i,d)*a1lnd(i,d)]
- if[c="Capital", CAP(i,d)*a1cap(i,d)]
- if[c="a1lab_o",  LAB_O(i,d)*(a1lab_o(i,d)+a1lab_od(i))]
- if[c="aprim",   PRIM(i,d)*a1prim(i,d)]
- if[c="atot",    VCST(i,d)*a1tot(i,d)]
- if[c="a1int_s",  sum{k,COM, PUR_S(k,i,d)*a1int_s(k,i,d)}]
+ if[c="prodtax", PRODTAX(i,d)*x1tot(i,d)]};
Equation E_contincind_d
 (all,i,IND)(all,c, CONTINC) contincind_d(i,c) =sum{d,DST,
contincind(i,d,c)};

Variable
(change) (all,c,COM)(all,s,SRC)(all,d,DST)
   continccom(c,s,d) # COMTAX contribution terms to national real income gdp
#;
Equation E_continccom
(all,c,COM)(all,s,SRC)(all,d,DST)  continccom(c,s,d) =
[RATIOXGDP/NATGDPINC]*{
  sum{i,IND, TAX(c,s,i,d)*x1(c,s,i,d)}
          + TAX(c,s,"hou",d)*x3(c,s,d)
          + TAX(c,s,"inv",d)*x2(c,s,d)
          + TAX(c,s,"gov",d)*xgov(c,s,d)
          + TAX(c,s,"exp",d)*xexp(c,s,d)};

Variable
(change) (all,d,DST)(all,c,CONTINC) contincagg(d,c)
    # Combined contribution terms to national real income gdp #;
Equation E_contincagg
 (all,d,DST)(all,c,CONTINC) contincagg(d,c) = sum{i,IND, contincind(i,d,c)}
+ if[c="comtax", sum{cc,COM,sum{s,SRC, continccom(cc,s,d)}}]
- if[c="atradmar",  sum{k,COM,sum{s,SRC,sum{m,MAR,sum{r,ORG,
        TRADMAR(k,s,m,r,d)*atradmar(k,s,m,r,d)}}}}];

Variable
(change) (all,c,CONTINC) contincagg_d(c)
    # Contribution terms to national real income gdp #;
Equation E_contincagg_d
```

```
 (all,c,CONTINC) contincagg_d(c) = sum{d,DST, contincagg(d,c)};

! COM and REG contributions to national export and import price indices !
Set TRADDIR (EXP, imp);
Coefficient (all,t,TRADDIR) RatioTOT(t) # Terms of Trade #;
Formula (initial)(all,t,TRADDIR) RatioTOT(t) = 1;
Update
 RatioTOT("exp") = NatMacro("ExportPI");
 RatioTOT("imp") = NatMacro("ImpsLandedPI");
Variable (change)(all,c,COM)(all,q,REG)(all,t,TRADDIR) contTrmsTrad(c,q,t)
   # Contributions to national export and import price indices #;

Equation E_contTrmsTradA
(all,c,COM)(all,q,REG) WNATMACRO("ExpVol")*contTrmsTrad(c,q,"exp") =
 RatioTOT("Exp")*PUR_S(c,"exp",q)*pfin("Exp",q);

Equation E_contTrmsTradB
(all,c,COM)(all,q,REG) WNATMACRO("ImpVolUsed")*contTrmsTrad(c,q,"imp") =
 RatioTOT("Imp")*TRADE_D(c,"imp",q)*pimp(c,q);


!***********************************************************************!
! Section 9.3.                                                          !
! Regional macro reporting variables                                    !
!***********************************************************************!


Equation
 E_MainMacroA (all,q,REG) MainMacro("RealHou",q) = xfin("Hou",q);
 E_MainMacroB (all,q,REG) MainMacro("RealInv",q) = xfin("Inv",q);
 E_MainMacroC (all,q,REG) MainMacro("RealGov",q) = xfin("Gov",q);
 E_MainMacroD (all,q,REG) MainMacro("ExpVol",q) = xfin("Exp",q);
 E_MainMacroE (all,q,REG) MainMacro("ImpVolUsed",q) = ximpused(q);
 E_MainMacroF (all,q,REG) MainMacro("ImpsLanded",q) = ximplanded(q);
 E_MainMacroG (all,q,REG) MainMacro("RealGDP",q) = xgdpexp(q);
 E_MainMacroH (all,q,REG) MainMacro("AggEmploy",q) = x1lab_io(q);
 E_MainMacroI (all,q,REG) MainMacro("realwage_io",q) = realwage_io(q);
 E_MainMacroJ (all,q,REG) MainMacro("p1lab_io",q) = p1lab_io(q);
 E_MainMacroK (all,q,REG) MainMacro("AggCapStock",q) = x1cap_i(q);
 E_MainMacroL (all,q,REG) MainMacro("CPI",q) = pfin("Hou",q);
 E_MainMacroM (all,q,REG) MainMacro("GDPPI",q) = pgdpexp(q);
 E_MainMacroN (all,q,REG) MainMacro("ExportPI",q) = pfin("Exp",q);
 E_MainMacroO (all,q,REG) MainMacro("ImpsLandedPI",q) = pimplanded(q);
 E_MainMacroP (all,q,REG) MainMacro("Population",q) = nhou(q);
```

```
 E_MainMacroQ (all,q,REG) MainMacro("NomHou",q) = wfin("Hou",q);
 E_MainMacroR (all,q,REG) MainMacro("NomGDP",q) = wgdpexp(q);

! Excerpt 29 of TABLO input file: !
! Aggregation of regional macro variables to national macro variables !



Equation E_NatMacro (all,m,MAINMACROS)
   WNATMACRO(m)*NatMacro(m) =  sum{q,REG,WMAINMACRO(m,q)*MainMacro(m,q)};
Equation E_contMainMacro (all,m,MAINMACROS)(all,q,REG)
   WNATMACRO(m)*contMainMacro(m,q)
      =  RATIOMMACRO(m)*WMAINMACRO(m,q)*MainMacro(m,q);

Variable (change) shrBoT # National real balance of trade as % of real GDP #;
! S=100B/G -> SG=100B -> SdG+GdS=100dB -> SGg/100+GdS=100dB -> Bg+GdS=100dB !
Equation E_shrBoT
[WNATMACRO("ExpVol")-WNATMACRO("ImpsLanded")]*NatMacro("RealGDP")
+ WNATMACRO("RealGDP")*shrBoT = WNATMACRO("ExpVol")*NatMacro("ExpVol")
                                   -
WNATMACRO("ImpsLanded")*NatMacro("ImpsLanded");

Variable ! shrBoTnom may be chosen as exogenous for trade balance constra1int
!
 (change) shrBoTnom # National nominal balance of trade as % of nominal GDP
#;
! S=100B/G -> SG=100B -> SdG+GdS=100dB -> SGg/100+GdS=100dB -> Bg+GdS=100dB !
Equation E_shrBoTnom
[WNATMACRO("ExpVol")-WNATMACRO("ImpsLanded")]*NatMacro("NomGDP")
+ WNATMACRO("RealGDP")*shrBoTnom =
  WNATMACRO("ExpVol")*[NatMacro("ExpVol")+NatMacro("ExportPI")]
- WNATMACRO("ImpsLanded")*[NatMacro("ImpsLanded")+NatMacro("ImpsLandedPI")];

! Commodity contributions to selected national results !

Set COMMACROS # MAINMACROS with commodity components #
 (RealHou,  RealInv, RealGov,ExpVol,
ImpsLanded, CPI,ExportPI, ImpsLandedPI);
Subset COMMACROS is subset of MAINMACROS;
Variable
(change)(all,c,COM)(all,m,COMMACROS)
   contComMacro(c,m) # Commodity contributions to national macro results #;

Equation
```

```
 E_contComMacroA
 (all,c,COM) WNATMACRO("RealHou")*contComMacro(c,"RealHou")
   = RATIOMMACRO("RealHou")*sum{d,DST, PUR_S(c,"Hou",d)*x3_sh(c,d)};
 E_contComMacroB
 (all,c,COM) WNATMACRO("RealInv")*contComMacro(c,"RealInv")
   = RATIOMMACRO("RealInv")*sum{d,DST, PUR_S(c,"Inv",d)*x2_s(c,d)};
 E_contComMacroC
 (all,c,COM) WNATMACRO("RealGov")*contComMacro(c,"RealGov")
  = RATIOMMACRO("RealGov")*sum{d,DST, PUR_S(c,"Gov",d)*xgov_s(c,d)};
 != RATIOMMACRO("RealGov")*sum{d,DST,sum{s,SRC,
PUR(c,s,"Gov",d)*xgov(c,s,d)}};!
 E_contComMacroD
 (all,c,COM) WNATMACRO("ExpVol")*contComMacro(c,"ExpVol")
  = RATIOMMACRO("ExpVol")*sum{d,DST, PUR_S(c,"Exp",d)*xexp_s(c,d)};
 != RATIOMMACRO("ExpVol")*sum{d,DST,sum{s,SRC, PUR
(c,s,"Exp",d)*xexp(c,s,d)}};!
 E_contComMacroE
 (all,c,COM) WNATMACRO("ImpsLanded")*contComMacro(c,"ImpsLanded")
   =
RATIOMMACRO("ImpsLanded")*sum{r,ORG,TRADE_D(c,"imp",r)*xtrad_d(c,"imp",r)};
 E_contComMacroF
 (all,c,COM) WNATMACRO("CPI")*contComMacro(c,"CPI")
   = RATIOMMACRO("CPI")*sum{d,DST, PUR_S(c,"Hou",d)*ppur_s(c,"Hou",d)};
 E_contComMacroG
 (all,c,COM) WNATMACRO("ExportPI")*contComMacro(c,"ExportPI")
   = RATIOMMACRO("ExportPI")*sum{d,DST, PUR_S(c,"Exp",d)*ppur_s(c,"Exp",d)};
 E_contComMacroH
 (all,c,COM) WNATMACRO("ImpsLandedPI")*contComMacro(c,"ImpsLandedPI")
   = RATIOMMACRO("ImpsLandedPI")*sum{r,ORG, TRADE_D(c,"imp",r)*pimp(c,r)};

Variable ! equations to help check above !
 (change) (all,m,COMMACROS) NatComMacrox(m) # Sum of contComMacro(c,m) #;
          (all,m,COMMACROS) NatComMacro(m)  # Check:should = NatComMacrox(m)
#;
Equation
E_NatComMacro  (all,m,COMMACROS) NatComMacro(m) = NatMacro(m);
E_NatComMacrox (all,m,COMMACROS) NatComMacrox(m) =
sum{c,COM,contComMacro(c,m)};
```

```
!***************************************************************************!
! Section 9.4.                                                              !
! Weights for variable aggregation of results                               !
!***************************************************************************!

Coefficient
 (all,c,COM)(all,q,REG) Import(c,q);
 (all,c,COM)(all,q,REG) Export(c,q);
Formula
 (all,c,COM)(all,q,REG) Import(c,q) = TRADE_D(c,"imp",q);
 (all,c,COM)(all,q,REG) Export(c,q) = PUR_S(c,"exp",q);
Write
Import to file SUMMARY header "IMP" longname "Internat regional imports";
HOUSHR to file SUMMARY header "HOUP";
INVEST_I to file SUMMARY header "VINV" longname "Investment costs";
LAB_O to file SUMMARY header "LAB" longname "Labour costs";
PRIM to file SUMMARY header "PRIM" longname "Primary factor payments";
VARCST to file SUMMARY header "VCST" longname "Variable costs";
CAP to file SUMMARY header "CAP" longname "Capital rentals";
Export to file SUMMARY header "VEXP" longname "Exports";
NATVTOT to file SUMMARY header "NATV" longname "National output";
HOUPUR_H to file SUMMARY header "HOUS";
Set WAGG
(xtot@@VTOT,
ptot@@VTOT,
x1cap@@CAP,
pcap@@CAP,
pimp@@IMP,
xhou@@HOUP,
xinv@@VINV,
p1lab@@LAB,
x1lab@@LAB,
aprm@@PRIM,
pvar@@VCST,
xexp@@VEXP,
ximp@@IMP,
nvto@@@@NATV,
munx@@@@MVTO,
xhos@@@@HOUS);
Write (Set) WAGG to file SUMMARY header "WAGG";
```

5. The theory of VERM

5.1 Introduction

In this chapter, we present a formal description of the linear form of VERM. The presentation is based on excerpts taken from the TABLO code.

The chapter is organised as follows. In section 5.2, we describe the AGE theory behind the demand for intermediate goods and primary factors, household and government demand, export demand and margin demand. Section 5.3 is devoted to the dynamic mechanisms of VERM. Section 5.3.1 explains the theory of investment over time, section 5.3.2 asset dynamics, and section 5.2.3 deals with the sluggish wage mechanism. The final section of the chapter deals with income and saving aggregates, the balance of payments, as well as government accounts, and shows how budgetary rules can be introduced in the model.

5.2 The AGE core of VERM

VERM recognises three broad categories of inputs: intermediate inputs, primary factors and energy . Industries are assumed to choose the mix of inputs which minimises the costs of production for their level of output. They are constrained in their choice of inputs by a three-level nested production technology, displayed in figure 2.2. At the first level, the intermediate-input bundles and the primary-factor bundles are used in fixed proportions to output. These bundles are formed at the second level. Intermediate input bundles are constant-elasticity-of-substitution (CES) combinations of international imported goods and domestic goods. The primary-factor-energy bundle is a CES combination of the primary factors labour, capital and land, which combines with a CES bundle of energy inputs electricity, fossil fuels, and in some industries, bioenergy. Finally, the input of labour is formed as a CES combination of inputs of labour from different occupational categories. We describe the derivation of the input demand functions working upwards from the bottom of the tree in Figure 2.2.

5.2.1 Import-domestic composition of intermediate demands

(Section 8.1.1 in VERM code)

At the bottom level of the intermediate good nest are the demands for commodities from various sources. The firms decide on their demands for the domestic commodities and the foreign imported commodities following a similar to the CES nest for occupational groups. Here, the firm chooses a cost-minimising mix of the domestic commodity and foreign imported commodities

$$\sum_{s \in SRC} P1X1(c,s,i)X1(c,s,i) \qquad i \in \text{IND} \tag{5.1}$$

where the subscript SRC denotes the sources of the commodities, *dom*, *eu*, and *non-eu*, subject to the production function

$$X1\_S(c,i) = CES\{\frac{X1(c,s,i)}{A1(c,s,i)}\} \qquad c \in \text{COM}, \ s \in \text{SRC}, \ i \in \text{IND} \tag{5.2}$$

where we denote a constant elasticity to scale aggregator by CES( ), with its arguments inside the brackets. The CES assumption amounts to the standard Armington assumption that domestic commodities are imperfect substitutes for foreign varieties.

The solution to the problem specified by (5.1) and (5.2) yields the input demand functions for the domestic and import commodities. In VERM code, these functions are presented by equation E_x1:

