

The MANAGE-WB Applied General Equilibrium Model of the World Bank

Model Documentation and User Guide

Lulit Mitik Beyene

Wolfgang Britz

Martin Christensen

Hasan Dudu

Ragchaasuren Galindev



WORLD BANK GROUP

Economic Policy Global Department

November 2025

Abstract

MANAGE-WB is a Computable General Equilibrium (CGE) model developed by a network of CGE modelers to support World Bank teams and clients in conducting macroeconomic analyses across a broad range of topics. It is a single-country, open-economy CGE model featuring multiple sectors, institutions, and factors of production.

This paper presents the model's structure and implementation, including its graphical user interface, methods for estimating and updating Social Accounting Matrices (SAMs), as well as tools for conducting sensitivity analyses and stochastic simulations.

This paper is a product of the Economic Policy Global Department. It is part of a larger effort by the World Bank to provide open access to its research and make a contribution to development policy discussions around the world. Policy Research Working Papers are also posted on the Web at <http://www.worldbank.org/prwp>. The authors may be contacted at rgalindev@worldbank.org.

The Policy Research Working Paper Series disseminates the findings of work in progress to encourage the exchange of ideas about development issues. An objective of the series is to get the findings out quickly, even if the presentations are less than fully polished. The papers carry the names of the authors and should be cited accordingly. The findings, interpretations, and conclusions expressed in this paper are entirely those of the authors. They do not necessarily represent the views of the International Bank for Reconstruction and Development/World Bank and its affiliated organizations, or those of the Executive Directors of the World Bank or the governments they represent.

The MANAGE-WB Applied General Equilibrium Model of the World Bank: Model documentation and user guide¹

Lulit Mitik Beyene, Wolfgang Britz, Martin Christensen, Hasan Dudu, Ragchaasuren Galindev[†]

Authorized for distribution by Frederico Gil Sander, Practice Manager, Economic Policy Global Department, World Bank Group

JEL codes: C63, C68

Key words: Computable General Equilibrium Modeling

¹ This documentation is based on the documentation of the MANAGE v2f and MANAGE-WB v3 with updates to equations and model structure.

[†] Corresponding author: rgalindev@worldbank.org

Content

1. Introduction.....	5
Overview.....	5
MANAGE-WB in a nutshell	6
2. Model specifications	8
Production block.....	10
Production functions	10
Multi-input and multi-output specifications	17
Income block.....	19
Factor income.....	19
Households	19
Government.....	21
Enterprises	24
Transfers	24
Domestic fund markets.....	25
New foreign government debt.....	27
Government borrowing	28
Debt stocks and savings.....	28
Demographic equations.....	29
Demand block.....	30
Household demand.....	30
Other final demand for bundles: government and investment.....	31
Demand for trade and transport margins	33
Stock change behavior.....	34
Armington decomposition: Import and domestic demand	35
Tax rates and tax-inclusive Armington price.....	36
Export supply and domestic sales, import and export prices.....	38
Factor market closure.....	40
Flexible nesting	40
Capital markets without vintage capital specification	43

Capital markets with the vintage capital specification	43
Endogenous technical progress	45
Foreign savings and balance of payments.....	48
Macro identities.....	50
Default closure rules.....	53
Emissions and land cover module.....	54
Climate change damage module	56
Air pollution emissions and related damages	57
Model Dynamics	57
Recursive dynamics versus limited forward-looking behavior.....	58
3. Model Implementation	59
Folder and File Structure.....	62
Main Files	67
Data input files.....	67
Include files	76
Main settings	76
Files for benchmarking.....	77
Recursive Dynamic files	78
Model Files.....	79
Files used to solve the model.....	79
Reporting Files	80
Simulation Files.....	80
4. Graphical User Interface	81
Introduction	81
Baseline generation	81
Overview	81
GUI controls in baseline mode.....	84
Simulations.....	86
Overview on simulation	86
GUI controls for simulation	87
Extended options on user interface.....	88

CCDR simulation driver	92
Background	92
Mitigation policies	95
Power mix and electricity price fixing.....	97
IEC damages.....	99
Energy projections	101
Reporting back-end.....	104
Concept	104
Using the reporting back-end and available reports	105
5. Social Accounting Matrix: Estimation and Update.....	108
5.1 Constructing SAMs for MANAGE-WB from the GTAP data base	108
5.2. Updating SAMs to macro-totals, balancing and splitting	110
5.2.1 GUI choices	110
5.2.3 Main processing steps.....	111
5.3.2 Exploring the results.....	116
6. Sensitivity analysis	118
7. Stochastic simulation runs.....	121
References.....	127

1. Introduction

Overview

MANAGE-WB has been developed and used by a network of CGE modelers to support World Bank teams and clients in macroeconomic analyses of a wide range of topics. In recent years, it has been extended in a modular fashion to better capture emissions and their abatement, climate change damages and related adaptation measures as well as features of energy markets to support macroeconomic analysis in Country Climate and Development Reports as well as Long Term Greening Strategy documents, and to better capture features of labor, capital and land markets. In its default layout, it takes the standard assumptions found in most single country CGE models: aggregate firms minimize costs under constant return-to-scale technologies, households maximize utility, economic agents own the production factors and supply them to firms, all agents are price takers in perfectly competitive markets for products and production factors.

MANAGE-WB is a recursive-dynamic model running in yearly steps, using by and large the neo-classical growth specifications – i.e., labor growth is exogenous, and capital accumulation derives from savings/investment decisions. Savings and investments are endogenously determined in bond markets where the interest rate on government debt and returns on private investment clears supply and demand of loanable funds. Optionally, a version with forward looking behavior over multiple periods captures endogenous stock changes and bases savings and investment decisions on expected returns and interest rates. The model allows for a wide range of productivity assumptions that include autonomous improvements, for instance, in energy efficiency, that can differ across agents and energy carriers. It also features levers to integrate results from sectoral models such as energy, transport or biophysical ones. These levers comprise endogenous CES twist parameters and endogenous productivity and demand shifters. Most parameters are defined as variables to enable high flexibility in such closure swaps. Finally, the model can incorporate an on-demand vintage structure for capital that allows for putty/semi-putty assumptions about sluggish mobility of installed capital which can be extended to a version with endogenous technological progress under monopolistic competition.

MANAGE-WB is a template model which can be calibrated to any Social Accounting Matrices (SAM) which adheres to a set of conventions on how to represent the economic structure such that sector, product, factor or tax details differ across projects. A larger set of pre-defined closure rules ease a project specific model application. The model is implemented in GAMS (General Algebraic Modeling System), a higher-level Algebraic Modeling Language and uses the GGIG (GAMS Graphical User Interface Generator) framework for its Graphical User Interface and reporting tools. It can be solved as a Mixed-Complementarity Problem (MCP) to capture regime switches, such as emission taxes becoming zero when the emission cap is not binding. Ready to use code allows for systematic sensitivity analysis using Latin-Hypercube based stratified random sampling with multiple runs executed in parallel, or to summarize a large set of Monte-Carlo draws for shocks with scenario reductions tools to a manageable set of stochastic runs, again executed in parallel. Routines allow estimating a balanced SAM using country specific data when available, the GTAP database or a mix of both. The code of MANAGE-WB is structured such as to easily use a single model installation

simultaneously in multiple projects. Flexible functional specifications allow the analysts to easily implement an appropriate nesting structure for the production functions, final demand and factor supplies, to depict substitution across products and segmentation of factor markets.

MANAGE-WB in a nutshell

The MANAGE-WB model is a recursive dynamic Computable General Equilibrium (CGE) model. In its default layout, the model is solved for one year to define a simultaneous static equilibrium in all commodity and factor markets. The equilibrium reflects certain dynamics such as population growth and capital accumulation or productivity changes between two consecutive years. Agents have perfect foresight for the current period but are myopic in the sense that they do not base their decisions on expectations of the future. Product and factor markets are competitive. Each static equilibrium relies on a relatively standard set of assumptions in CGE modeling as discussed next.

Production is depicted by several aggregated firms (or sectors, typically multiple dozen) which minimize costs under constant-returns-to-scale-technologies. The related production functions are modeled by trees of nested constant-elasticity-of-substitution (CES) functions to capture the substitutions and complements across the different inputs such as capital, labor and intermediates, and consider taxes on factors, intermediates and outputs, including taxation of emissions related to product or factor use, or output levels. This production function nesting can be easily adjusted according to database detail and the focus of the analysis. For instance, in climate change related analysis, different energy carriers act as substitutes, while total demand for energy is a complement with capital. An optional structure captures semi-putty/putty relations across inputs with more elastic long-run behavior. For instance, in the short run, energy is a near-complement with capital in the short-run but a substitute in the long-run. Accordingly, rising energy prices tend to lead to larger increases in production costs in the short-run when substitution is weak, but long-run responses encompass investing into energy-saving technologies that dampen the cost-push factor.