```
E_x1 #Dom/imp substitution in intermediate demands#
(all,c,COM)(all,s,SRC)(all,i,IND)(all,d,DST)
x1(c,s,i,d) = x1_s(c,i,d) - SIGMADOMIMP(c)*[ppur(c,s,i,d)-
ppur_s(c,i,d)];                                             (5.3)
```

This equation shows that the demands for the domestic commodities and imports are proportional to the demand for the domestic-import aggregate (X1_S(c,i,d)) and to a price term. The X1_S(c,i,d) are exogenous to the producer's problem at this level of the nest. The percentage-change form of the price term is an elasticity of substitution, SIGMADOMIMP(c) multiplied by a price ratio representing the percentage change in the price of the domestic commodity or of the import commodities relative to the purchaser's price of the domestic-import aggregate (ppur_s(c,j,d).

The price index ppur_s(c,i,d) in equation (5.3) is defined in equation E_ppur_s in section 8.7 of the code:

```
Equation E_ppur_s
(all,c,COM)(all,u,USR)(all,d,DST)
         ppur_s(c,u,d) =
sum{s,SRC,SRCSHR(c,s,u,d)*ppur(c,s,u,d)};                   (5.4)
```

which is a Divisia index of the individual prices.

5.2.2 Demand for primary factors

(Section 8.1.2. in VERM code)

The demand for primary factors consists of several nests. At the lowest-level nest in the primary-factor nest, producers choose a composition of o occupation-specific labour inputs to minimise the costs of a given composite labour aggregate input. The demand equations for labour of the various occupation types are derived from the following optimisation problem, where industry i's problem is to choose inputs of occupation-specific labour type m, X1LABO(i,o), to minimise total labour cost

$$\sum_{o \in occ} P1LAB(i,o)X1LAB(i,o) \tag{5.5}$$

subject to

$$X1LAB\_O(i) = \underset{o \in occ}{CES}\{X1LAB(i,o)\} \tag{5.6}$$

Exogenous to this problem are the price paid by the *i*th industry for each occupation-specific labour type (P1LAB(i,o)) and the industries' demands for the effective labour input (X1LAB_O(i)). The solution to this problem, in percentage-change form, is given by equations *E_x1lab* and *E_p1lab* in VERM code:

```
Equation E_x1lab   # Demand for labour by industry and skill
group #
  (all,i,IND)(all,o,OCC)(all,d,DST)
   x1lab(i,o,d) = x1lab_o(i,d) - SIGMa1lab(i)*[p1lab(i,o,d) -
p1lab_o(i,d)]
    +f_x1lab(i,o,d);                                        (5.7)

Equation E_p1lab_o # Price to each industry of labour
composite #
  (all,i,IND)(all,d,DST)
 ID01(LAB_O(i,d))*p1lab_o(i,d) = sum{o,OCC,
LAB(i,o,d)*p1lab(i,o,d)};                                   (5.8)
```

Equation (5.7.), or *E_x1lab* in the VERM code, indicates that the demand for labour type *o* in each industry and each region is proportional to the demand for the effective composite labour demand and to a price term. The price term consists of an elasticity of substitution, SIGMA1LAB(i,o), multiplied by the percentage change in a price ratio representing the wage of occupation o (p1lab(i,o)) relative to the average wage for labour in industry i (p1lab_o(i))). Changes in the relative wages of the occupations induce substitution in favour of relatively cheapening occupations. In equation *E_p1lab*, the coefficient V1LAB(i,o) is the wage bill for occupation o employed by industry i, whereas the coefficient V1LAB_O(i) is the total wage bill of industry i. Thus, p1lab(j,q) is a Divisia index of the p1lab(i,o).

Summing the percentage changes in occupation-specific labour demands across occupations, using appropriate occupational shares, for each industry gives the percentage change in industry labour demand.

At the next level of the primary-factor branch of the production nest, we determine the composition of demand for primary factors. Primary factors and energy are assumed to be combined with energy to form a KLE-nest. Their derivation follows the same CES pattern as the previous nests. Here, total primary factor costs for industry i are given by

$$P1LAB\_O(i)X1LAB\_O(i)+P1CAP(i)X1CAP(i)+P1LND(i)X1LND(i) \qquad (5.9)$$

where P1CAP(i) and P1LND(i) are the unit costs of capital and land (in agriculture and forestry) and X1CAP(i) are the industry's demands for capital. These costs are minimised subject to the production function

$$X1PRIM(i)=CES\left[\frac{X1LAB(i)}{A1LAB(i)},\frac{X1CAP(i)}{A1CAP(i)},\frac{X1LND(i)}{A1LND(i)}\right] \qquad (5.10)$$

where X1PRIM(i) is the industry's overall demand for primary factors. The above production function allows us to impose factor-specific technological change via the variables A1LAB(i), A1CAP(i) and A1LND(i).

The solution to the problem in percentage-change form is given by equations *E_x1lab_o*, *E_x1cap*, *E_p1lnd* and *E_x1lnd* in VERM code:

```
Equation E_x1lab_o  # Industry demands for effective labour #
  (all,i,IND)(all,d,DST) x1lab_o(i,d) - a1lab_o(i,d)-
a1lab_od(i) =
x1prim(i,d)-SIGMAPRIM(i)*[p1lab_o(i,d) +
a1lab_o(i,d)+a1lab_od(i) - p1prim(i,d)]
-SIGMAPRIM(i)*(a1lab_o(i,d)+a1lab_od(i)-a1primsum(i,d))
           +SHRLK(i,d,"capital")*twistlk(i);                (5.11)
```

```
Equation E_p1cap  # Industry demands for capital #
  (all,i,IND)(all,d,DST)  x1cap(i,d) - a1cap(i,d) =
   x1prim(i,d) - SIGMAPRIM(i)*[p1cap(i,d) + a1cap(i,d) -
p1prim(i,d)]
-SIGMAPRIM(i)*(a1cap(i,d)-a1primsum(i,d))
            -SHRLK(i,d,"Labour")*twistlk(i);
```
(5.12)

```
Equation E_plnd  # Industry demands for land #
  (all,i,IND)(all,d,DST)  x1lnd(i,d) - a1lnd(i,d) =
   x1prim(i,d)
       - SIGMAPRIM(i)*[plnd(i,d) + a1lnd(i,d) - p1prim(i,d)]
       - SIGMAPRIM(i)*(a1lnd(i,d)-a1primsum(i,d));
```
(5.13)

From these equations, we see that for a given level of technical change, industries' factor demands are proportional to overall factor demand (X1PRIM(i)) and a relative price term. In change form, the price term is an elasticity of substitution (SIGMAPRIM(i,d)) multiplied by the percentage change in a price ratio representing the unit cost of the factor relative to the overall effective cost of primary factor inputs. Changes in the relative prices of the primary factors induce substitution in favour of relatively cheapening factors.

5.2.3 Demand for energy carriers

(Section 8.1.3. in VERM code)

The demand for the energy composite is derived from the minimisation of the cost of the primary-factor-energy composite. This, in turn, is a given for the optimisation problem yielding the demands for individual energy carriers. These demands are given by E_x1_s_f in VERM code:

```
Equation E_x1_s_f  # Demands for fuels composites #
 (All,c,FUEL)(all,i,IND)(all,d,DST)
x1_s(c,i,d) - [a1int_s(c,i,d)] = x1fuel(i,d)
  -  SIGMAFUELS(i)*{ppur_s(c,i,d)-p1fuel(i,d)}
  -  SIGMAFUELS(i)*{a1int_s(c,i,d)-a1_sfuel(i,d)};
```
(5.14)

In the equation, x1fuel is the demand for the energy composite, which is derived in the next section. While energy carriers are included in the primary factor energy nest, the demand for source-specific energy carriers is derived similarly to other commodities. Thus,the demand for source-specific energy goods is given by equation E_x1 (5.3).

5.2.4 Demand for primary factor and energy composites

(Subsection 8.1.4. in VERM code)

The demand for the energy composite and the primary factor composite is derived from the cost-minimisation problem

$$P1PRIM(i)X1PRIM(i) + P1FUEL(i)X1FUEL(i)$$

subject to the production function

$$X1PRIM\_F(i) = CES\left[\frac{X1PRIM(i)}{A1PRIM(i)}, \frac{X1FUEL(i)}{A1FUEL(i)},\right] \qquad (5.15)$$

where P1PRIM(i) and P1FUEL(i) are the unit costs of primary factors and energy, and X1FUEL(i) is the industry's demand for energy. In percentage-change form, the solution yields the demand functions for primary factor and energy composites, given by equations E_x1prim and E_x1fuel:

```
Equation E_x1prim  # Demands for primary factor composite #
 (all,i,IND)(all,d,DST)  x1prim(i,d) - [a1prim(i,d) ] =
x1prim_f(i,d)
-  SIGMAGREEN(i)*(p1prim(i,d)-p1prim_f(i,d))
-  SIGMAGREEN(i)*(a1prim(i,d)-a1prim_fsum(i,d));             (5.16)

Equation E_x1fuel  # Demands for energy composite #
 (all,i,IND)(all,d,DST)  x1fuel(i,d) - [a1_fuel(i,d) ] =
x1prim_f(i,d)
-SIGMA1PRIMEN(i)*{p1fuel(i,d)-p1prim_f(i,d)}
-SIGMA1PRIMEN(i)*{a1_fuel(i,d)-a1prim_fsum(i,d)};            (5.17)
```

It is noteworthy that the demand for primary factors depends on the substitutability of energy inputs and primary factors. This substitutability is reflected by the elasticity SIGMA1PRIMEN(i), showing that changes in the demand for primary factors depends on the changes in the price of the primary factor composite relative to the primary factor-energy composite.

5.2.5 Demand for intermediate and primary factor-energy and other cost components

(Subsection 8.1.5. in VERM code)

The top level of the input nest combines the commodity and primary factor-energy composites. This nest also contains other costs, which represent other production costs not accounted for elsewhere. The demand equations for commodity composites are derived from the following optimisation problem for the ith industry. Total intermediate inputs, the primary-factor-energy composite and 'other costs' are combined using a Leontief production function, denoted by MIN(), given by

$$Z(i) = \frac{1}{A1TOT(i)} \times MIN\left(\frac{X1\_S(c,i)}{A1\_S(c,i)}, \frac{X1PRIM\_F(i)}{A1PRIM\_F(i)}, \frac{X1OCT(i)}{A1OCT(i)}\right) \tag{5.18}$$

In the above production function, Z(i) is the output of the ith industry, and the A variables are Hicks-neutral technical change terms.

As a consequence of the Leontief specification of the production function, each of the two categories of inputs identified at the top level of the nest are demanded in direct proportion to Z(i,q), as indicated in equations *E_x1_s* and *E_x1prim.*

```
Equation E_x1prim_f
(All,i,IND)(all,d,DST) x1prim_f(i,d) -a1prim_f(i,d) -
a1tot(i,d) = x1tot(i,d);
```
(5.19)

```
E_x1_s #Demand for composite intermediate goods#
(all,c,COMLFUEL)(all,i,IND)(all,d,DST)
   x1_s(c,i,d) = a1tot(i,d) + a1int_s(c,i,d) + x1tot(i,d)
       -0.15*{ppur_s(c,i,d) + a1int_s(c,i,d) - pint(i,d)};
```
(5.20)

Equation *E_x1prim_f* indicates that the demand for primary factors is directly proportional to output.

Equation *E_x1_s* indicates that the demand for commodity composite i is proportional to total production in industry I except in the case of energy commodities.

5.2.6 Total cost of output

(Subsection 8.1.6 in VERM code)

VERM allows for commodities to be produced by multiple industries. This is often the case when operating with highly disaggregated data. This has the implication that the cost of an industry's output does not necessarily equal the cost of producing a given commodity. From an industry perspective, the total cost of production is defined as the sum of the costs of primary factors, intermediate inputs, other cost tickets, and production taxes.

Variable industry costs arge given by equation E_pvart:

```
Equation E_pvar #Variable cost#
(all,i,IND)(all,d,DST)
ID01(VARCST(i,d))*[pvar(i,d)-a1tot(i,d)] =
 LAB_O(i,d)*[p1lab_o(i,d) + a1lab_o(i,d)+a1lab_od(i)] +
PUR_CS(i,d)*pint(i,d);                                    (5.21)
```

whereas total costs excluding taxes are given by equation E_pcst:

```
Equation E_pcst #Total cost of production excluding taxes#
(all,i,IND)(all,d,DST)
ID01(VCST(i,d))*[pcst(i,d)-a1tot(i,d)] =
 PRIM(i,d)*[a1prim(i,d)+p1prim(i,d)] +  PUR_CS(i,d)*pint(i,d);
                                                          (5.22)
```

Production taxes, in turn, are are given by:

```
Equation E_delPTX #Change in production taxes#
(all,i,IND)(all,d,DST)
delPTX(i,d) =
  0.01*PRODTAX(i,d)*[x1tot(i,d)+pcst(i,d)]    +
VCST(i,d)*delPTXRATE(i,d);                                (5.23)
```

Total costs are then given by equation E_pcst:

```
Equation E_ptot #Total cost# (all,i,IND)(all,d,DST)
 ID01(VTOT(i,d))*[ptot(i,d) + x1tot(i,d)] =
 VCST(i,d)*[pcst(i,d) + x1tot(i,d)] + 100*delPTX(i,d);    (5.24)
```

which states that the change in total costs of output depends on changes in ex-tax costs and on changes in production taxes.

VERM assumes perfect competition. This implies that the producer price of goods reflects only the costs of production. However, since a good can be produced by many industries, a link has to be established between producer prices and total costs. This link is given by equation

```
Equation E_xtotA # Average price received by multi-product
industries #
  (all,i, MIND)(all,d,REG)
    ptot(i,d) = sum{c,MINDCOM, MAKESHR1(c,i,d)*pmake(c,i,d)};
                                                          (5.25)

Equation E_xtotB # Price received by single-product industries
#
  (all,i,SIND)(all,d,REG) ptot(i,d) = pdom(SIND2COM(i),d);
                                                          (5.26)
```

where MAKE denotes the matrix relating commodities to the industries producing them, and p0com is the basic price (price excluding margins and indirect taxes) of commodity c.

5.2.7 Commodity mix of output

(VERM Subsections 8.1.6 VERM code)

In VERM, most industries produce multiple products, and most products can stem from several industries. Notable examples are for energy products, where several petroleum products stem from refining, and service sectors, which produce many different services. There are also some products, such as wood and wood residue used for heating and energy production that can stem from several industries. When an industry produces several outputs, its output mix may be affected by relative prices of the products. The supply of goods to the domestic market as opposed to export markets may also be modelled as a multi-production decision.

The output split of the production of each industry is described by the MAKE matrix. It is assumed that the product mix is adjusted, when the relative prices of the commodities change. Formally, the choice of output mix becomes one of maximising profits by deciding on the product split, subject to the market prices for the commodities in question (Dixon et al. 1992). Industry i's problem is to choose outputs, Q1(c,i), to maximise revenue

$$\sum_{C \in COM} P0COM(c,i)Q1(c,i) \tag{5.27}$$

subject to

$$X1TOT(i) = \underset{C \in COM}{CET}\left\{\frac{Q1(c,i)}{A1Q(c,i)}\right\}, \tag{5.28}$$

where CET() denotes Constant elasticity of transformation technology.

In VERM code, the solution to the output mix problem is given by the supply function E_q1:

```
E_xmake  # Supplies of commodities by industries #
 (all,c,COM)(all,i,IND)(all,d,REG)
  xmake(c,i,d)+a1q(c,i,d) = x1tot(i,d) +
SIGMAOUT(i)*[pmake(c,i,d)- ptot(i,d)]
            + SIGMAOUT(i)*[ -a1q(c,i,d) + a1qsum(i,d)];
```
(5.29)

In equation E_xmake, SIGMAOUT(i) is the transformation elasticity between the commodities the industry produces. The equation specifies that the percentage change in the supply of commodity c by multiproduct industry i is made up of two parts. The first is x1tot(i,d), the percentage change in the overall level of output of industry i in region d. The second is a price-transformation term. This compares the percentage change in the price received by industry i for product c (pmake(c,i,d)) with the weighted average of the percentage changes in the prices of all industry i's products (ptot(i,d)). Finally, a1q is output c-enhancing technological change.

The total output of each commodity is given by

```
Equation
E_xcom # Total output of commodities made by several
industries #
(all,c,COM)(all,d,REG)
xcom(c,d) = sum{i,IND,
MAKESHR2(c,i,d)*xmake(c,i,d)}+f_x0com(c,d);
```
(5.30)

where xcom(c) is the total output of commodity c.

The supply of commodities to the domestic and export markets is modelled similarly. The idea here is that supply to export markets reacts to the relative prices of the domestic and the export markets.

The supply is determined by four equations in VERM code. The first of these relates the supply decision between domestic and export markets to relative prices in these markets:

```
!zero profit!
Equation E_xtotA # Average price received by industries #
  (all,i, MIND)(all,d,REG)
    ptot(i,d) = sum{c,MINDCOM, MAKESHR1(c,i,d)*pmake(c,i,d)};
```

(5.31)

and

```
Equation E_xtotB # Price received by single-product industries
#
  (all,i,SIND)(all,d,REG) ptot(i,d) = pdom(SIND2COM(i),d);
```

(5.32)

VERM assumes that users of goods regard variants of a particular good produced by different industries as imperfect substitutes. This implies that the demand for these varieties is price-elastic. The demand for these varieties is implied by

```
Equation E_pmake  # Demands for commodities from industries #
(all,c,COM)(all,i,IND)(all,d,REG)
    pmake(c,i,d) = pdom(c,d) - 0.5*[xmake(c,i,d) - xcom(c,d)];
```

(5.33)

This, together with the output and production equations above completes the description of supplies.

5.2.8 Demands for investment goods

(Section 8.2 in VERM code)

VERM follows standard AGE practice in modelling the production of capital goods with an accounting sector, whose task it is to combine inputs to form units of capital. In choosing these inputs they cost minimise subject to a Leontief technology. Capital is produced with inputs of domestically produced and imported commodities. No primary factors are used directly as inputs to capital formation. Where VERM differs from most AGE models is the description of the capital goods themselves. In VERM, capital is genuinely sector specific, in other words, the commodity inputs for capital to each industry are unique. This means that capital is not malleable but that it will only adjust slowly, over time.

The model's demand for investment goods are derived from the solutions to the investor's two-stage cost-minimisation problem. At the bottom level, the total cost of the domestic/foreign-import composite (X2_s(c,i)) is minimised subject to the CES production function

$$X2\_S(c,i) = CES\{X2(c,s,i)\} \quad \mathrm{c} \in \mathrm{COM}\ ,\ \mathrm{i} \in \mathrm{IND}. \tag{5.34}$$

At the top level of the nest, the total cost of commodity composites is minimised subject to the Leontief function

$$X2TOT(i) = \underset{c \in com}{MIN}\{\frac{X2\_S(c,i)}{A2\_S(c,i)}\} \quad \mathrm{i} \in \mathrm{IND}, \tag{5.35}$$

where the total amount of investment in each industry (X2TOT(i)) is exogenous to the cost-minimisation problem and the A2_S(c,i) are technological-change variables in the use of inputs in capital creation. The variable X2_S(c,i) represents the demand for commodity composite c used for investment by industry i. The resulting demand equations for the composite inputs to capital creation (E_x2) correspond to the demand equations for the composite input to current production, and are, in VERM code, given by

```
Equation E_x2 # Dom/imp substitution #
(all,c,COM)(all,s,SRC)(all,d,DST)
    x2(c,s,d) = x2_s(c,d) - SIGMADOMIMP(c)*[ppur(c,s,"inv",d)-
pinvest(c,d)];                                              (5.36)
```

In equation E_x2 (5.36), the parameter SIGMADOMIMP(c) is the Armington elasticity between domestic and imported commodities. Furthermore, as in the case of intermediate inputs, there are twist variables that present changes in the domestic-imported composition of the composite goods X2_S(c,i).

While the commodity demands related to investment are determined by the equations above, they do not determine the level of investment. In year-to-year dynamic simulations, investment dynamics is determined by the dynamic theory of VERM, which is described in section 5.3. In the context of the present section, it may be important to realise that the fact that capital stocks are not malleable implies path-dependency.

5.2.9 Household demand

(Section 8.3 in VERM code)

Household demand is modelled with a linear expenditure system in VERM. The representative household determines the optimal composition of its consumption bundle by choosing commodities to maximise a Stone-Geary utility function subject to a household budget constraint. A *Keynesian* consumption function determines aggregate household expenditure as a function of household disposable income. The consumption of domestic and import commodities is given by equation E_x3 and is of similar form than those for intermediate input and investment demands:

```
Equation E_x3 # Dom/imp substitution in consumption#
(all,c,COM)(all,s,SRC)(all,d,DST)
x3(c,s,d) = x3_sh(c,d) - SIGMADOMIMP(c)*[ppur(c,s,"hou",d)-
p3(c,d)];
```
(5.37)

As can be seen from equation E_x3 (5.37), there are twist variables also for household demand. The parameter SIGMA3(c) gives the Armington elasticity between domestic and import commodities in consumption.

With the Stone-Geary utility function, total consumption of each commodity composite (X3_S(c)) consists of two components: a subsistence (or minimum) part (X3SUB(c)) and a luxury (or supernumerary) part (X3LUX(c)):

$$X3(c) = X3SUB(c) + X3LUX(c) \qquad c \in \text{COM} \tag{5.38}$$

A characteristic of the Stone-Geary function is that only the luxury components affect per-household utility (UTILITY), which has the Cobb-Douglas form

$$UTILITY = \frac{1}{QHOUS} \times \prod_{c \in COM} X3LUX(c)^{S3LUX(c)} \tag{5.39}$$

with $\sum_{c \in COM} S3LUX(c) = 1.$

Because the Cobb-Douglas form gives rise to exogenous budget shares for spending on luxuries,

$$P3(c)X3LUX(c) = S3LUX(c)W3LUX(c) \qquad c \in \text{COM}, \tag{5.40}$$

where S3LUX(c) may be interpreted as the marginal budget share of total spending on luxuries (W3LUX). Rearranging (5.40), substituting into (5.38) and linearising gives equation the VERM equation E_x3lux:

```
Equation E_xlux # Luxury demand for composite commodities #
  (all,c,COM)(all,d,DST)(all,h,HOU)
   xlux(c,d,h) + p3(c,d) = wlux(d,h) + alux(c,d,h);        (5.41)
```

while the subsistence component is given by

```
Equation E_xsub # Subsistence demand for composite commodities
#
  (all,c,COM)(all,d,DST)(all,h,HOU) xsub(c,d,h) = nhouh(d,h) +
asub(c,d,h);                                                  (5.42)
```

where the subsistence component is proportional to the number of households q and to a taste-change variable (a3sub(c)).

Finally, the total demand for composite commodities is given by

```
Equation E_x3_s  # Total household demand for composite
commodities #
  (all,c,COM)(all,d,DST)(all,h,HOU)
   x3_s(c,d,h) = BLUX(c,d,h)*xlux(c,d,h) + [1-
BLUX(c,d,h)]*xsub(c,d,h);                                     (5.43)
```

In (5.43), $B3LUX(c) = -EPS(c)/FRISCH$, FRISCH is the Frisch parameter, i.e. the share of supernumerary expenditure on commodity c in total expenditure on commodity c, and EPS(c) is the expenditure elasticity for commodity c.

The taste changes are controlled by a number of equations that allow the budget shares to be shocked. This may change the expenditure elasticities. Expenditure elasticities are also affected by prices and income.

The equations above determine the composition of household demands, but do not determine total consumption. This is determined by household disposable income, which is taken up in a later section.

5.2.10 Export and inventory demands

(Section 8.4. in VERM code)

Exports are divided into two groups in VERM: traditional exports, which comprise most manufacturing industries, and non-traditional exports, which cover the exports of services and transports. Exports account for a relatively large share of the total sales of traditional export commodities. In contrast, the non-traditional, or collective, exports stem from industries which are mostly domestically oriented. The advantage of the distinction is that it prevents world market prices from feeding into the prices of service exports. The division of exports into the two categories is controlled by simple set definitions.

For each category, the model allows for a number of ways to change the composition and volume of export demand, for example, to accommodate export forecasts for specific export industries or forecasts for aggregate exports. Forecasts or scenarios of world price developments can also be easily accommodated.

Traditional export commodities are assumed to face a downward-sloping foreign-export demand functions

$$X4(c,d) = F4Q(c,d) \times F4QALL \times \left(\frac{P4(c,d) \times EXCHR}{F4P(c,d) \times F4PALL} \right)^{EXP\_ELAST(c)}$$

$$c \in \text{TRADEXP} \quad d \in \text{XDEST} \qquad (5.44)$$

X4(c,d) is the export volume of commodity d to destination d. The parameter EXP_ELAST(c) is the (constant) own-price elasticity of foreign-export demand. As EXP_ELAST(c) is negative, traditional exports are a negative function of their foreign-currency prices on the export markets (P4(c,d)). They are also affected by the exchange rate. The variables F4P(c,d) and F4PALL allow for horizontal (quantity) and vertical (price) shifts in the demand schedules. The variables F4Q(c,d) and F4QALL allow for economy-wide horizontal and vertical shifts in the demand schedules. In VERM code, the demand equation for traditional exports is given by E_xexp, which defines individual export demand functions for each of the traditional export goods:

```
Equation E_xexp #Export demands #
(all,c,COM)(all,s,SRC)(all,d,DST)
   xexp(c,s,d) = f_qexp_dc +
f_qexp_d(c)+f_qexp_c(d)+f_qexp(c,d)+fqexp(c,s)
- ABS[EXP_ELAST(c)]*[ppur(c,s,"Exp",d)-f_pexp_dc -fpexp(c,s)-
f_pexp_d(c)-phi];
```

(5.45)

Equation (5.45) is the percentage change version of (5.44), but makes explicit that VERM allows for exchange rate changes to affect exports. Exchange rate changes are given by the variables phi_eu and phi_non_eu, with the dummies DUM_X(d) controlling which of the export markets is in question.

5.2.11 Inventory demands

(Section 8.14. in the VERM code)

While stocks may be important for the supply of non-perishable goods, VERM does not attempt to offer a theory of the determination of stocks. It does allow for two alternative assumptions, namely, either the assumption that stocks follow current production, or that they are exogenously determined. The determintation of stocks is given by

```
Equation E_xstocks #Inventory demands #(all,i,IND)(all,d,DST)
  xstocks(i,d) = x1tot(i,d) + f_stocks(i,d)+ f_stocks_d(i);
```

(5.46)

5.2.12 Margin demands

(Section 8.5. in the VERM code)

In VERM, some of the commodities are used as margins. Typical margin commodities are wholesale and retail trade, and transport services provided by several transport sectors. These commodities, in addition to being consumed directly by the users (e.g., consumption of transport services when taking holidays or commuting to work), are also consumed to facilitate trade (e.g., the use of transport services to ship commodities from point of production to point of consumption). The latter type of demand for transport is called demand for margins. Margin demands are assumed to be proportional the demand for each commodity. The demand for margins related to consumption demand is given by

```
Equation E_xtradmar # Leontief demand for margins #
 (all,c,COM)(all,s,SRC)(all,m,MAR)(all,r,ORG)(all,d,DST)
  xtradmar(c,s,m,r,d) =  xtrad(c,s,r,d) + atradmar(c,s,m,r,d)
```

(5.47)

5.2.13 Regional demand for delivered goods

(Section 8.6. in the VERM code)

An important part of a regional model is the handling of trade between the regions in the country. VERM assumes that goods stemming from each region are imperfect substitutes, implying again price-sensitivity in their demand. Margins have an important role here, since they account for the difference between delivered-prices of goods stemming from different regions and consumed within a region. The demand equations are given by

```
Equation E_xtrad # CES between goods from different regions #
(all,c,COM)(all,s,SRC)(all,r,ORG)(all,d,DST)
xtrad(c,s,r,d)=xtrad_r(c,s,d)-SIGMADOMDOM(c)*[pdelivrd(c,s,r,d)-
puse(c,s,d)]
 +  sum{k,ORG,  [if(k eq r,1) -
DELIVRD(c,s,k,d)/{DELIVRD_R(c,s,d)+0.000000000001}]*twistsrc(c,s
,k) };
```

(5.48)

5.2.14 Purchaser's prices and zero-profit conditions

(Section 8.7. in the VERM code)

Prices in VERM are divided into base prices, prices as delivered, and purchasers' prices, which include taxes. The base price of a product reflects the costs of production including production taxes but excluding indirect taxes levied on the output. Importantly, they also exclude margins. The base price comes close to the concept of producer's price – and is often called that – but it actually differs slightly from standard usage of the term producer's price because base prices exclude margins. The distinction is made because VERM models margins explicitly.

The delivery price of goods to stemming from different regions is given by

```
Equation E_puse # Delivered price of regional composite good
c,s                    to                  d                    #
 (all,c,COM)(all,s,SRC)(all,d,DST)
  ID01(DELIVRD_R(c,s,d))*puse(c,s,d)  !  CES  price  index  !
                       =
sum{r,ORG,DELIVRD(c,s,r,d)*pdelivrd(c,s,r,d)};              (5.49)
```

The third price concept, purchaser's prices, includes margins and indirect taxes and is comparable to the standard usage of the term. The basic values of all flows are evaluated at basic prices, net of taxes, but taxes and margins are included in the purchasers' price. This means purchasers' prices may differ from customer to customer. For example, for household demand we have

```
Equation E_ppur # Purchasers prices #
(all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST)
 ppur(c,s,u,d) = puse(c,s,d) + tuser(c,s,u,d);              (5.50)
```

VERM assumes perfect competition in the sense that there are no mark-ups in pricing. If there is imperfect competition, it is taken into account by the inclusion of the other cost tickets. Equation E_p3 thus imposes zero profits for consumer prices. In the equation, p3(c,s) gives the change in purchasers' prices, which is linked to changes in the basic price p0(c,s), changes in the prices of margins (p0dom(c,s)), and changes in indirect consumption taxes (t3(c,s). Similar equations apply to investment, export, government and margin demands. Together, these equations define the purchasers' prices for all markets.

Of interest in equation (5.50) are the indirect taxes. They are a typical policy variable in simulations, and we explain their presentation in the model in more detail in the next sections.

5.2.15 Market-clearing equations

(Section 8.8 in the VERM code)

Market clearing implies that the demands for all commodities equal supply. This is achieved via the adjustment of prices. In VERM, we first define sales aggregates that summarise the sales to each market.

```
Equation E_pdomA # Demand = supply for non-margins #
 (all,c,NONMAR)(all,r,REG)
   xcom(c,r) = xtrad_d(c,"dom",r);
```

(5.51)

```
Equation E_pdomB # Demand = supply for margins #
 (all,m,MAR)(all,p,REG)
   MAKE_I(m,p)*xcom(m,p) =
TRADE_D(m,"dom",p)*xtrad_d(m,"dom",p)
                          + SUPPMAR_RD(m,p)*xsuppmar_rd(m,p);
```

(5.52)

5.2.16 Indirect taxes and indirect tax revenues

(Section 8.9. in the VERM code)

The VERM database covers indirect taxes in great detail, allowing the model to be used for the study very specific changes in them. The presentation of indirect taxes in TABLO code introduces a convention that is not familiar for most readers, however.

TABLO follows the convention that taxes are expressed in terms of powers of taxes. The meaning of a power of a tax is best illustrated with an example.

The power of a tax is given by

$$\Pi=1+r \tag{5.53}$$

is the power (one plus the rate) of the tax applicable to the basic value of the flow or output.

The indirect tax revenue on any commodity flow or production tax revenue on the basic value of an industry's output can be thus be written as

$$T = P*X*(\Pi-1)\ , \tag{5.54}$$

where

T is the tax revenue;

P is the basic price of the relevant commodity or industry output; and

X is the quantity flow or output; and

From (5.54), we obtain

$$100*del\_T = T*(p + x) + (T + B)*\pi \tag{5.55}$$

where

del_T	is the change in tax revenue;
B	is the basic value of the flow, i.e., B= P*X; and
p, x and π	are percentage changes in P, X and Π.

The reason for the convention of using powers instead of rates can be seen with the help of equation (5.55). By using powers of taxes instead of their rates, there is no danger of running into zeroes. If a tax has a zero rate, its power equals one, whence, say, the introduction of a non-zero rate can easily be accomplished. The use of powers also facilitates the analysis of tax policies. The power of a tax, Π, can be expressed as the product of shift variables, Πg,1…Πg,n:

$$\Pi = \Pi_{g,1\ldots}\Pi_{g,n} \quad . \tag{5.56}$$

This leads to the percentage-change equation:

$$\pi = \sum_{q=1}^{n} \pi_{g,q} \quad + \quad \sum_{q=1}^{m} \pi_{ph,q} \tag{5.57}$$

With obvious translations in notation, equation E_tuser gives the change in indirect taxes on goods:

```
Equation E_tuser # Usr x Dst taxes driven by commodity
specific shifter #
(all,c,COM)(all,s,SRC)(all,u,USR)(all,d,DST)
tuser(c,s,u,d)=tuser_ud(c,s)+tuser_sd(c,u);                    (5.58)
```

As can be seen from the equation, indirect taxes can be given commodity and industry specific shocks – or all indirect taxes can be given uniform shocks.. The advantage of this formulation is that it is easy to set up simulations where the tax for a particular good is changed.

Equation (5.54) also provides the underlying theory for keeping track of the evolution of indirect tax revenues in equations E_delV1TAX through E_delV5TAX in VERM code. For example, the revenue of indirect consumption taxes is given by

```
E_delTAXhou (all,c,COM)(all,s,SRC)(all,d,DST)
  delTAXhou(c,s,d)   =
0.01*TAX(c,s,"hou",d)*[x3(c,s,d)+puse(c,s,d)]
                 + 0.01*PUR(c,s,"hou",d)*tuser(c,s,"hou",d);
```
(5.59)

where delTAXhou(c,s,d) in the change in the revenue, TAX(c,s,"Hou") the coefficient presenting the basic flow of taxes, x3(c,s,d) the percentage change in the consumption of commodity c from source s, and tuser(c,s,"hou",d) is the percentage change in the power of tax for commodity c stemming form source s. With the help of the revenue equations, it is easy to link changes in taxes to their revenue, and to impose revenue neutrality or budget balance conditions as required.

5.2.17 GDP aggregates

(Sections 8.10.and 8.11. in the VERM code)

This section collects the income and expenditure aggregates of the GDP, as well as their price indices. Their definitions are straightforward, but it is worth noting that observing GDP from both the income and the expenditure sides provides a check for the consistency of the model's results.

The percentage change of the GDP from the income side is given by equation E_delGDPINCa and the following four equations by expenditure category. For example, the change in GDP attributable to Labour is given by

```
E_delGDPINCc (all,d,DST) delGDPINC(d,"Labour")
   = 0.01*LAB_IO(d)*wlab_io(d);
```
(5.60)

Nominal income side GDP is an add-up of these categories, given by

```
Equation E_wgdpinc
 (all,d,DST) GDPINC(d)*wgdpinc(d) = 100*sum{i,GDPINCCAT,
delGDPINC(d,i)};
```
(5.61)

Similarly, the percentage change of the expenditure side GDP in equation is an add-up of the changes attributable to the demand categories, with, for example, E_delGDEXPa, giving the change due to final demand:

```
Equation
 E_delXGDPEXPa (all,d,DST)(all,u,FINDEM) delXGDPEXP(d,u)
    = 0.01*PUR_CS(u,d)*xfin(u,d);                                    (5.62)
```

The percentage change of the expenditure side GDP equation E_xgdpexp:

```
Equation E_xgdpexp
 (all,d,DST) GDPEXP(d)*xgdpexp(d) = 100*sum{i,GDPEXPCAT,
delXGDPEXP(d,i)};                                                    (5.63)
```

and its value by

```
Equation E_wgdpexp   (all,d,DST) wgdpexp(d) =
xgdpexp(d)+pgdpexp(d);                                               (5.64)
```

VERM code also contains GDP decompositions which are dealt with in a later section.

5.2.18 Trade flows

(Sections 8.13. in the VERM code)

5.3 VERM dynamics

VERM contains several dynamic links between periods. These are each described in the current section. Central to the results that the model produces are the theory of investment, the lagged adjustment mechanisms in real wages and – if so specified – national saving and government finances. The section commences with investment theory and the wage mechanism is described in section 5.3.2. Government finances are covered in the context of national income and expenditure aggregates in section 5.4.

5.3.1 Government accounts, balance of payments and national saving

(Sections 8.14 in the VERM code)

VERM covers the three main levels of government in Finland, namely, the central government, the communal sector, and the social security funds. Each of these is subject to genuine budget constraints and can also run deficits independently if necessary. Transfers between the sectors also form an important part of the expenditures and incomes of these sectors.

To commence with the expenditures, VERM contains no explicit theory for the determination of the government and inventory demands for commodities. Most often it is assumed that the commodity composition of government demand is exogenous. However, it is possible to endogenise total government expenditure or parts of it. In year-to-year simulations, there are also many parts of government demand that typically stem from exogenous scenarios outside of the model, such as the demand for health care and educational services provided by the public sector. Inventory demand also contains no explicit theory. In forecast simulations, it is usually assumed that inventories do not change or that they follow output.

In VERM code, equation E_x5 (5.45) gives government demand for commodities:

```
Equation E_xgov #Public sector demands
#(all,c,COM)(all,s,SRC)(all,d,DST)
      xgov(c,s,d) =  fgovtot(d) + fgov(c,s,d) + fgov_s(c,d) +
fgovgen;                                                    (5.65)
```

Equation E_x5 contains a useful device for allowing the growth of government demand to follow overall consumption growth. In the equation, the percentage change in government demand is affected by the shift variables f5(c,s) and f5tot. investment

```
Equation E_w2totg_is # Value of public sector investments #
(all,d,DST)(all,s,PSEC)
V2TOT_G_IS(d,s)*w2totg_is(d,s) =
!sum{i,IND,V2TOTSG_(d,i,s)*w2totg_s(d,i,s)};!
sum{i,IND,V2SHRSG_(d,i,s)*INVEST_C(i,d)*(s2gov(i)+pinvitot(i,d
)+x2tot(i,d))};                                             (5.66)
```

The determination of government incomes has been touched upon in the previous discussion on tax and tariff revenues. We have also discussed government demands for commodities and investment goods. As already noted, VERM contains no explicit theory for the determination of specific commodity demands by the government. It does contain various alternatives for linking overall government demands to the rest of the economy, however. Here, we concentrate on the more dynamic elements of the public sector, especially transfer payments from the government to other sectors in the economy, as well as the various specifications for overall government spending that VERM allows for.

In VERM, government revenue consists of income taxes and indirect taxes, and transfers from other sectors. The revenue from taxes and tariffs is defined in earlier parts of the VERM code.

In the model, all incomes are taxed, and thus the revenue is the sum of taxes on labour and capital incomes, as well as taxes on benefits (which include age-dependent, unemployment, and other benefits). The coefficients are calculated form base data by formulas in section 5.10, and include aggregate income taxes and aggregate tax revenues:

```
INCTAX = TAX_LAB + TAX_CAP + TAX_LND + TAX_AB + TAX_OB + AX_UB;
```
(5.67)

and

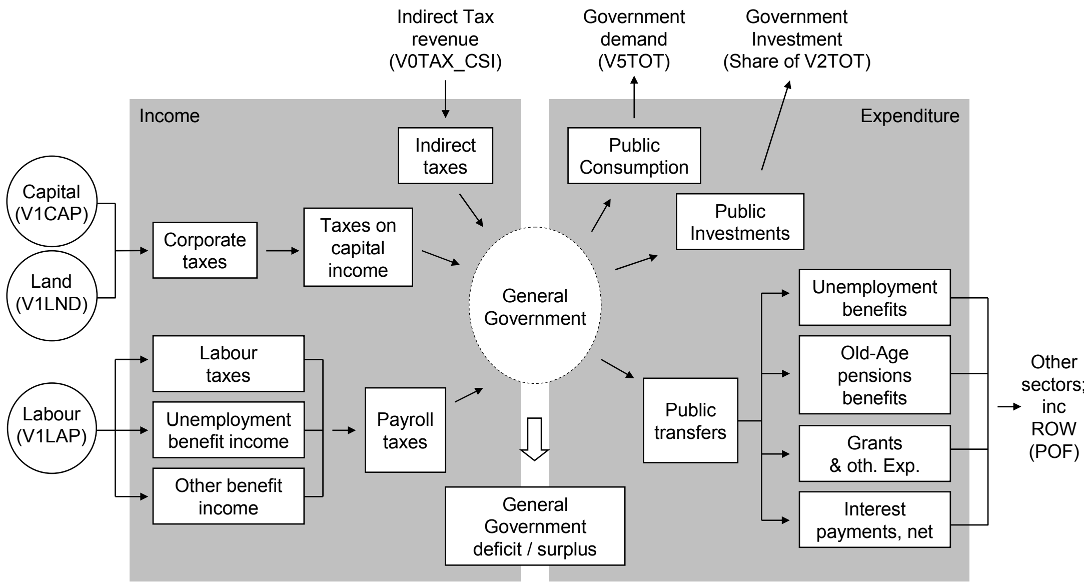
```
NET_TAXTOTG = V0TAX_CSI + INCTAX + COL_PAYRTOT;
```
(5.68)

Government spending, on the other hand, consists of the demand for commodities for immediate consumption and investment, and transfers payments. The transfers are made up of age-dependent, unemployment, and other benefits, grants, and net interest payments on the public sector debt. The corresponding coefficients are:

```
TRANS = UNEMPBEN + AGEBEN + OTHBEN + GRANT + NETINT_G;
```
(5.69)

Figure 5.1. shows schematically the allocation of overall government spending is allocated and the collection of government revenue.

Figure 5.1 Government spending and revenue in VERM



The public sector deficit is defined as the difference between expenditures and revenues, including net interest on government debt. In VERM code, government deficit evolves according to

```
Equation E_d_gov_defs1
  # Public sector deficit, or public sector financing
transactions #
(all,d,DST)(all,s,GSEC)
  100*d_gov_defs(d,s) =
                V5TOTS(d,s)*w5tots(d,s) +
V2TOT_G_IS(d,s)*w2totg_is(d,s)
                +      100*d_othcapgovs(d,s)
                     - NET_TAXTOTS(d,S)*nettaxtots(d,s)
                     - OTHGOVREVS(s,d)*oth_govrevs(d,s) +
100*d_transfs(d,s)
                +GOVTOMUN(d)*f_govtomun(d)
                +GOVTOSSF(d)*f_govtossf(d)
                -MUNTOGOV(d)*f_muntogov(d)
                -SSFTOGOV(d)*f_ssftogov(d)
;
```

(5.70)

where the change in public deficit is caused by changes in the value of public consumption, changes in the value of public investment demand, change in other government capital expenditure, changes in net tax revenue, changes in other government revenue; and changes in transfer payments.

The accumulation of government debt is linked to deficits and is given by

```
Equation E_d_psd_t1s # Public sector debt, end of year #
(all,d,DST)(all,s,PSEC)
d_psd_t1s(d,s) = d_psd_ts(d,s) +
d_gov_defs(d,s)+d_f_psd_t1s(d,s);                          (5.71)
```

In other words, current deficits contribute to the debt stock. As with other financial assets, the evolution of debt is governed by the following equation:

```
Equation E_d_f_psd_ts
  # Gives shock to start-of-year public sector debt, yr-to-yr
sims #
(all,d,DST)(all,s,PSEC)
d_psd_ts(d,s) = [PSDATT_1_B(d,s) - PSDATT_B(d,s)]*d_unity +
d_f_psd_ts(d,s);                                           (5.72)
```

We now turn to the determination of the transfers from the government. VERM assumes that these depend on the size of the population receiving them but also allows them to be indexed to consumer prices and/or real wages, or to be given entirely exogenously. For example, changes in pensions and age-related benefits can be indexed to real wages and consumer prices:

```
Equation E_age_bens # Old age benefits paid by government #
(all,d,DST)(all,s,PSEC) age_bens(d,s) =
popaged(d)-popagednat+f_age_bens(d,s)+f_age_ben_nat;       (5.73)
```

where pop_aged gives the change in old-age population, p3tot changes in consumer prices and real_wage_c is the change in the pre-tax real wage rate. However, it is also possible to give the forecast of age benefits exogenously, using information from outside of the model.

Finally, overall government investment can also be determined exogenously, or it can be tied down to the evolution of the overall level of private investments. The choice between exogenous scenarios and the in-built mechanisms in the model depend entirely on the nature of the policy experiment.

VERM allows for different budget, or fiscal rules to be used. One of them is dynamic and involves the balancing of deficits with an eye on overall national saving. This mechanism allows for a specification where the government budget is not immediately balanced. This is accomplished by specifying a budget rule that links national saving – which is related to government deficits as will be explained late – and employment, to allow for deficits while employment is below normal (or rather, expected).

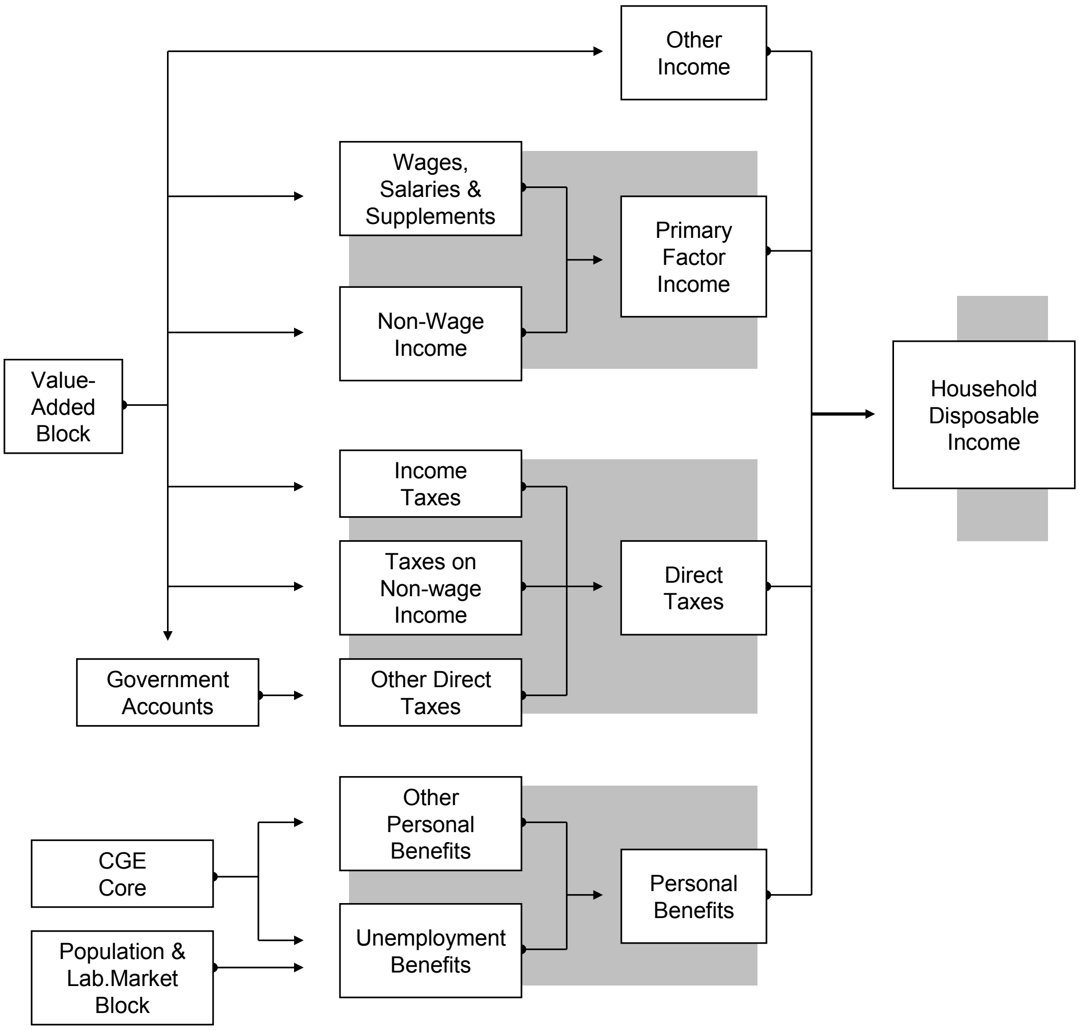
5.3.2 Income and Saving aggregates

(Sections 8.15. in the VERM code)

Section 8.15. 1 in the VERM code contains definitions for household income and saving aggregates. The section commences with household disposable income.

In VERM, household income stems from several sources: factor incomes, transfers from the government, and from foreign assets. These sources are summarised in figure 5.2.

Figure 5.2 Household disposable income



In VERM code, household income is given by

```
Equation E_hdispinc # Household disposable income #
(all,d,DST) HOUS_DIS_INC(d)*hdispinc(d) =
    GDPEXP(d)*wgdpexp(d)
+ (sum{s,PSEC,100*d_transfs_h(d,s)
   - {NET_TAXTOTS(d,s)*nettaxtots(d,s) +
OTHGOVREVS(s,d)*oth_govrevs(d,s)}})
   +SHRASS_H(d)*[C_INTAD*intad+C_INTAE*intae -C_INTLD*intld -
C_INTLE*intle];                                              (5.74)
```

On the RHS of the equation, factor incomes are expressed in terms of GDP, from which government expenditures and debt servicing costs are deducts. The second but last line in the equation represents asset and equity income from abroad less interest on foreign debt and equities.

```
Equation E_housav # Household saving #
(all,d,DST) HOUS_SAV(d)*housav(d) =
    HOUS_DIS_INC(d)*hdispinc(d) - V3TOT(d)*wfin("Hou",d);
```

(5.75)

The level of consumption in VERM is determined by a consumption function relating disposable income to consumption. The consumption function is given by

```
Equation E_f3tot # Consumption related to disposable income #

(all,d,DST)(all,h,HOU)  HOUPUR_C(d,h)*w3tot(d,h) =
    AV_PROP_CON(d)*HOUS_DIS_INC(d)*[f3tot(d) + hdispinc(d)];
```

(5.76)

where AV_PROP_CON is the average propensity to consume out of disposable income. A constant propensity to consume is a characteristic of the steady state of many theories of economic growth. VERM displays this characteristics but it also allows for short run deviations from a constant propensity to consume. VERM also allows the propensity to consume to be determined by the model in forecast simulations, for example, to accommodate for short term macroeconomic forecasts of the macro variables stemming from outside the model. Finally, the consumption function also implies a link between disposable income and saving.

5.3.3 Balance of payments

(Subsection 8.15.2 in the VERM code)

Equation E_p0imp_c, defines an index for the landed-duty-paid prices of imports. Comparison of results for p0imp_c defined in this equation with those for the GDP deflator (p0gdpexp) provides a useful indication of changes in the competitive position of imports in the domestic economy. This can be important in explaining simulated shifts in the ratio of imports to GDP.

```
E_w0imp_c          w0imp_c = x0imp_c + p0imp_c;
```

(5.77)

Equation E_toft, defines the percentage change in the terms of trade as the difference between the percentage changes in the f.o.b. price index for exports and the c.i.f. price index for imports.