The model allows for both multi-input and multi-output production. Multi-input depicts, for instance, that electricity is produced by multiple power generation activities. Multi-output allows for a single activity to produce more than one product, for instance, oil seed crushing outputting both vegetable oils and oil cakes.

Factor incomes are largely allocated to households with enterprises as pass-through accounts. Enterprises take saving decisions and transfer the remaining income to households, to the Rest-of-the World to consider foreign factor ownership, and to the government to depict (partially) state owned enterprises. Enterprise factor income can be subjected to taxation. Government revenue is derived from a detailed list of both direct and indirect taxes as well as from transfers and factor income from ownership of enterprises. Household demand is modeled using the constant-differences-in-elasticity (CDE) demand function, which is the standard utility function used in the GTAP model. Final demanders consume commodity bundles composed of supplied goods or CES composite thereof, based on a completely flexible nested CES approach, such that the composition of bundles is price dependent. For instance, a part of fuel demand can be combined with demand for different transport services and vehicles to define a transport demand bundle while the remainder of fuel demand, related to heating, could be combined with other products to form a housing service bundle.

Goods are evaluated at basic prices with tax wedges. The model can incorporate trade and transport margins such that basic and end-user prices before taxation differ. These margins are differentiated across transport nodes, i.e., from farm/factory gate to domestic buyers or to the border for exports, and from border to end-user for imports.

Import demand is driven by the ubiquitous Armington assumption which considers quality differences between the imported and domestic origin. Implemented by a CES function, the level of the (often called Armington) elasticity determines the degree of substitutability across domestic sales and imports. Domestic production is analogously distributed to exports and domestic sales based on a CET (Constant-Elasticity-of-Transformation) function where the ability of producers to switch between domestic and foreign markets is determined by the level of the transformation elasticity. All CES and CET functions can employ the so-called additive version to ensure aggregation in physical volumes. The decision where physical aggregation is used rests with the analyst, it is for instance used regularly for land supply and often also for the distribution of energy demands. As an alternative to the CES or CET case, the model can treat imports and exports for selected markets as perfect substitutes to domestically produced output such that the law-of-one price holds.

Market equilibrium for domestic output sold domestically is assumed through market-clearing prices. The small open economy assumption implies exogenous export and import world prices before margins and taxes. Alternatively, the model can use iso-elastic functions for selected markets to render these prices depending on import respectively export levels.

The model assumes market-clearing wages in the labor markets with the possibility of an upward sloping labor supply schedule and sluggish mobility of labor across multiple nested labor market segments, such as agricultural versus non-agricultural sectors. Alternatively, fully elastic labor supply at fixed prices can be assumed.

In dynamic simulations using the vintage mechanism, *new* capital is generated by current year investments and allocated across sectors to equalize the rate of return, i.e., assumed fully mobile. *Old* capital, considering depreciation, is largely considered immobile and remains in use in its original sector unless the sector is in decline. In this case, a sector does not attract new capital and only the old vintage is active. Some of its *old* capital can be sold based on an upward sloping supply schedule depending on the sector's returns-to-capital, such that capital returns in declining sectors drop below the economy-wide average. The *old* capital in expanding sectors earns the same rate of return as the *new* capital. The vintage mechanism can be combined with an endogenous technological progress extension where firms charge mark-ups under monopolistic competition. The sector specific markup income finances R&D activities which produce new, more productive varieties such that fast-growing sectors benefit from higher technical progress.

This vintage mechanism can also be removed and capital movement across sectors can then be depicted by sluggish or perfect factor mobility, also using segmented markets. The same flexibility exists for markets for land, for instance to depict land cover in multiple Agri-Environmental Zones, while natural resources such as mines, oil fields or fish stocks are considered sector specific and thus immobile.

The dynamics of MANAGE-WB are composed of three major elements. Population and labor stock growth are exogenous where the latter is often equated to the growth of the working-age population. The aggregate capital stock grows according to the overall level of total savings (enterprises, households, public and foreign), considering depreciation and shares of non-productive investments. Investments are equal to domestic and foreign savings net of new government debt. The third component relies on productivity assumptions where usually labor productivity is endogenous during baseline generation to replicate a given growth path of GDP. As an option, labor productivity growth differences between the primary, secondary and third sector can be driven by empirical relations depending on GDP growth, while productivity changes for capital, land and natural resources can be linked partially or fully to changes in labor productivity. Additional endogenous specific productivity shifters or update of share parameters are often employed to superimpose specific developments in baseline construction or counterfactuals, such as changes in the energy mix.

2. Model specifications

MANAGE-WB is a template model such that the model's code is independent of the details of the SAM and other data used as inputs. Model equations and data transformations for benchmarking, dynamics, solving for a new equilibrium and reporting provide a structural template which represents the concepts and methodology underlying the model. An actual implementation of the equations and data transformations in a project reflects the data details as defined by the main sets and subsets of the model as shown in Table 2.1, and the provided data and model parameters. A well-documented and structured EXCEL workbook comprises all project specific data, parameters, sets and maps and acts as a bridge between the specific project and the generic model template. Additional coding for a project mainly implements the code for the shocks of specific counterfactuals.

The input SAM has a set (labels and long-texts) of accounts denoted by is from which subsets are derived. The dimension of the SAM is hence $is \times is$. Sectors have three classifications: a , i , and k , where a are (production) activities, i are marketed commodities and k are consumed commodity bundles, respectively. In a more traditional CGE model, the three sets are identical, i.e., each activity outputs one product not produced by others, and these products enter without being bundled the final demand functions. In the MANAGE-WB model, with its multi-input and multi-output production structure, the output of a product from potentially multiple activities (a) (also called the make matrix) solved domestically together with imports define domestic market appearances of product (i). This allows, for example, to have multiple activities producing a single commodity (for example, electricity from nuclear, wind, fossils) and to have single activities producing multiple commodities (e.g., the vegetable oil and cake sector producing oils and cakes as differentiated outputs). In addition, MANAGE-WB allows for commodities (and/or bundles) in final demand (indexed by k) to differ from marketed commodities (i). A consumer-based 'make' or transition matrix assigns supplied commodities in shares to multi-level demand nests, with price-dependent shares. This allows for more realistic demand behavior, for instance to model demand for energy carriers. Household demand for transportation services can be a combination of demand for automobiles and (a part of the) demand for fuel. If the price of fuel goes up, the combined demand for fuel/autos would decline. It also allows for specific treatment of the demand for fuel and intra-fuel substitutability. A remaining part of the fuel demand

(heating, cooking) could then be combined with (shares of) other goods and services into a housing service bundle.

Table 2.1: Core sets

Set	Description
<i>is</i>	Full set of SAM Accounts
<i>aa(is)</i>	Armington agents—includes all production activities and final demand
<i>a(aa)</i>	Production activities
<i>oa(aa)</i>	Other Armington agents—mostly final demand accounts
<i>fd(oa)</i>	Final demand accounts (excludes the trade and transport margin accounts)
<i>h(oa)</i>	Household accounts
<i>f(oa)</i>	Other final demand accounts
<i>i(is)</i>	Commodities
<i>e(i)</i>	Energy commodities
<i>kNest</i>	Consumed commodity bundles
<i>inst(is)</i>	Institutions (for transfers)
<i>fp(is)</i>	Factors of production
<i>l(fp)</i>	Labor types
<i>lnd(fp)</i>	Land types
<i>cap(fp)</i>	Capital types
<i>natres(fp)</i>	Natural resources
<i>v</i>	Vintages (Old and New)

The next sections describe the different blocks or modules of the model using the rather traditional circular flow scheme of economics, i.e. starting with production and factor incomes, income distribution, demand, trade and macro closures. It is followed by a discussion on the model's dynamics.