```
E_p0toft            p0toft = p4tot - p0imp_c;
```
(5.78)

Finally, equation E_p0realdev defines the percentage real devaluation. This is normally measured by

```
E_p0realdev      p0realdev =  p0imp_c - p0gdpexp;
```
(5.79)

In E_realdev, we measure domestic inflation by the percentage change in the GDP deflator (p0gdpexp), and we assume that foreign inflation is reflected in the foreign currency price of imports, implying that it equals the percentage change in the c.i.f. domestic price of imports (p0cif_c).

We still have to establish the link between the domestic economy and foreign sectors. This link is provided by accounting for national saving, foreign assets, the trade balance, and the accumulation of new assets.

National saving is defined as an aggregate private and government saving. In the code, it is given by

```
Equation E_natsav
  # National saving: household saving plus public sector
surplus #
  NAT_SAV*natsav = HOUS_SAV_d*housav_d +100*d_govsav_nat;
```
(5.80)

where the public sector surplus is defined by

```
Equation E_d_govsav_nat # Government saving #
  d_govsav_nat = - d_gov_def_nat
                  + 0.01*V2TOT_G_I_D*w2totg_is_ds
                  + d_othcapgov_nat;
```
(5.81)

Equation (5.81), a simple national accounting identity, links household disposable income, government deficits, and government investments.

Figure 5.3. shows how the domestic economy is linked to the foreign sectors. The trade balance reflects the exports and imports of the economy, the theory of which was explained in the first part of this chapter. The second link is formed

by the (historically given) ownership of foreign assets and liabilities, which determines asset incomes from abroad and interest payments on foreign assets and equities. The current account balance provides the final link, connecting changes in the ownership of assets to changes in domestic saving and domestic deficits, as well as changes in the trade balance.

The current account balance combines the trade balance with financial account balances. In VERM code, changes in the current account are given by

```
Equation E_d_cab
100*d_cab =
  [V4TOT_D*w4tot + C_INTAD*intad +C_INTAE*intae]
    - [V0IMP_C_D*w0imp_c +  C_INTLD*intld + C_INTLE*intle ]
+100*d_nettrn;                                              (5.82)
```

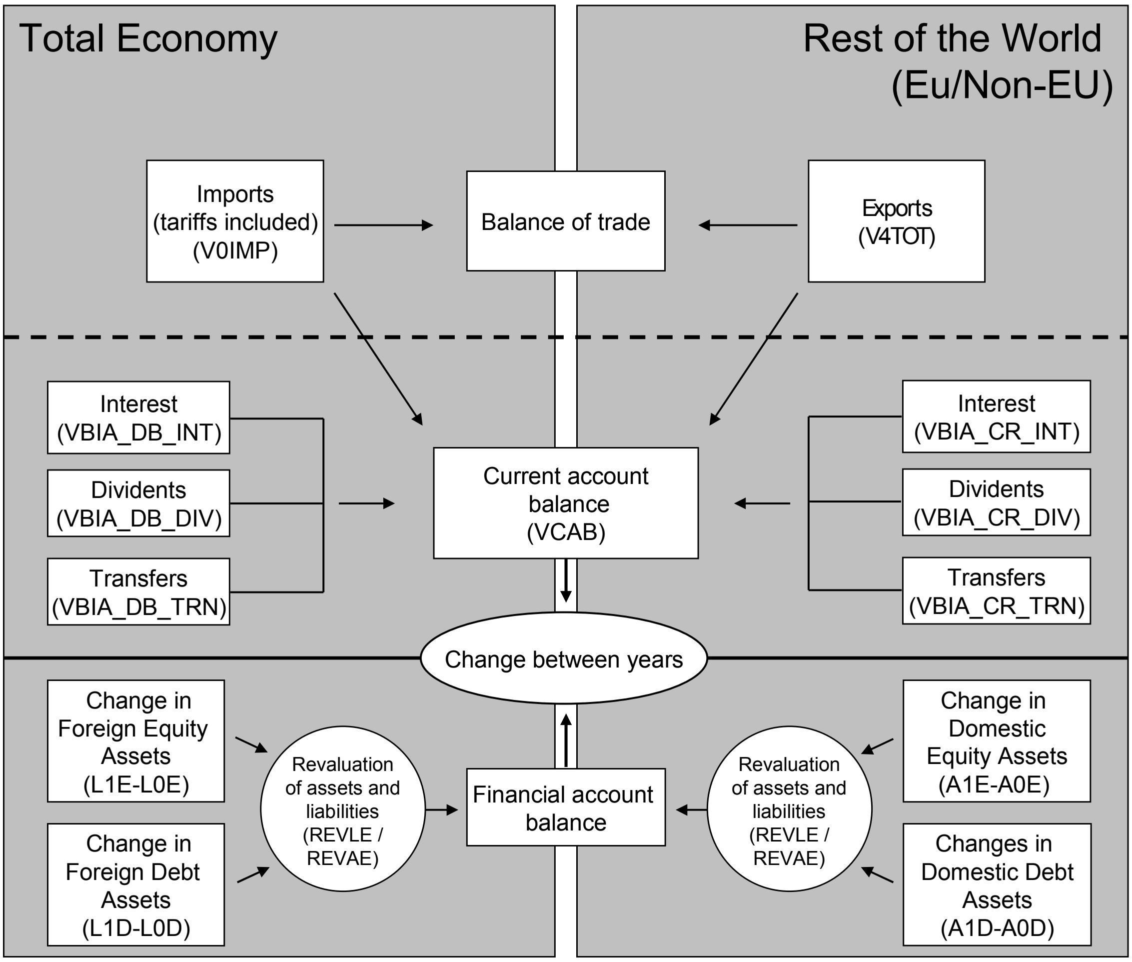
where C_INTAD and C_INTAE are incomes from foreign assets and equities, and C_INTLD and C_INTLE are outstanding foreign debt and liabilities. The changes in the contribution of these assets to the current account balance (the terms intad, intae, intld and intle), account for changes in the interest rates on these assets, on to changes in their amounts. The term d_nettrn present changes in net transfers abroad, which consist of payments to international organisations and the like that can be taken as exogenous. VERM assumes that changes in the ownership of foreign assets are due to changes in national saving, or to revaluation effects. Part of national saving is thus directed towards foreign assets and equities.

The accumulation of foreign assets is affected by changes in national saving. The accumulation of foreign liabilities, on the other hand, is affected by changes in the net financial position of the economy. For example, when the current account balance is weakening, because of unfavourable changes in relative prices increasing the trade balance deficit, the debt servicing costs do not respond in the short run. Assuming national savings are not affected, the deficit must be met with increased foreign liabilities. This connection is recognised in equation E_d_newle, which links changes in liabilities to the current account deficit:

```
Equation E_d_newle
100*d_cab = C_FE*NAT_SAV*(fe+natsav) +
C_FD*NAT_SAV*(fd+natsav)
                        - 100*d_newle - 100*d_newld;        (5.83)
```

VERM assumes that the split of new lending between equities and debt is only changed from what it historically was if there are changes in the rates of interest, which are exogenously given.

Figure 5.3 The links between the current account and asset accumulation in VERM



5.3.4 Investment over time

(Section 8.16 in the VERM code)

5.3.4.1 Capital growth and the logistic investment function

In VERM, investment over time is determined by functions which specify that investors are willing to supply increased funds to industry j in response to increases in j's expected rate of return. However, investors are cautious. In any year, the capital supply functions limit the growth rate of the capital stocks so that disturbances in rates of return are eliminated only gradually.

The resulting dynamics for capital stocks and investment can be compared with that in models recognizing costs of adjustment (see, for example, Bovenberg and Goulder, 1991). In costs-of-adjustment models, industry i's capital growth (and investment) in any year is limited by the assumption that the costs per unit of installing capital for industry i in year t are positively related to the i's level of investment in year t. In VERM, it is assumed that the level of i's investment in year t has only a negligible effect (via its effects on unit costs in construction and other capital supplying industries) on the costs per unit of i's capital. Instead of assuming increasing installation costs, we assume that i's capital growth in year t is limited by investor perceptions of risk. Investors are willing to allow the rate of capital growth in industry i in year t to move above i's historically normal rate of capital growth only if they expect to be compensated by a rate of return above i's historically normal level.

This MONASH theory of investment is fully covered in Dixon and Rimmer (2002); here, we give an introduction to it while explaining the central equations in VERM model code.

In VERM code, the most important equations concerning the dynamics of investment are given by the accumulation equations for capital and the equations concerning expected and actual rates of return on capital.

In levels, the accumulation of capital in any industry is defined by equation

$$K_{+1} = (1 - D)*K + I \tag{5.84}$$

where

K is the capital stock at the beginning of year t;

K_{+1} is the capital stock at the end of year t;

I is investment during year t; and

D is a parameter giving the rate of depreciation.

In change form, (5.65) can be written as

$$K_{+1}*k_{+1} = (1 - D)*K*k + I*y \ , \tag{5.85}$$

where k_{+1}, k and y are percentage deviations in the values of K_{+1}, K and I from their values in the initial solution for year t. E_x1cap_at_tplus1 can be obtained from (5.66) by adding industry identifiers i, by making obvious notational conversions and by including the coefficient, TINY which prevents indeterminacy problems that arise if QCAPATTPLUS1(j) is zero.

In VERM code, equation (5.66) is given by E_x1cap_at_tplus1:

```
Equation E_x1cap_tplus1
  # Capital accum thru the fcst year (t) related to investment in
    the year #
  (all,i,IND) [QCAPATTPLUS1(i) + TINY]*x1cap_tplus1(i)
          = [1-DEP(i)]*QCAPATT(i)*x1cap(i) + QINVEST(i)*x2tot(i);
```
(5.86)

The coefficients QCAPATT(j) and QINVEST(j) are both evaluated from values and prices, while QCAPATTPLUS(j) is evaluated from QCAPATT(j) and QINVEST(j) according to (5.67). An intuitive interpretation for the equation is that it defines the quantity of investments needed to satisfy the growth rate of capital stock in period (t+1) specified by the inverse logistic relationship described shortly.

VERM dynamics requires an initial solution. In most simulations, the initial solution for year t is the final solution for year t–1. In these simulations, the initial solution for an industry's opening capital stock is the opening capital stock in the previous year. In the year t computation, the percentage deviation (k) that should then be imposed on the opening capital stock (K) is given by

$$k = 100*(KBASE_{+1} - KBASE)/KBASE \tag{5.87}$$

or equivalently by

$$k = 100*(IBASE - D*KBASE)/KBASE \quad , \tag{5.88}$$

where KBASE, $KBASE_{+1}$ and IBASE are the initial solutions for K, K_{+1} and I, i.e., KBASE and $KBASE_{+1}$ are the opening and closing capital stocks in year t-1, and IBASE is investment in year t-1. In the model code, this equation is given by