Production block

Production functions

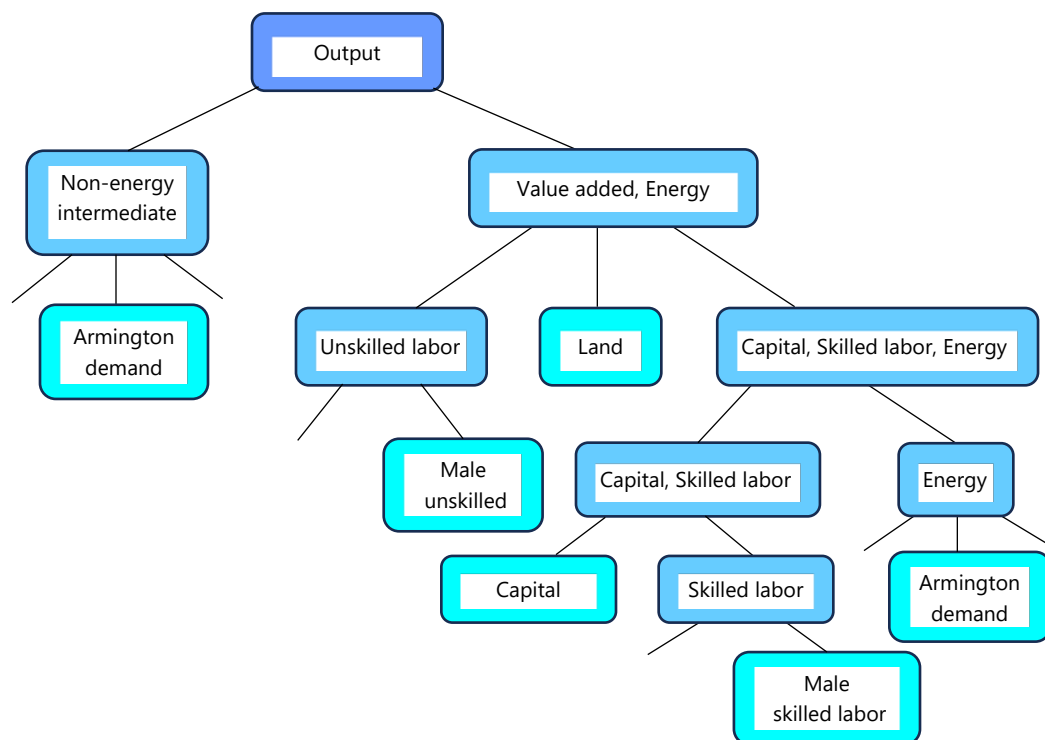
All equations related to production are indexed by a for production activities and v for capital vintages (*old* and *new* where *new* uses capital equipment newly installed at the beginning of the period while *old* uses capital greater than a year old)². Note that the list of commodities i is not identical to the list of activities. The MANAGE-WB production structure is a set of nested, potentially activity specific, constant-elasticity-of-substitution (CES) functions, an example is depicted in Figure 2.1. The layered CES structures are depicted in the model code by three core equations, which stay untouched regardless the nesting structure or its parameterization is changed; this approach is called “flexible nesting”.

The purpose of the nested CES structure is to replicate the substitution and complementary relations among various production inputs (intermediate and factors). A substitution elasticity of 1 implies a Cobb-Douglas technology where cost shares are fixed, while a zero elasticity implies a Leontief relation with fixed physical factor shares³. The model differentiates between the pre-tax returns to factors and producers’ factor costs which comprise activity and factor specific ad-valorem factor taxes and subsidies. Similarly, for intermediates, producer cost can comprise activity and commodity specific ad-valorem taxes and subsidies.

² Capital vintages can be understood as *old* and *new* technologies of production. This is an “on-demand” feature of the model so that an analyst can choose to run the model with or without the vintage structure. The vintage structure allows for two impact channels. First, it is typically assumed that the *old* vintage (technology) has lower substitution elasticities for certain nests than the *new* vintage. This reflects that firms can choose for new investments between the vintages (technologies) which, for instance, differ in their capital and energy intensity. Once the decision is made and the capital matches the chosen technology installed, the flexibility in these intensities is smaller. Higher saving rates will lead to a higher share of the *new* vintage and thus greater overall flexibility in adjusting production to the changes in input prices and productivity. The second channel is through the allocation of capital across sectors. *New* capital is assumed to be perfectly mobile across sectors whereas *old* capital is linked to the *old* vintage. In declining sectors, the return to capital will be less than the economy-wide average and some old capital will be released according to an upward-sloping supply curve, such that *old* capital in expanding sectors is mobile and sluggish in declining ones. This is explained in greater detail in the market equilibrium section.

³ See, for instance, van der Mensbrugghe (2020) for the key analytical properties of the CES and constant-elasticity-of-transformation (CET) functions and the derivations of optimality problems using these functions.

Figure 2.1: Example of a typical flexible production nesting structure



In Figure 2.1, the boxes in green depict inputs bought from factor or commodity markets – i.e., intermediate inputs (Armington demand for different goods and/or services) or factors of production (here capital, different types of labor and land). These bought inputs are combined, here over multiple layers of CES aggregates, to produce total output depicted in dark blue at the top. The light blue boxes depict these CES aggregates, composed of these inputs and/or of other CES aggregates.

In the example above, a top nest combines the value-added-energy bundle and non-energy intermediate inputs into output via a CES function. The non-energy bundle is a CES aggregate of all non-energy intermediate commodities. The value-added-energy bundle is a CES aggregate of demand for an unskilled labor bundle, demand for land (where appropriate for a sector) and the capital-skilled labor-energy bundle. Unskilled labor is a CES aggregate of male and female unskilled labor, where such differentiation is available in the data. Having a composite bundle comprising capital and energy carriers is crucial in climate change abatement, to allow firms to reduce the energy intensity of production by investments. Finally, the energy bundle depicts the substitution possibilities between different energy carriers, such as coal, oil, gas and petroleum products.

Without the flexible nesting, extending or modifying a nesting structure such as shown above requires (re)defining several (new) variables (quantities and prices), equations and parameters (shares, shifters, and elasticities of substitution) for each node, and modifying or adding GAMS code for the benchmarking. With the flexible nesting, an appropriate production structure for a specific country or project can now be easily defined, including exemptions for specific activities. Instead of being fixed in the GAMS code, the names of the nests, their structure and substitution elasticities are defined in an EXCEL workbook, called the bridge-file. This file also comprises all other project specific sets, data and parameters, such as the SAM for the country.

Figure 2.2 shows how a flexible production nesting is specified in the bridge-file, using a quite evolved example. The top row comprises the parent nests. For example, *f-top*, *va* and *nd* denote total production, value-added and total intermediate, respectively. The second row defines the default elasticity of substitutions in the parent nests; vintage specific exemptions can be inputted in another sheet in the bridge-file. The Leontief case with zero-elasticity must be inputted by *eps*. A flag of 1 in the third row activates the additive variant of the CES function, if the cell is left empty, the default form of the CES function is used. The first column in the table comprises either the label "all" to indicate the default nesting structure for all activities or a label of an activity such as "*a-ctl*" for the cattle activity to define activity-specific nesting structures replacing the defaults. The second column gives the name of the child nest. The numbers in the child rows indicate the distribution of this child's cost at the benchmark to one or several parent nests. These shares must add up to unity. In the example below, only 1:1 relation between children and parent nests are shown.

With the exemption of the top-level nest *f-top*, all other parent nests shown in the first row must also occur as a child in the table. Equally, all intermediate inputs and all factors of production must be found in a row in the table to allocate them in shares to one or multiple parent nests.

Figure 2.2: Example of flexible nesting information for the production function in the bridge-file

		labSk	labus	cap_labSk	va	cap_ener	ener	ely_tnd	gas_gdt	ener_gasoil	ener_fossil	wtr_land	aez_land	crops	anim_meat	f-top	nd
	subsElas	0.5	0.5	0.2	0.7	0.25	0.5	eps	eps	0.7	0.5	0.5	7.5	2	2	eps	eps
	additive						1			1	1						
		labSk	labus	cap_labSk	va	cap_ener	ener	ely_tnd	gas_gdt	ener_gasoil	ener_fossil	wtr_land	aez_land	crops	anim_meat	f-top	nd
all	f-off_mgr_pros	1															
all	f-tech_aspros	1															
all	f-m_skl	1															
all	f-f_skl	1															
all	f-clerks		1														
all	f-service_shop		1														
all	f-ag_othlowsk		1														
all	f-m_nsk		1														
all	f-f_nsk		1														
all	labSk			1													
all	f-Capital			1													
all	f-NatlRes				1												
all	labus				1												
all	cap_ener				1												
all	wtr_land				1												
all	cap_labSk					1											
all	ener					1											
all	ener_fossil						1										
all	ely_tnd						1										
all	c-p_c						1										
all	c-TnD							1									
all	c-ely							1									
all	c-gdt								1								
all	c-gas								1								
all	c-oil									1							
all	gas_gdt									1							
all	ener_gasoil										1						
all	c-coa										1						
all	f-Land											1					
all	aez_land											1					
all	f-water											1					
all	f-AEZ1												1				
all	f-AEZ2												1				
all	c-pdr													1			
all	c-wht													1			
all	c-gro													1			
all	c-v_f													1			
all	c-osd													1			
all	c-c_b													1			
all	c-pfb													1			
all	c-ocr													1			
all	c-ocr													1			
all	c-ctl														1		
all	c-ctlRais														1		
all	c-oap														1		
all	c-rmk														1		
all	c-wol														1		
all	c-cmt														1		
all	c-omt														1		
all	va															1	
all	nd																1
all	crops																1
all	anim_meat																1
all	c-frs																1

Note: The colors are used to better visualize the structure but have no impact on the nesting itself.

The production structure under the flexible nesting is represented by the following five equations which encompass the differentiation between the standard CES and its additive variant. Equation (1) determines the cost-minimal level of demand for child in sector a and vintage v at time t , $xFac_{Chl,a,v,t}$, in case of the normal CES function. The term Chl states that a factor or commodity is a *child* of a parent factor or commodity nest denoted by Par .⁴ Parents combine factors and/or commodities into

⁴ The terms Chl and Par are used here for convenience only. The GAMS codes have $fpChl$ and $fpPar$ for Chl and Par respectively to differentiate them from the other flexible nests in the model.

production bundles as well as bundles into bundles. For instance, a composite commodity of skilled labor *labsk* could represent the parent factor of male and female skilled labor (*f-m_skl* and *f-f_skl*), while itself being a child of a parent *cap_labSk* combining it with capital *f-capital* as in Figure 2.2.

$xFac_{Chl,a,v,t} = \sum_{Par} \left(\alpha_{Chl,Par,a,v}^{xFac} (1 + \chi_{Chl,Par,a,v,t}^{\alpha_{xFac}}) (1 + twist_{Chl,Par,a,v,t}) \lambda_{Chl,a,v,t}^{xFac} \sigma_{Par,a,v}^{xFac-1} \left(\frac{pxFac_{Par,a,v,t}}{pxFac_{Chl,a,v,t}} \right)^{\sigma_{Par,a,v}^{xFac}} xFac_{Par,a,v,t} \right) + markUpNonComp_{Chl,a,t} xFac_{fptop,a,v,t}$	(1)
---	-----

The above optimality condition states that the quantity demanded of a child (a single factor, a commodity or a CES composite) depends on the factor augmenting (Hicks neutral) productivity change $\lambda_{Chl,a,v,t}^{xFac}$, the prices including taxes $pxFac_{Par,a,v,t}$ and $pxFac_{Chl,a,v,t}$, the share parameter of the child in the parent $\alpha_{Chl,Par,a,v,t}^{xFac}$, the quantity of parent $xFac_{Par,a,v,t}$, and the elasticity of substitution between children within a parent $\sigma_{Par,a,v}^{xFac}$. $\chi_{Chl,Par,a,v,t}^{\alpha_{xFac}}$ is non-zero only under the forward-looking extension. Similarly, $twist_{Chl,Par,a,v,t}$ remains zero if the twist option is not active. Finally, the demand for a child in a parent nest is summed over the parent nests to find the total demand for the child, to accommodate the case where a child is comprised in shares in multiple parents. The code allows to add in a Leontief-fashion child demands driven by the total output and a factor $markUpNonComp_{Chl,a,t}$ which could be interpreted as non-competitive cost terms.

Equation (2) determines the price index of each parent nest. It is a non-linear aggregator of the prices (and/or price indexes) of the children comprised in this parent, taking the share parameters and productivity changes into account.

$pxFac_{Par,a,v,t} = \left(\sum_{Chl} \alpha_{Chl,Par,a,v}^{xFac} (1 + \chi_{Chl,Par,a,v,t}^{\alpha_{xFac}}) (1 + twist_{Chl,Par,a,v,t}) \left(\frac{pxFac_{Chl,a,v,t}}{\lambda_{Chl,a,v,t}^{xFac}} \right)^{1-\sigma_{Par,a,v}^{xFac}} \right)^{\frac{1}{1-\sigma_{Par,a,v}^{xFac}}}$	(2)
--	-----

For additive CES functions, the expression in (1) is written as follows:

$xFac_{Chl,a,v,t} = \sum_{Par} \left(\alpha_{Chl,Par,a,v}^{xFac} (1 + \chi_{Chl,Par,a,v,t}^{\alpha_{xFac}}) (1 + twist_{Chl,Par,a,v,t}) \lambda_{Chl,a,v,t}^{xFac} \sigma_{Par,a,v}^{-\sigma_{Par,a,v}^{xFac}} \left(\frac{pxFacInd_{Par,a,v,t}}{pxFac_{Chl,a,v,t}} \right)^{\sigma_{Par,a,v}^{xFac}} xFac_{Par,a,v,t} \right) + markUpNonComp_{Chl,a,t} xFac_{fptop,a,v,t}$	(3)
--	-----

The dual price index $pxFacInd_{Par,a,v,t}$ in equation (2) is determined in equation (4).

$pxFacInd_{Par,a,v,t} = \left(\sum_{Chl} \alpha_{Chl,Par,a,v}^{xFac} (1 + \chi_{Chl,Par,a,v,t}^{\alpha_{xFac}}) (1 + twist_{Chl,Par,a,v,t}) \lambda_{Chl,a,v,t}^{xFac} e^{-\sigma_{Par,a,v}^{xFac} pxFac_{Chl,a,v,t}} e^{-\sigma_{Par,a,v}^{xFac}} \right)^{\frac{1}{-\sigma_{Par,a,v}^{xFac}}} \quad (4)$	
---	--

In the additivity case, one needs to define the revenue exhaustion price $pxFac_{Par,a,v,t}$ as a function of $pxFacInd_{Par,a,v,t}$ as follows:

$pxFac_{Par,a,v,t} = \left(\sum_{Chl} \alpha_{Chl,Par,a,v}^{xFac} (1 + \chi_{Chl,Par,a,v,t}^{\alpha_{xFac}}) (1 + twist_{Chl,Par,a,v,t}) \lambda_{Chl,a,v,t}^{xFac} e^{-\sigma_{Par,a,v}^{xFac} pxFac_{Chl,a,v,t}} e^{-\sigma_{Par,a,v}^{xFac}} \right)^{\frac{1}{-\sigma_{Par,a,v}^{xFac}}} pxFacInd_{Par,a,v,t}^{\sigma_{Par,a,v}^{xFac}} \quad (5)$	
---	--

Equations (6) and (7) link $xFac_{Chl,a,v,t}$ and $pxFac_{Chl,a,v,t}$ with the intermediate Armington demand $xa_{i,a,t}$ and the user-specific price $paf_{i,a,t}$ for commodity i . The intermediate demand for commodity i from activity a is $xa_{i,a,t}$ is the sum over the vintages of the factor demands for this commodity, while the prices faced by the vintages are identical and equal to the tax inclusive user price.

$xa_{i,a,t} = \sum_v xFac_{i,a,v,t} \quad (6)$	
$pxFac_{i,a,v,t} = \chi_i^{pa} paf_{i,a,t} \quad (7)$	

A so-called twist is a cost-neutral shifter in two share parameters in a CES nest. In MANAGE-WB, it is typically used to target a cost share or total cost of a specific production factor. One might be tempted to simply increase respectively decrease the two share parameters by the same amount, but doing so neglects the impact of an updated price relation compared to the benchmark. The update-formula is therefore somewhat more evolved and requires the updated cost share of one of the children, called here $shrxFac$ and defined in equation (9) below. For a derivation, see van der Mensbrugghe (2020). If the twist option is active for a combination of two children in a parent nest, the following equation defines $twist_{Chl,Par,a,v,t}$ where $\chi_{Par,a,t}^{twFa}$ and $\chi_{Par,t}^{twF}$ are endogenous or exogenous productivity shifters affecting the demand of the parent nest.

$twist_{Chl,Par,a,v,t} = \left[\frac{1 + \sum_{Chl} (\chi_{Par,a,t}^{twFa} + \chi_{Par,t}^{twF})}{1 + ((\chi_{Par,a,t}^{twFa} + \chi_{Par,t}^{twF}) \sum_{Chl} shrxFac_{Chl,Par,a,old,t-1})} (1 + twist_{Chl,Par,a,v,t-1}) - 1 \right] tw_{Par,a,v,t} \quad (8)$	
$shrxFac_{Chl,Par,a,old,t} = \frac{\sum_v xFac_{Chl,a,v,t} pxFac_{Chl,a,v,t}}{\sum_v xFac_{Par,a,v,t} pxFac_{Par,a,v,t}} \quad (9)$	

Equation (10) determines the level of input augmenting productivity change:

$\lambda_{fpNest,a,v,t}^{xFac} = \chi_{fpNest,t}^{xFacf} \chi_{a,t}^{xFacA} \chi_{v,t}^{xFacV} \chi_{fpNest,a,t}^{xFacFA} (1 + \chi_t^{xFacL}) (1 + \chi_t^{xFacK}) (1 + \chi_t^{xFacKL}) (1 + \chi_t^{xFacFP}) \chi_{aagr,t}^{xFacAgr} (1 + \chi_t^{xFacNrg})$	(10)
---	------

where the set $fpNest$ contains all Chl and Par sets, $\chi_{fpNest,t}^{xFacf}$ is the general productivity shifter for nest $fpNest$ for all activities and vintages, $\chi_{a,t}^{xFacA}$ is the sector-specific productivity shifter for all vintages and nests, $\chi_{v,t}^{xFacV}$ is the vintage-specific productivity shifter for activities and nests, $\chi_{fpNest,a,t}^{xFacFA}$ is the activity-and-nest-specific productivity shifter for all vintages, χ_t^{xFacL} is the productivity shifter for labor in all sectors, χ_t^{xFacK} is the productivity shifter for capital in all sectors, χ_t^{xFacKL} is the productivity shifter capital and labor, χ_t^{xFacFP} is the productivity shifter for natural factors (land, labor, capital etc.), $\chi_{aagr,t}^{xFacAgr}$ is the productivity shifter for a group of activities and $\chi_t^{xFacNrg}$ is the productivity shifter for energy carriers.