```
Equation E_d_f_ac_p_y
  # Gives shock in yr-to-yr forecasting to capital at
beginning of year t #
(all,d,DST)(all,i,IND) [QCAPATT(d,i) +
TINY]*x1cap(i,d)!x1cap(d,i)!=
     100*{QINV_BASE(d,i) -
DEP(d,i)*QCAPATT_B(d,i)}*d_unity+100*d_f_ac_p_y(d,i);
```
(5.89)

In equation (5.70), the variable d_unity merits a note. Its function is to give a shock of unity to the equation, implying that the change on the RHS of the equation obtains the value 100 times QINV_BASE(i) – DEP(i)*QCAPATT_B(i). It is also worth emphasizing that, for the parts of a simulation that cover history, this shock stems from the data for the base year, implying the model has to satisfy the national accounting identities for capital and investment for several consecutive years. Finally, in making the year t computation, we could treat k as an exogenous variable and compute its value outside the model in accordance with (5.68) or (5.69). It is more convenient, however, to compute k inside the model.

The most intricate part of the investment theory involves the way expected rates of return and investment are related.

In simplified notation, expected rates of return are given by

$$EROR_i = EEQROR_i + DIS_i \tag{5.90}$$

where

$EROR_i$ is the expected rate of return[1] in year t to owners of industry i's capital;

$EEQROR_i$ is the equilibrium expected rate of return, i.e., the expected rate of return required to sustain indefinitely the current rate of capital growth in industry i; and

DIS_i is a measure of the disequilibrium in i's current expected rate of return.

The equilibrium expected rate of return is specified as an inverse logistic function:

$$\begin{aligned} EEQROR_i = {} & \{RORN_i + F\_EEQROR\_I_i + F\_EEQROR\} \\ & + (1/C_i)*[\ln(K\_GR_i - K\_GR\_MIN_i) - \ln(K\_GR\_MAX_i - K\_GR_i) \\ & - \ln(TREND\_K_i - K\_GR\_MIN_i) + \ln(K\_GR\_MAX_i - TREND\_K_i)] \end{aligned} \tag{5.91}$$

where

K_GR_i is the rate of growth of capital in industry i through year t, that is,

$$K\_GR_i = \left(\frac{K_{i,t+1}}{K_{it}} - 1\right);$$

$K\_GR\_MIN_i$ is the minimum possible rate of growth of capital and is set at the negative of the rate of depreciation in industry i;

[1] What we mean by j's expected rate of return is defined precisely in the next subsection.

$TREND_i$ is the industry's historically normal capital growth rate. This is an observed growth rate in capital over an historical period;

$K\_GR\_MAX_i$ is the maximum feasible rate of capital growth in industry i. It is calculated by adding a difference, DIFF, to TRENDi. For example, if DIFF has been set at 0.06, and the historically normal rate of capital growth in an industry is 3 per cent, then we impose an upper limit on its simulated capital growth in any year t of 9 per cent;

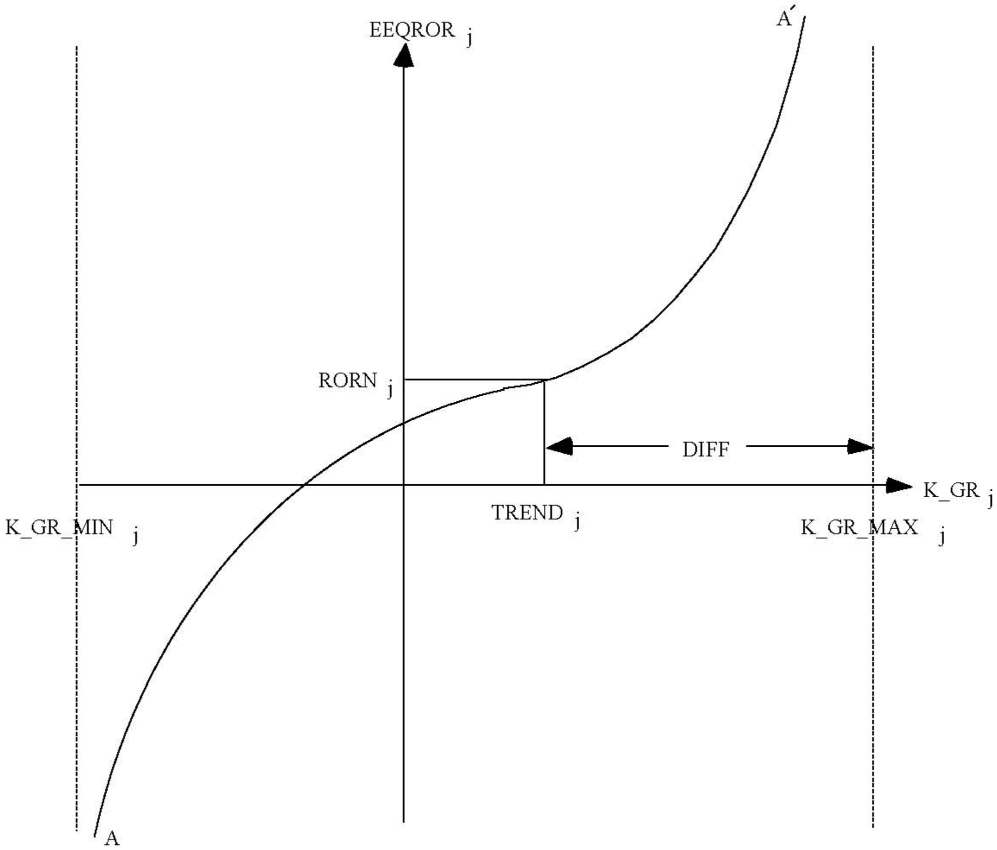
$RORN_i$ is the industry's historically normal rate of return. For each industry j, RORNi is an estimate of the average rate of return that applied over the historical period in which the industry's average annual rate of capital growth was TRENDi;

C_i is a positive parameter that controls the sensitivity of the equilibrium expected rate of return to variations in its capital growth (and consequently the sensitivity of i's capital growth to variations in its equilibrium expected rate of return). The value of this parameter is chosen so that the responses of investment to changes in rates of growth are realistic.

The inverse logistic function is illustrated in figure 5.1. For understanding (5.71) and (5.72), it is helpful to start by assuming that $F\_EEQROR\_J_i$, F_EEQROR and DIS_i are all fixed at zero. Then (5.71) and (5.72) mean that for industry i to attract sufficient investment in year t to achieve a capital growth rate of $TREND\_K_i$, it must have an expected rate of return of $RORN_i$. For the industry to attract sufficient investment in year t for its capital growth to exceed $TREND\_K_i$, its expected rate of return must be greater than $RORN_i$. Similarly, if the expected rate of return in the industry is less than that observed in the historical period, then provided that there is no disequilibrium, (5.71) and (5.72) imply that investors will restrict their supply of capital to the industry to below the level required to generate capital growth at the historically observed rate.

An important characteristic of the logistic equation is that the shift variables on the RHS of (5.72) allow for vertical shifts in the capital supply curves, the AA′ curves in Figure 5.4. Being able to move the AA′ curves is useful in forecasting and historical simulations. In these simulations we often have information from outside the model on either investment by industry or aggregate investment.

Figure 5.4 The equilibrium expected rate of return schedule for industry j



We assume that this disequilibrium disappears over time according to the schedule:

$$DISEQ_{i} = (1 - i_{j})*DISEQ\_B_{i} , \qquad (5.92)$$

where DISEQj and DISEQ_Bj are the gaps between industry j's expected rate of return and the industry's expected equilibrium rate of return in the current year and in the data year (t–1), and Φ_j is a parameter with a value between 0 and 1.

5.3.4.2 Actual and expected rates of return under static and forward-looking expectations

The investment dynamics in VERM can be run either under static expectations or under forward-looking expectations. The latter are important when evaluating policies that involve, say, delayed tax increases to compensate for present tax cuts. However, for many other applications forward-looking expectations may not be relevant.

We next derive the rate of return on investment. To start with, the present value (PV) of purchasing in year t a unit of physical capital for use in industry i is given by:

$$PV_{i,t} = -\Pi_{i,t} + [Q_{i,t+1}*(1-T_{t+1}) + \Pi_{i,t+1}*(1-D_i) + T_{t+1}*\Pi_{i,t+1}*D_i]/[1 + INT_t*(1-T_{t+1})] \quad (5.93)$$

where

$\Pi i,t$ is the cost of buying or constructing in year t a unit of capital for use in industry j;

Di is the rate of depreciation;

Qi,t is the rental rate on i's capital in year t, i.e., the cost of using a unit of capital in year t;

Tt is the income-tax rate in year t;

and

$INTt$ is the nominal rate of interest in year t.

In (5.74), we assume that units of capital bought or constructed in year t yield to their owners three benefits in year t+1. First, they generate rentals with a post-tax value of $Q_{i,t+1}*(1-T_{t+1})$. Second, they can be sold at the depreciated value of $\Pi_{i,t+1}*(1-Di)$. Third, they give a tax deduction. We assume that this is calculated by applying the tax rate (T_{t+1}) to the value of depreciation ($\Pi i_{,t+1}*D_j$). To obtain the present value (value in year t) of these three benefits, we discount by one plus the tax-adjusted interest rate $[INT_t*(1-T_{t+1})]$.

Equation (5.74) is converted to a rate of return formula by dividing both sides by $\Pi_{i,t}$, i.e., we define the actual[6] rate of return, $ROR\_ACT_{i,t}$, in year t on physical capital in industry j as the present value of an investment of one euro. This gives

$$ROR\_ACT_{i,t} = -1 + [(1-T_{t+1})*Q_{i,t+1}/\Pi_{i,t} + (1-D_i)*\Pi_{i,t+1}/\Pi_{i,t} + T_{t+1}*D_i*\Pi_{i,t+1}/\Pi_{i,t}]/[1 + INT_t*(1-T_{t+1})]. \quad (5.94)$$

As we saw in the previous subsection, the determination of capital growth and investment depends on expected (rather than actual) rates of return. In most simulations, we assume that capital growth and investment in year t depend on expectations held in year t concerning $ROR\_ACT_{i,t}$.

Under static expectations, we assume that investors expect no change in the tax rate (i.e., they expect T_{t+1} will be the same as T_t) and that rental rates (Q_i) and asset prices (Π_i) will increase by the current rate of inflation (INF). Under these assumptions, their expectation of $ROR\_ACT_{i,t}$ is

$$ROR\_SE_{i,t} = -1 + [(1-T_t)*Q_{i,t}/\Pi_{i,t} + (1-D_i) + T_t*D_i]/(1+R\_INT\_PT\_SE_t), \quad (5.95)$$

where ROR-SEj,t is the expected rate of return on capital in industry j in year t under static expectations, and $R\_INT\_PT\_SE_t$ is the static expectation of the real post-tax interest rate, defined by

$$1 + R\_INT\_PT\_SE_t = [1 + INT_t*(1-T_t)]/[1+INF_t] . \quad (5.96)$$

Under forward-looking or rational expectations, we assume that investors correctly anticipate actual rates of returns, i.e., their expectation of $ROR\_ACT_{i,t}$ is $ROR\_ACT_{i,t}$. This implies that we need to set i's expected rate of return equal to i's actual rate of return in a year-t simulation. The difficulty is that i's actual rate of return in year t depends on future rentals ($Qi_{,t+1}$), future tax rates (T_{t+1}) and future asset prices ($\Pi i_{,t+1}$). In the sequential approach to computing VERM solutions, the values of variables in year t+1 cannot normally be known in the computation for year t. This problem is solved by adopting an algorithmic approach.

In the first iteration of the algorithm used for solving VERM, we compute solutions for years 1 to T under the assumption of static expectations. Thus, if are happy to assume static expectations, we require only one iteration. However, if we wish to assume forward-looking expectations, then we will usually need further iterations (i.e., further calculations of solutions for years 1 to T). This is because the expected rates of return assumed for year t [$ROR\_SE_{i,t}$] are unlikely to equal the actual rates of return [$ROR\_ACT_{i,t}^1$] implied in the first iteration by the solutions for years t and t+1.

For the final year (T), we do not generate information on future values of variables. We assume that industry j's actual rate of return in year T [$ROR\_ACT_{i,T}^1$] is the same as that in year T-1 [$ROR\_ACT_{i,T-1}^1$].

In the second iteration, we assume that the expected rates of return in years 0 to T are the actual rates of return calculated from the first iteration, i.e.,

$$EROR_{i,t}^{2} = ROR\_ACT_{i,t}^{1},\ t = 0, \ldots T. \tag{5.97}$$

From the resulting solutions for years 1 to T and the data for year 0, we compute the implied actual rates of return, $ROR\_ACT_{i,t}^{2}$, t=0, …T-1. As in the first iteration, we assume that the actual rates of return in the final year are equal to the actual rates of return in the second last year, i.e.,

$$ROR\_ACT_{i,T}^{2} = ROR\_ACT_{i,T-1}^{2}\ . \tag{5.98}$$

For the third and subsequent iterations, we adjust the expected rates of return according to

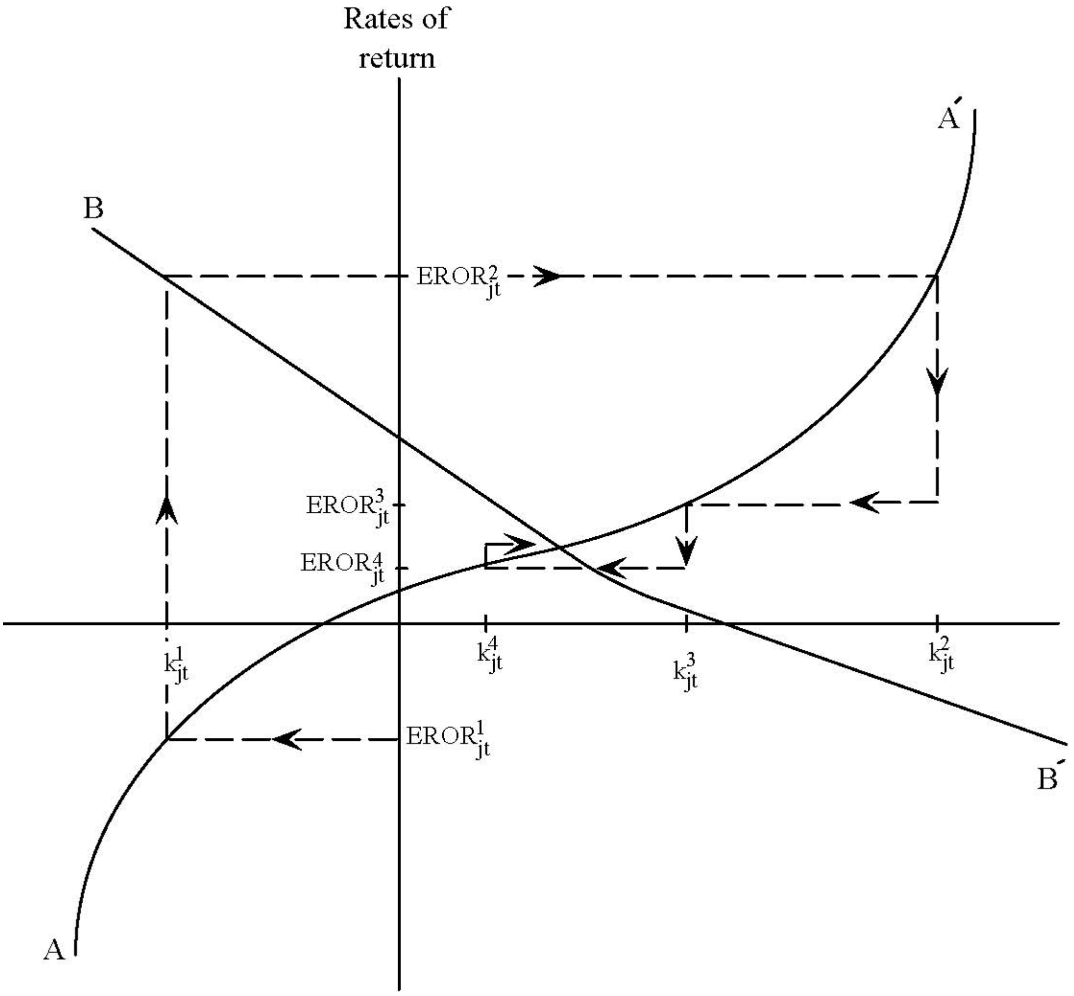
$$EROR_{i,t}^{n} = EROR_{i,t}^{n-1} + ADI\_RE_{i} *(ROR\_ACT_{i,t}^{n-1} - EROR_{i,t}^{n-1}),$$
$$\text{for } n>2,\ t=0,\ldots,T,^{7} \text{ and i belongs to IND,} \tag{5.99}$$

where ADJ_RE_{i} is a parameter set between 0 and 1.

Convergence is achieved when

$$EROR_{i,t}^{n} = ROR\_ACT_{i,t}^{n} \text{ for all i and t.} \tag{5.100}$$

Figure 5.5 Convergence of the algorithm for imposing forward-looking expectations



As in Figure 5.5., we assume that there is no disequilibrium in expected rates of return. Thus, we assume that the VERM outcomes for expected rates of return and rates of capital growth in year t in industry j are on the AA′ schedule. We also assume that VERM outcomes for actual rates of return and rates of capital growth are on the BB′ schedule. In drawing BB′ we have in mind the capital demand schedule for year t+1 which, other things being equal, implies a negative relationship between the availability of physical capital to industry j in year t+1 and its rental rate in year t+1, and thus a negative relationship between capital growth in year t and the actual rate of return in year t. In VERM computations, BB′ moves between iterations and we do not necessarily operate on AA′. Nevertheless, we find Figure 5.5. a useful device for thinking about the convergence of our algorithm. For example, with the AA′ and BB′ curves in our diagram, convergence is very rapid when ADJ_REj is set at 0.5 (the illustrated case). If ADJ_REj is set at 1.0, then readers will find, after a little experimenting with the diagram, that the algorithm may become stuck in a non-converging cycle, or converge very slowly.

In VERM code, the logistic equation (5.72) is given by