Equation (11) determines the aggregate unit cost for each sector $px_{a,t}$ as the weighted average of the vintage-specific unit costs, using vintage-specific output levels as weights. In this equation, the subscript $fpTop$ stands for the top node of the nested structure – i.e., output. Equation (12) determines the final market price for output $pp_{a,t}$ that is equal to the aggregate unit cost augmented by the ad-valorem output tax ($\tau_{a,t}^{Prd}$ and $\tau_{a,t}^{Phn}$), the ad-valorem subsidy $\tau_{a,t}^{Psb}$ and the emission tax $\tau_{em,a,t}^{Emi}$. The emission tax is a function of the emissions per unit of activity $\rho_{em,a}^{EmiXP}$, a parameter determining whether the sector's emissions are taxed $\phi_{em,a,t}^{EmiXP}$, and an emission- and sector-specific tax value shifter $\chi_{em,a,t}^{EmiXP}$. The latter allows us to differentiate the emission tax rates by the type of emission and sector.

The equivalence of the tax-adjusted unit cost to the output price is an implication of two key assumptions: constant-returns-to-scale technologies where average costs are equal to marginal costs, and perfect competition such that no mark-ups over marginal after-tax costs occur. These assumptions also exclude fixed production costs. There are two exemptions from these: (1) non-Leontief markups (on demand) which drive a share of factor demand as a function of total output, they are not subject to factor specific productivity gains, (2) and markups which finance R&D if the endogenous technical progress module is active.

$px_{a,t} xp_{a,t} = \sum_{fpTop,v} xFac_{fpTop,a,v,t} (1 + markup_{a,t}) pxFac_{fpTop,a,v,t} + \sum_{fp,v} markupNonComp_{fp,a,t} xFac_{fpTop,a,v,t} pxFac_{fp,a,v,t}$	(11)
$pp_{a,t} = (1 + \tau_{a,t}^{Prd} + \tau_{a,t}^{Psb} + \tau_{a,t}^{Phn}) px_{a,t} + \sum_{em} \tau_{em,a,t}^{Emi} \phi_{em,a,t}^{EmiXP} \chi_{em,a,t}^{EmiXP} \rho_{em,a}^{EmiXP}$	(12)

Equation (13) allows to target the $yield_{fp,a,t}$ for factors (output per unit of factor use) of production in each sector. $m_{dmgAgg,dmg,fp,a}$ is a macro representing climate change damages on factor fp in sector a . It is active if the yield is fixed. A variant of this equation in the model aggregates over all types of land.

$xp_{a,t} = m_dmgAgg_{dmg,"fp,a"} yield_{fp,a,t} \sum_v xFac_{fp,a,v,t}$	(13)
---	------

MANAGE-WB allows for differentiated labor productivity for primary, secondary and tertiary sectors of production in the baseline scenario. The variable $labProd_t$ refers to the average labor productivity in service sectors and is implicitly defined by the targeted real GDP level. Labor productivity differences for non-service sectors are a quadratic function of real GDP growth as expressed in equation (14). For details, see Roson (2019) and Roson and Britz (2021).

$\chi_{fp,a,t}^{xFacFA} = -\min \left(-labProd_t, -labProd_t \left(prodShift_{a',a} + prodShift_{b',a} \left(\frac{rGDPmp_t}{rGDPmp_{t-1}} - 1 \right) + prodShift_{c',a} \left(\frac{rGDPmp_t}{rGDPmp_{t-1}} - 1 \right)^2 \right) \right) + \chi_{fp,a,t-1}^{xFacFA}$	(14)
---	------

As in other CGE models, the supply of land and natural resources is not directly linked to population growth unlike labor and or capital. It reacts instead with a typical low factor supply elasticity to changes in the respective rental rates. In a growing economy, this implies that these rental rates may grow at an unreasonable rate. This can be addressed by assuming productivity improvement in the use of land and natural resources. Equations (15) to (17) provide an option to link productivity shifts for land, natural resources and capital with a multiplicative factor to changes in labor productivity – either with $labProd_t$ if differentiated labor productivity is selected or χ_t^{xFacL} if not.

$\chi_{lnd,t}^{xFacF} = 1 - \min(0, -((labProd_t + labProd_{t-1}) \text{ or } \chi_t^{xFacL}))$	(15)
$\chi_{nates,t}^{xFacF} = 1 - \min(0, -((labProd_t + labProd_{t-1}) \text{ or } \chi_t^{xFacL}))$	(16)
$\chi_{cap,t}^{xFacF} = 1 - \min(0, -((labProd_t + labProd_{t-1}) \text{ or } \chi_t^{xFacL}))$	(17)

Multi-input and multi-output specifications

This section of the supply block describes the multi-input and multi-output specifications of the model. Multi-input allows for multiple activities producing the same commodity. Such a commodity can be assumed quality differentiated across these activities based on a CES function or to be homogenous implying the law-of-one-price such that the unit cost of production is identical across activities. A multi-input specification is useful, for instance, to depict electricity production differentiated by hydro, solar, oil or coal powered such that power mix adjust under carbon taxation. The multi-output part of the model allows for joint production of multiple outputs by the same activity, for example producing both ethanol and DDGS in an ethanol production activity.

To address the multi-output case, activity a produces quantities $x_{a,i,t}$ of a suite of commodities indexed by i . In the case of a diagonal make matrix, each activity will produce exactly one commodity. Distributing the total production capacity of activity a to its potentially multiple outputs is implemented based on a Constant-Elasticity-of-Transformation (CET) structure, where ω_a^p denotes the

elasticity of transformation. Zero transformation elasticity implies fixed physical output shares. Equation (18) defines the supply of $x_{a,i,t}$, emanating from production output $xp_{a,t}$ where the law-of-one-price holds in the case of infinite transformation. Equation (19) represents the revenue balance for the multi-output production function. Again, $\chi_{a,i,t}^p$ are parameters used as normalization factors so that the price variables are initialized at 1. Wherever the normalized prices are used in equations, they need to be scaled by the relevant $\chi_{a,i,t}^p$ to preserve the correct accounting.

$\begin{cases} x_{a,i,t} = ap_{a,i,t} \left(\frac{\chi_{a,i,t}^p p_{a,i,t}}{pp_{a,t}} \right)^{\omega_a^p} xp_{a,t} & \text{if } \omega_a^p \neq \infty \\ \chi_{a,i,t}^p p_{a,i,t} = pp_{a,t} & \text{if } \omega_a^p = \infty \end{cases}$	(18)
$\begin{cases} pp_{a,t} = \left(\sum_i ap_{a,i,t} (\chi_{a,i,t}^p p_{a,i,t})^{1+\omega_a^p} \right)^{\frac{1}{1+\omega_a^p}} & \text{if } \omega_a^p \neq \infty \\ xp_{a,t} = \sum_i \chi_{a,i,t}^p x_{i,a,t} & \text{if } \omega_a^p = \infty \end{cases}$	(19)

In the next step, multiple streams of output, such as electricity produced from oil, gas, coal etc., can be combined into a single supplied commodity $xs_{i,t}$ when entering the market, based either on a CES aggregator or assuming homogeneity. In the latter case, the cost of each stream must be equal, considering the price normalization factors. Equation (20) determines the demand for produced commodity $x_{a,i,t}$. In the case of a finite elasticity of substitution σ_i^s , it employs a CES formulation. With an infinite substitution elasticity, the law-of-one price must hold – i.e., the producer price of each component must be equal in efficiency units. Equation (21) determines the equilibrium condition in the form of the cost function or (primal) volume equality if the law-of-one price holds. The formulas allow for shifts in preferences via the $\gamma_{a,i,t}^p$ parameter.

$\begin{cases} x_{a,i,t} = as_{a,i,t} (\gamma_{a,i,t}^p)^{\sigma_i^s - 1} \left(\frac{\chi_{a,i,t}^{ps} p_{s,i,t}}{\chi_{a,i,t}^p p_{a,i,t}} \right)^{\sigma_i^s} xs_{i,t} & \text{if } \sigma_i^s \neq \infty \\ p_{a,i,t} \chi_{a,i,t}^p = \gamma_{a,i,t}^p \chi_{a,i,t}^{ps} p_{s,i,t} & \text{if } \sigma_i^s = \infty \end{cases}$	(20)
$\begin{cases} p_{s,i,t} \chi_{a,i,t}^{ps} = \sum_a \left(as_{a,i,t} \left(\frac{\chi_{a,i,t}^p p_{a,i,t}}{\gamma_{a,i,t}^p} \right)^{1-\sigma_i^s} \right)^{\frac{1}{1-\sigma_i^s}} & \text{if } \sigma_i^s \neq \infty \\ xs_{i,t} = \sum_a x_{a,i,t} & \text{if } \sigma_i^s = \infty \end{cases}$	(21)

The optimality conditions and the dual price indexes in equations (22) and (23) are derived in the case of a normal CES structure. In the case of an additive (or volume-preserving CES structure) with $\sigma_i^s \neq \infty$, they can be written as follows:

$x_{a,i,t} = a s_{a,i,t} (\gamma_{a,i,t}^p)^{-\sigma_i^s} \left(\frac{\chi_i^{ps} pInd_{i,t}}{\chi_{a,i,t}^p p_{a,i,t}} \right)^{\sigma_i^s} x s_{i,t}$	(22)
$pInd_{i,t} \chi_i^{ps} = \sum_a \left(a s_{a,i,t} (\gamma_{a,i,t}^p \chi_{a,i,t}^p p_{a,i,t})^{-\sigma_i^s} \right)^{\frac{1}{1-\sigma_i^s}}$	(23)

Again, the revenue exhaustion price, $ps_{i,t}$, in the case of an additive CES function needs to be defined as follows:

$ps_{i,t} = \left(\sum_a a s_{a,i,t} (\gamma_{a,i,t}^p)^{-\sigma_i^s} p_{a,i,t}^{1-\sigma_i^s} \right) pInd_{i,t}^{\sigma_i^s}$	(24)
--	------

Equation (25) computes the share of electricity generated with each electricity source $elecMix_{aElec,iElec,t}$ where $aElec$ is a set of activities producing a set of electricity commodities $iElec$.

$elecMix_{aElec,iElec,t} = \frac{\chi_{aElec,iElec,t}}{\sum_{aElec} \chi_{aElec,iElec,t}}$	(25)
--	------

Income block

Factor income

Equation (26) defines gross factor income $factY_{fp,t}$ – i.e., the pre-tax level of total factor remuneration from each factor of production defined in the fp set in each activity a and vintage v covering wages, capital, natural resource and land rents. Pre-tax refers here to a case where factor taxes paid by the activities. Equation (27) defines gross profits $kapY_t$ as the sum of all capital earnings.

$factY_{fp,t} = \sum_{a,v} pxFac_preTax_{fp,a,v,t} xFac_{fp,a,v,t}$	(26)
$kapY_t = \sum_{cap} factY_{cap,t}$	(27)

Households

Equation (28) defines gross income $yH_{h,t}$ for household h based on four terms. The first one distributes the economy-wide payments of each factor of production across households based on their shares in the respective factor income governed by the matrix of coefficients $\chi_{h,fp,t}^{fac}$. It is used for the factors of production where one aggregate agent manages the economy-wide factor stocks. The second term considers the case in which factor stocks are allocated and managed by individual institutions (for more, see the section on factor supplies). This allows for a more refined description of income allocation, for instance, to link households' wage incomes to specific production activities.

The third term adds up $transfers_{h,inst,t}$ from other institutions to household h and the fourth term is the debt payment (principal and interest) received from the government, $debtPay_{h,t}$. Households'

disposable income $yHDsp_{h,t}$ is defined in equation (29) as the gross income, net of personal income taxes and deducting transfers $totTr_{h,t}$ from household h to the other institutions.

$yH_{h,t} = \sum_{fp} \chi_{h,fp,t}^{fac} factY_{fp,t}$ $+ \sum_{fmp,vf} xfmSup_{fmp,fmp,h,vf,t} pfmSup_{fmp,fmp,h,vf,t} corrFac_{fp,t}$ $+ \sum_{inst} transfers_{h,inst,t} + debtPay_{h,t}$	(28)
$yHDsp_{h,t} = yH_{h,t}(1 - t_{h,t}^{pit}) - totTr_{h,t}$	(29)

The next block of equations refers to the allocation of households' income. Equation (30) determines household savings $savH_{h,t}$ which is a share $asav_{h,t}$ of the disposable income and evolves with a shifter χ_t^{ms} , updated by the relations of returns to savings $fundPrice_t$ relative to a composite price $pindH_{h,t}$, exponent elasticity ω_h^{sav} . The GDP deflator $pGDPmp_t$ enters in these equations to maintain the stationarity of the ratio in the event of positive inflation. The price dependence of the marginal propensity to save vanishes if ω_h^{sav} is set to zero. According to equation (31), $pindH_{h,t}$ is a composite price index of the returns to savings (a weighted average of expected returns to new investments and the interest rate on new government debt, see section on domestic fund markets below) and the aggregate expenditure price for household consumption $pf_{h,t}$. Equation (32) defines the marginal propensity to save. If not fixed, it serves for information only. Equation (33) specifies the level of transfers from households to other institutions, $totTr_{h,t}$ as a share of total income. Equation (34) determines available household income for purchases of goods and services $yf_{h,t}$ as the difference between disposable income and savings.

$savH_{h,t} = \chi_t^{ms} asav_{h,t} yHDsp_{h,t} \left(\frac{fundPrice_t pGDPmp_t}{pindH_{h,t}} \right)^{\omega_h^{sav}}$	(30)
$pindH_{h,t} = \left(asav_{h,t} (pGDPmp_t fundPrice_t)^{\omega_h^{sav}} + (1 - asav_{h,t}) pf_{h,t}^{\omega_h^{sav}} \right)^{\frac{1}{\omega_h^{sav}}}$	(31)
$aps_{h,t} = \frac{savH_{h,t}}{yHDsp_{h,t}}$	(32)
$totTr_{h,t} = \chi_{h,t}^{totTr} yH_{h,t}$	(33)
$yf_{h,t} = yHDsp_{h,t} - savH_{h,t}$	(34)

Equation (35) approximates the Gini coefficient using income of households in different income groups $yH_{h,t}$, based on the relative position of each element in the set of households (usually this will be the position of a decile in income distribution) ord_h and the total number of household groups $card_h$. Equation (36) computes the so-called dependency ratio, a demographic metric defined as the ratio of the population of a given cohort to the working-age population.

$gini_t = 2 \frac{\sum_h ord_h yH_{h,t}}{\sum_h card_h yH_{h,t}} - \frac{card_h + 1}{card_h}$	(35)
$depRate_{cohort,t} = \frac{pop_{cohort,t}}{pop_{15-64,t}}$	(36)

Government

Government tax revenues $yG_{g,t}$ stem from multiple tax streams g where the set g comprises the following tax elements: factor taxes in equation (37), subsidies⁵ on production in equation (38), production taxes in equation (39), phantom taxes in equation (40), domestic and import value-added taxes in equations (41) and (42), export and import taxes in equations (43) and (44), total agent-specific sales taxes and subsidies in equation (45), taxes on emissions (from both the use of energy commodities and process emissions in equations (46) and (47), respectively) and direct taxes in equation (48). Note that there is no set called dtx in the model code. It is introduced here for convenience only. Rather, the model code calculates household-specific direct tax revenue under $yG_{pit,h,t}$ and enterprise-specific direct tax revenue under $yG_{cit,entr,t}$.

Whereas production taxes and subsidies are reported in the database, phantom taxes are not observable but can be used to recover exogenous sector shares, for instance, for different activities producing electricity. The sum of the phantom tax revenue is then usually fixed to zero to render them revenue neutral. For instance, fossil-fuel based power sectors can be taxed and renewables subsidized. Keeping the sum of these taxes and subsidies at zero will imply that the phantom taxation does not change (directly) the average electricity price.

⁵ Subsidies are considered as negative taxes in the model. In general, however, they are viewed as public expenditure in government statistics.

$yG_{fac,t} = \sum_a \sum_v \tau_{a,fp,t}^{fac} px_{fac_preTax_{fp,a,v,t}} x_{Fac_{fp,a,v,t}}$	(37)
$yG_{psb,t} = \sum_a \tau_{a,t}^{psb} px_{a,t} xp_{a,t}$	(38)
$yG_{prd,t} = \sum_a \tau_{a,t}^{prd} px_{a,t} xp_{a,t}$	(39)
$yG_{phn,t} = \sum_a \tau_{a,t}^{phn} px_{a,t} xp_{a,t} = 0$	(40)
$yG_{vatDom,t} = \sum_i (\chi_{i,t}^{pd} pd_{i,t} + pMarg_{i,t} tm g_{i,t}^d) \tau_{i,t}^{vat,d} xd_{i,t}$	(41)
$yG_{vatImp,t} = \sum_i (eR_t(1 + \tau_{i,t}^{imp}) pwm_{i,t} + pMarg_{i,t} tm g_{i,t}^m) \tau_{i,t}^{vat,m} xm_{i,t}$	(42)
$yG_{exp,t} = \sum_i \tau_{i,t}^{exp} \chi_{i,t}^{pe} pe_{i,t} xe_{i,t}$	(43)
$yG_{imp,t} = \sum_i \tau_{i,t}^{imp} eR_t pwm_{i,t} (xm_{i,t} + mdelst_{i,t})$	(44)
$yG_{paf,t} = \sum_i \sum_{aa} \tau_{i,aa,t}^{paf} \chi_i^{pa} pa_{i,t} xa_{i,aa,t}$	(45)
$yG_{emi,t} = \sum_{aa} \sum_{emSrc} \sum_{em} \tau_{em,t}^{emi} emi_{em,t}^{cal} \chi_{em,i,aa,t}^{emi} \rho_{em,emSrc,aa}^{emi} \phi_{em,emSrc,aa}^{emi} XA_{emSrc,aa,t}$	(46)
$yG_{emiXp,t} = \sum_a \sum_{em} \tau_{em,a,t}^{emi} \rho_{em,a}^{emiXp} \phi_{em,a,t}^{emiXp} \chi_{em,a,t}^{emiXp} xp_{a,t} \\ + \sum_{em,fp} \tau_{em,a,t}^{emi} \rho_{em,fp,a}^{emiFac} \phi_{em,fp,a,t}^{emiFac} \chi_{em,fp,a,t}^{emiFac} \sum_{a,v} x_{Fac_{fp,a,v,t}}$	(47)
$yG_{dtx,t} = \sum_h \tau_{h,t}^{pit} yH_{h,t} + \sum_{entr} \tau_{h,t}^{cit} entrY_{entr,t} + gTax_t$	(48)