```
Equation E_d_f_eeqror
   # Change in equilibrium expected rate of return in forecast
year #
(all,d,DST)(all,i,IND) d_eeqror(d,i) = [1/COEFF_SL(d,i)]*
    [1/[K_GR(d,i)-K_GR_MIN(d,i)]+1/[K_GR_MAX(d,i)-
K_GR(d,i)]]*d_k_gr(d,i)
                        + d_f_eeqror_i(d,i) +
d_f_eeqror(d)+d_f_eeqror_d(i);                        (5.101)
```

where d_eeqror(i) is the expected change in the equilibrium rate of return for industry I, and d_f_eeqror(i) and d_f_eeqror are shifters that affect capital growth.

The expected change of the rates of return, corresponding to (5.101), is given by

```
Equation E_d_eeqror
  # Expected ror equals equil. expec. ror plus disequilibrium
in expec. ror #
(all,d,DST)(all,i,IND) d_eror(d,i) = d_eeqror(d,i) +
d_diseq(d,i);                                         (5.102)
```

where d_diseq(i) gives the disequilibrium in expectations. The disequilibrium arises, since usually the data for year t-1 (either observed or the final t-1 simulated solution) for expected rates of return and for capital growth in industry j will not usually give a point on i's AA′ curve (in figure 5.4.). Consequently, in our data for year t-1, DISEQj will normally be non-zero.

In year-to-year simulations we usually assume that this disequilibrium is eliminated over time. If the expected rate of return in industry j was initially high relative to j's rate of capital growth (DISEQi > 0) then for any given change in the expected rate of return [d_eror(i)], the elimination of DISEQi will (via E_d_eeqror) increase d_eeqror(i) and hence (via E_d_f_eeqror_i) increase i's rate of capital growth [d_k_gr(i)]. Thus we tend to forecast an increased rate of capital growth for any industry currently experiencing a lower growth rate than would be anticipated in light of its expected rate of return and its AA′ curve.

The elimination of disequilibrium is achieved according to:

either

$DISEQ_i = (1 - \Phi_i)*DISEQSE\_B_i$, (5.103a)

or

$DISEQ_i = (1 - \Phi_j)*DISEQRE\_B_i$, (5.103b)

where

$DISEQ_i$ is the value for $DISEQ_i$ in the current year (t);

$DISEQSE\_B_i$ and $DISEQRE\_B_i$ are alternative values for $DISEQ_i$ in the data (t–1);

and

Φ_i is a parameter with a value between 0 and 1, usually set at 0.5.

The alternative measures of $DISEQ_i$ reflect alternative definitions of expected rates of return. $DISEQSE\_B_i$ is used for static expectations and $DISEQRE\_B_i$ for forward-looking expectations.

From the view point of the TABLO code, it is useful to express (5.84a) as

$$DISEQ_i - DISEQSE\_B_i = - \Phi_i*DISEQSE\_B_i*UNITY \qquad (5.104)$$

In our year t computation, $DISEQSE\_B_i$ is treated as a parameter and $DISEQ_i$ and UNITY are variables. Under static expectations, (5.85) is satisfied by the initial solution for year t because in that solution we assume that $DISEQ_i$ equals $DISEQSE\_B_j$ and that UNITY equals zero. In the required or final solution for year t, UNITY must be one to allow the final value of $DISEQ_i$ to satisfy (5.75a). Thus, if we are assuming static expectations, we evaluate the change in $DISEQ_i$ in year t (d_diseq_i) from its initial value ($DISEQSE\_B_i$) according to the equation:

$$d\_diseq_i = -\Phi_i*DISEQSE\_B_i*d\_unity \quad , \qquad (5.105)$$

where d_unity is set exogenously at 1. In slightly different notation and with the inclusion of a shift variable, (5.86a) appears in the TABLO code as E_d_diseq.

```
Equation E_d_diseq
  # Gives shock to disequil. in s.e. rors, moves them towards
zero #
(all,d,DST)(all,i,IND) d_diseq(d,i) =
    - ADJ_COEFF(d,i)*DISEQSE_B(d,i)*d_unity + d_f_diseq(d,i);
```
(5.106)

This completes the description of investment dynamics. We next turn to lagged real wage adjustment.

5.3.5 Labour markets and wage rigidities

(Section 8.17. in the VERM code)

While it is customary to assume perfectly competitive labour markets in comparative static simulations, in dynamic simulations this assumption may be less satisfactory. VERM contains an alternative theory for wage setting over time that allows for gradual adjustment in the labour markets as a response to policy shocks. This theory can be motivated by Layard-Nickell type monopoly unions, and parameters for the kind of centralised wage setting the theory implies have also been estimated for Finland.

The theory assumes that real wages adjust sluggishly over time, which implies that, in the short run, labour market adjustment takes place via employment. In the code, the real wage equation is given by

```
Equation E_del_f_wage_pt
# Relates deviation in CPI-deflated post-tax wage to deviat.
in employment #
(RWAGE_PT/RWAGE_PT_OLD)*(real_wage_pt - real_wage_pt_o)
    =    100 * ((RWAGE_PT_B/RWAGE_PT_O_B)
    -   (RWAGE_PT_L_B/RW_PT_O_L_B))*d_unity
    +    ALPHA1*(EMPLOY/EMPLOY_OLD)*(employ_i - employ_i_o)
    -    100*ALPHA1*((RWAGE_PT_B/RWAGE_PT_O_B)^ALPHA2
    -   (RWAGE_PT_L_B/RW_PT_O_L_B)^ALPHA2)*d_unity
    +    del_f_wage_pt;
```
(5.107)

where RWAGE_PT gives the actual, post tax real wage in year t, and RWAGE_PT_OLD gives the baseline forecast for the post-tax real wage in year t, while real_wage_pt and real_wage_pt_o give the percentage changes in them. The theory allows for the possibility that wage setting is affected by changes in employment as well, which is captured by the terms involving the coefficients EMPLOY and EMPLOY_OLD and their respective percentage changes. The coefficients ALPHA1 and ALPHA2 are parameters determined outside the model that control the speed of adjustment. With ALPHA2 set to zero, employment tends to return to its forecast growth path, whereas with a non-zero ALPHA2, employment may change from forecast even in the long run.

It is important to realise that the wage mechanism is not in use in the forecast simulation. Instead, the forecast values of employment and wages take place are obtained from the forecast simulation and used in the policy simulations. This necessitates a re-run of the model using policy closures. During the rerun of the forecast, the predicted baseline values of these variables are stored in the database, to be used in the dynamic equations in the policy simulation. Finally, it may be worth emphasising that when the model is run under rational expectations, these forecasts are model consistent.

5.4 Reporting variables and miscellanea

The final section of VERM code contains useful reporting variables, as well as some equations that can be used to link taxes to each other. Greenhouse gas accounting is also included among the miscellaneous equations.

The miscellanea in VERM code contain some indexes not elsewhere defined, but also a useful device for forecasting that allows us to introduce forecasts based on more highly aggregated data than is otherwise used. These equations do not introduce any new theory to the model.

References

Alho, K. (2002): The Equilibrium Rate of Unemployment and Policies to lower It: The Case of Finland. Keskustelualoite 839, ETLA.

Badri, N. G. – Walmsley, T.L. (eds.) (2008): Global Trade, Assistance and Production: The GTAP 7 Data Base. Center for Global Trade Analysis, Purdue University.

Bank of Finland (2002–2008): Balance of Payments Accounts.

Bovenberg, A. L. – Goulder, L. H. (1991): Introducing Intertemporal and Open Economy Features in Applied General Equilibrium Models. H. Don – T. van de Klundert – J. van Sinderen (Eds.): Applied General Equilibrium Modeling. Dordrecht: Kluwer Academic Publishers; 47–64.

Dervis, K. – de Melo, J. – Robinson, S. (1982): General Equilibrium Models for Development Policy, Cambridge University Press, Cambridge, U.K.

Dixon, P.B. – Parmenter, B.R. – Powell, A.A. – Wilcoxen, P.J. (1992): Notes and Problems in Applied General Equilibrium Economics. North-Holland, Amsterdam.

Dixon, P. – Rimmer, M. (2002): Dynamic General Equilibrium Modelling for Forecasting and Policy. Contributions to Economic Analysis 256. North-Holland Publishing Company, Amsterdam.

ECFIN (2006): The impact of ageing on public expenditure: projections for the EU25 Member States on pensions, health care, long term care, education and unemployment transfers (2004–2050). Special report 1/2006. European Commission. Directorate-General for Economic and Financial Affairs, Brussels.

Eurostat (1997): Euroopan kansantalouden tilinpito. EKT95. Euroopan yhteisöjen virallisten julkaisujen toimisto, Luxemburg.

Eurostat (2003): Handbook on Social Accounting Matrices and Labour Accounts, European Commission, Leadership Group SAM, Population and Social Conditions Report 3/2003/E/Numero 23, Luxenburg.

Harrison, W.J. – Pearson, K.R. (2002): GEMPACK user documentation for version 8. Centre of Policy Studies and Impact Project. Monash University, Melbourne.

Harrison, W.J. – Pearson, K.R. (2005): GEMPACK user documentation for version 9. Policy Studies and Impact Project. Monash University, Melbourne.

Honkatukia, J. (2009): VATTAGE – A Dynamic, Applied General Equilibrium Model of The Finnish Economy. Research report 147, VATT, Helsinki.

Honkatukia, J . – Ahokas, J. – Kinnunen, J. (2011): Ikääntymisen vaikutukset taloudelliseen kehitykseen Suomen maakunnissa ja Keski-Suomen seutukunnissa - kuinka vastata kuntatalouden menopaineiden kasvuun? VATT-tutkimuksia 167, VATT, Helsinki.

Honkatukia, J. – Simola, A. (2011): Selvitys Suomen nykyisestä ja tulevasta puunkäytöstä. VATT-tutkimuksia 164, VATT, Helsinki.

Horridge, M. (2011): The TERM model and its data base. Centre of Policy Studies, Monash University. General Paper No. G-219 July 2011.

Jalava, J. – Pohjola, M. – Ripatti, A. – Vilmunen, J. (2005): Biased Technical Change and Capital-Labor Substitution in Finland, 1902–2003. The B.E. Journal of Macroeconomics. Topics, Vol 6, Issue 1, Article 8.

Johansen L. (1960): A Multi-Sector Study of Economic Growth. North-Holland, Amsterdam.

McMorrow, K.C. – Roeger, R. (2000): Time-Varying Nairu/Nawru. Estimates for the EU's Member States. Economic papers 145, ECFIN.

Scarf, H.E. (1973): The Computation of Economic Equilibria. Yale University Press, New Haven/London.

Scarf, H.E. – Shoven, J.B. (eds.) (1984): Applied General Equilibrium Analysis. Cambridge University Press, New York.

Shoven, J.B. – Whalley, J. (1972): A general equilibrium calculation of the effects of differential taxation of income from capital in the U.S. Journal of Public Economics 1; 281–321.

Shoven, J.B. – Whalley, J. (1973): General equilibrium with taxes: A computational procedure and an existence proof. Review of Economic Studies 40; 475–489.

Shoven, J.B. – Whalley, J. (1974): On the computation of competitive equilibrium in international markets with tariffs. Journal of International Economics 4; 341–354.

Shoven, J.B. – Whalley, J. (1984): Applied General-Equilibrium Models of Taxation and International Trade: An Introduction and Survey. Journal of Economic Literature 22; 1007–1051.

Shoven, J.B. – Whalley, J. (1992): Applying General Equilibrium, Cambridge University Press, New York.

Tilastokeskus (2000): Sektoriluokitus 2000. Käsikirjoja 5, Tilastokeskus. Helsinki.

Tilastokeskus (2009): Kansantalouden tilinpito.

Valtiovarainministeriö. (2007): Suomen vakausohjelman tarkistus. Taloudelliset ja talouspoliittiset katsaukset 4a/2007.

NACE (Nomenclature Générale des Activités Economiques dans les Communanautés Européennes) is statistical industry classification used in European Union.
ISIC (International Standard Industrial Classification of All Economic Activities) is a statistical industry classification confirmed by United Nations.
CPA (Statistical classification of products by activity in the European Economic Community) is used by the European Union in national and regional accounts for input-output analysis.

VALTION TALOUDELLINEN TUTKIMUSKESKUS
STATENS EKONOMISKA FORSKNINGSCENTRAL
GOVERNMENT INSTITUTE FOR ECONOMIC RESEARCH

Valtion taloudellinen tutkimuskeskus
Government Institute for Economic Research
P.O.Box 1279
FI-00101 Helsinki
Finland

www.vatt.fi

ISBN 978-952-274-073-1 (PDF)
ISSN 1795-3340