For information, the model code calculates factor tax revenue by activities for each production factor, agent-specific tax revenue for each commodity under $yG_{paf,i,t}$, and emissions tax revenue from energy consumption for Armington agents under $yG_{emi,aa,t}$ and from process emissions for activities under $yG_{emiXp,a,t}$. These equations are not shown here.

The next block of equations determines closures for the government accounts.⁶ Equation (49) defines the total revenues for the government $yGov_t$, comprised of three components: aggregate tax revenues net of subsidies $yGToT_t$ as defined in equation (50), transfers from the other institutions and shares on factors income. Equation (51) calculates government direct taxes on its revenue. Equation (52) shows that total public transfers are a share, $\chi_{gov,t}^{tottr}$, of gross domestic product (GDP at market prices), and are affected by $fiscalRespDrv_t$ generated by the difference between actual and target debt-to-GDP ratios, when the fiscal responsibility option is activated. The term $relExpInc$ captures the strength of the debt reducing policy via expenditure cuts.

$yGov_t = yGToT_t + \sum_{inst} transfers_{gov,inst,t} + \sum_{fp} \chi_{gov,fp,t}^{Fac} factY_{fp,t} + \sum_{fmp,vf} xfmSup_{fmp,fmp,gov,vf,t} pfmSup_{fmp,fmp,gov,vf,t} corrFac_{fp,t}$	(49)
$yGToT_t = \sum_g yG_{g,t}$	(50)
$gTax_t = dgovTax_t yGov_t$	(51)
$totTr_{gov,t} = \chi_{gov,t}^{tottr} (1 + (relExpInc/100) fiscalRespDrv_t) GDPmp_t$	(52)

The so-called primary balance of the government $primaryBalance_t$ as defined in equation (53) is the difference between total government revenue $yGov_t$ minus taxes paid by government, public transfers including redistributed tax revenues, public consumption and government (productive and adaptive) investments. It does not consider any financial flows related to government debt. The fiscal balance $fiscalBalance_t$, as defined in equation (54), considers in addition interest payments, $DdbtInterest_{instfd,t}$, to foreign and domestic debt holders $instfd$. The government balance $govBalance_t$ in equation (55) takes additionally the principal payments $DebtPrincipal_{inst,t}$ into account. Equation (56) defines the government balance in real terms.

In case of a positive $govBalance_t$, the government will provide additional savings to finance private investments, otherwise, this balance will be equal to new government debt $debtNew_t$ (see section on fund markets).

⁶ In many recursive dynamic CGE models, a common closure rule is to fix the ratios of government spending to GDP and public investment to GDP. When government debt is incorporated into such models, it is essential to ensure that the debt-to-GDP ratio remains below a specified upper threshold. This typically requires adjusting either the government spending and investment ratios or tax rates. In MANAGE-WB, this functionality is supported. Additionally, for greater flexibility, the model's GUI offers an optional automatic adjustment mechanism to help maintain the debt-to-GDP ratio within acceptable limits.

$primaryBalance_t = yGov_t - gTax_t - totTr_{gov,t} - yTaxRedist_t - yf_{gov,t} - \sum_{ginv} yf_{ginv,t} - \sum_{adpinv \in GOV} yf_{adpinv,t}$	(53)
$fiscalBalance_t = primaryBalance_t - \sum_{instfd} debtInterest_{instfd,t}$	(54)
$govBalance_t = fiscalBalance_t - \sum_{instfd} debtPrincipal_{instfd,t}$	(55)
$rGovBalance_t = govBalance_t / pgdpmp_t$	(56)

Enterprises

The next set of equations relates to the enterprise accounts as pass-through accounts with minimal behavior⁷. Equation (57) defines enterprise incomes from three sources: from factors of production, income from transfers, and from debt payments received. Equation (58) determines enterprise savings that add to the funds available for savings/investment and domestic new government debt, depending on the share parameter $asavEntr_t$ and a shifter variable χ_t^{ms} . The share reacts to changes in the returns to the average return to savings, $fundPrice$ if $\omega_{savEntr,t}$ is non-zero. Equation (59) determines aggregate enterprise transfers as a residual, after direct taxes paid by the enterprise and enterprise savings are subtracted from total enterprise income.

$entrY_{entr,t} = \sum_{fp} \chi_{entr,fp,t}^{fac} factY_{fp,t} + \sum_{fmp,vf} xfmSup_{fmp,fmp,entr,vf,t} pfmSup_{fmp,fmp,entr,vf,t} corrFac_{fp,t} + \sum_{inst} transfers_{entr,inst,t} + debtPay_{entr,t} + \sum_{fptop,v,a} markup_{a,t} xFac_{fptop,a,v,t} pxfac_{fptop,a,v,t} - \sum_f xfdRD_{f,t} pf_{f,t}$	(57)
$savEntr_{entr,t} = \chi_t^{ms} asavEntr_t entrY_{entr,t} (1 - \tau_{entr,t}^{cit}) fundPrice_t^{\omega_{savEntr}}$	(58)
$totTr_{entr,t} = entrY_{entr,t} (1 - \tau_{entr,t}^{cit}) - savEntr_{entr,t}$	(59)

Transfers

The next set of equations in the income block allocates transfers across institutions, handling the domestic and foreign transfers separately. Equation (60) allocates the total transfers of domestic agents $totTr_{dinst,t}$, indexed by $dinst$, using a matrix of distribution coefficients $\chi_{dinst,dinst,t}^{tr}$ to the different domestic and foreign institutions $inst$. In the case of the government account, transfers can be complemented by a share of redistributed tax revenues, an option used in the context of recycling

⁷ Enterprises represent financial and nonfinancial corporations in the System of National Accounts.

of carbon taxes. The transfers from the rest of world $transfers_{inst,t}^{ROW}$ are either linked with population or GDP and in foreign currency terms and are converted to domestic currency units by the exchange rate eR_t in equation (61). They are typically either driven by changes in GDP or in population.

$transfers_{inst,dinst,t} = \chi_{inst,dinst,t}^{tr} totTr_{dinst,t} + \chi_{taxRedist,inst,t}^{tr} yTaxRedist_t$	(60)
$transfers_{inst,row,t} = eR_t transfers_{inst,t}^{ROW}$	(61)

Domestic fund markets

MANAGE-WB defines a virtual domestic investor who collects households' saving $savH_{h,t}$ and enterprise savings $savEntr_t$ into a variable called $fundT_t$ as defined in equation (62):

$fundT_t = \sum_h savH_{h,t} + \sum_{entr} savEntr_{entr,t}$	(62)
--	------

The investor distributes $fundT_t$ to finance private investments $fundP_t$ and the government's new domestic debt $fundG_t$ in equation (63).⁸

$fundT_t = fundP_t + fundG_t$	(63)
-------------------------------	------

The distribution is based on maximizing revenues under an additive CET function. Equation (64) shows the related composite price $fundPriceD_t$ as an average of relative changes in the interest rate $interest_t$ and in the expected returns to capital $rore_t$, considering the transformation elasticity ω_{fund} .

$fundPriceD_t = \left(aFundG_t \chi_t^{fundG} \left(\frac{interest_t pGDPmp_t}{interest_{t0}} \right)^{\omega_{fund}} + (1 - aFundG_t \chi_t^{fundG}) \left(\frac{rore_t pGDPmp_t}{rore_{t0}} \right)^{\omega_{fund}} \right)^{\frac{1}{\omega_{fund}}}$	(64)
---	------

The normalization with the benchmark variables $interest_{t0}$ and $rore_{t0}$ implies that $fundPriceD_t$ is equal to unity at the benchmark. $aFundG_t$ is a share parameter calibrated from the benchmark portfolio composition. If ω_{fund} is zero, these distribution shares are fixed.

The government's new domestic debt $fundG_t$ is defined in equation (65):

$fundG_t = aFundG_t \chi_t^{fundG} \chi_t^{fundG} fundT_t \left(\frac{interest_t pGDPmp_t}{interest_{t0} fundPriceD_t} \right)^{\omega_{fund}}$	(65)
--	------

The additive CET form implies that the dual price aggregator $fundPriceD_t$ is not equal to the exhaustion price $fundPrice_t$ which is therefore defined in equation (66):

⁸This is a key feature of the model. For this mechanism to function, the SAM includes an additional "debt" account, which tracks how government borrowing (i.e., the fiscal deficit) is financed—whether through domestic, foreign, or a combination of sources—and shows how government debt payments are distributed among economic agents, including households, enterprises, and the rest of the world.

$fundPrice_t fundT_t = \frac{interest_t}{interest_{t0}} fundG_t + \frac{rore_t}{rore_{t0}} fundP_t$	(66)
---	------

The expected rate of return to capital $rore_t$ is defined as:

$rore_t = \left(\frac{kstocke_t}{kstock_t} \right)^{\varepsilon_t^{ror}} rorc_t$	(67)
---	------

If the forward-looking option is used, $rore_t$ is defined over a set of the current and future years $tNext$ and related weights $exptWgts_{tNext}$:

$rore_t = \frac{\sum_{tNext \geq t} exptWgts_{tNext} \left(\frac{kstockE_t}{kstock_{tNext}} \right)^{\varepsilon_t^{ror}} rorc_{tNext}}{\sum_{tNext \geq t} exptWgts_{tNext}}$	(68)
---	------

The variable $rore_t$ corrects the cost adjusted rate of return to capital $rorc_t$ with a factor depending on the expected capital stock $kStocke_t$ over the current capital stock $kstock_t$ exponent an elasticity. This correction reflects that at unchanged other endowments, an increase in the capital stock should imply a drop in the returns the capital stock. The expected capital stock $kStockE_t$ is current year stock minus depreciation plus productive investments $prodInv_t$:

$kStockE_t = (1 - depr_t)kstock_t + prodInv_t$	(69)
--	------

The productive investments $prodInv_t$ defined in equation (70) are the sum of private investments $xfid_{inv,f}$, a share $phiPcInv_t$ of government $xfid_{ginv,t}$, minus investments into R&D services under the endogenous technology module and exogenous non-productive investments $nonProdInv_{inv,t}$:

$prodInv_t = \sum_{inv} xfid_{inv,f} + phiPcInv_{inv,t} \sum_{ginv} xfid_{ginv,t} - xfRD_{inv,t} - nonProdInv_{inv,t}$	(70)
--	------

The cost adjusted rate of return to capital $rorc_t$, often called Tobin's Q, is determined by the economy-wide return to capital $trent_t$, the price index of investments $pf_{inv,t}$ and the depreciation rate $depr_t$, see equation (71).

The economy-wide depreciation rate $depr_t$ is defined in equation (72) as a weighted average of the activity specific depreciation rates $depr_{a,t}$, which can be subject to climate change damages to capital stocks captured by the macro $m_dmgAgg_{dmgDepr,"a"}$. Equally, in short run shocks, changes in the capital utilization rate driven by the short run elasticity for intensive margin changes in factor use $\varepsilon_{cap,a}^{intens}$ applied to changes in capital rents $pxFac_preTax_{cap,a,old,t}$ compared to the previous year update the depreciation rate. The capital supplies $xfmSupc_{a,cap,"sumInst",v,t}$ act as weights in defining the economy wide average depreciation rate.

$rorc_t = \frac{trent_t}{pf_{inv,t}} - depr_t$	(71)
$depr_t = \frac{\sum_a \sum_{cap} \sum_v xfmSup_{a, cap, "sumInst", v, t} deprA_{a, t} m_dmgAgg_{dmgDepr, "a"} \left(1 + \varepsilon_{cap, a}^{intens} \left(\frac{pxFac_preTax_{cap, a, old, t}}{pxFac_preTax_{cap, a, old, t}} - 1 \right) \right)}{\sum_a \sum_{cap} \sum_v xfmSup_{a, cap, "sumInst", v, t}}$	(72)

Another relevant variable is the capital utilization rate ror_t which is determined in the following equation (73).

$\frac{ror_t}{ror_0} = \left(\frac{trent_t}{trent_{t0} pGDPmp_t} \right)^{\varepsilon_t^{ror}}$	(73)
--	------

The capital utilization rate ror_t is a ratio between capital stock $kStock_t$ and capital use $kapS_t$ in equation (74). $kapS_t$ is equal to the economy-wide supply of capital and thus exhausts the sum of capital demands of the production activities. If ε_t^{ror} is zero, the capital utilization rate is constant. A non-zero ε_t^{ror} implies that an increase in the returns to capital leads to a higher utilization rate of capital stock and vice versa. This, in turn, should have consequences on the capital depreciation rate, but this feedback is not yet captured in the model. Accordingly, it is recommended to keep ε_t^{ror} at zero.

$kapS_t = m_dmgAgg_{dmgSup, "cap"} ror_t kStock_t$	(74)
---	------

New foreign government debt

The contribution of the ROW to new government debt $fundF_t$ is captured by equation (75). This foreign new debt $fundF_t$ is defined as a share of domestic GDP at current market prices $GDPmp_t$, modified by changes in the interest rate on foreign debt of government $debtIntF_t$ according to elasticity σ^{fint} and a sovereign risk correction term driven by a sigmoid function depending on $debtGDP_t$, the debt to GDP ratio:

$fundF_t = \gamma_t^{fundF} \left(interest_t \sum_{instf} riskPrem_{instf, t} \right)^{\sigma^{fint}} \frac{GDPmp_t}{GDPmp_{t0}} pGDPmp_t [1 - \min(0, (sigmoid([debtGDP_t - 0.5] * 4) - 0.5) * 2)]$	(75)
---	------

The risk correction term $-\min(0, (sigmoid([debtGDP_t - 0.5] * 4) - 0.5) * 2)$ is equal to zero at a debt to GDP ratio below 50%, ~0.2 at 60%, ~0.38 at 70%, ~0.54 at 80%, ~0.66 at 90%, 0.76 at 100%, showing the typical saturation towards unity for a sigmoid function. The correction term implies that foreign investors will require higher interest rates for the same amount of new debt at given GDP if the debt-to-GDP ratio increases.

Government borrowing

The amount of government borrowing $debtNewT_t$ in equation (76) is the new gross debt and equal to the difference between government income $yGov_t$ and government spending (consumption, investments, transfers, debt principal payment and interest on debt) – i.e., negative $govBalance_t$.⁹

$debtNewT_t = \min(0, govBalance_t)$	(76)
--------------------------------------	------

The total amount of new debt, as defined in equation (77), is equal to the sum of foreign and private funds allocated to government debt. This clearance condition implicitly defines the interest rate on government debt:

$debtNewT_t = fundG_t + fundF_t$	(77)
----------------------------------	------

Equations (78) and (79) correspond to the demand and supply of government borrowing and define the interest rate $interest_t$ paid on new government debt. In addition, the following conditions hold where the sets $instd$ and $instf$ have domestic and foreign institutions, respectively.

$debtNew_{instd,t} = \chi_{instd,t}^{debtNew} fundG_t$	(78)
$debtNew_{instf,t} = fundF_t$	(79)

Debt stocks and savings

The next set of equations governs the dynamics of government debt. Equation (80) determines the stock of government debt held by institutions $debtStk_{instfd,t}$, indexed with $instfd$ which contains both domestic and foreign institutions. Current year stock is equal to the stock in the previous period plus the new government debt $debtNew_{instfd,t}$ minus debt principal payments $debtPrincipal_{instfd,t}$. Equation (81) calculates the total debt stock $debtStkT_t$ by summing over the debt stocks of the different institutions. Equation (82) defines the debt-to-GDP ratio $debtGDP_t$. Equation (83) determines the total debt payments by government to different institutions $instfd$. It considers the interest payments $debtInterest_{instfd,t}$ as defined in equation (84) and principal payments $debtPrincipal_{instfd,t}$ as defined in equation (85). $rDebt_t^{stkT}$ and $rDebt_{instfd,t}^{stk}$ are common and institute specific repayment rates respectively, while $avgDebtInt_{instfd,t}$ is the average interest rate, calculated in equation (86) as the weighted average of past interest rates. Equation (87) computes the total new debt.

$debtStk_{instfd,t} = debtStk_{instfd,t-1} + debtNew_{instfd,t} - debtPrincipal_{instfd,t}$	(80)
$debtStkT_{instfd,t} = \sum_{instfd} debtStk_{instfd,t}$	(81)
$debtGDP_t = \frac{debtStkT_t}{GDPmp_t}$	(82)

⁹ The minimum operator is not differentiable. To avoid related numerical problems, it is technically implemented in the code based on a differentiable smoothing function.

$debtPay_{instfd,t} = debtInterest_{instfd,t} + debtPrincipal_{instfd,t}$	(83)
$debtInterest_{instfd,t} = avgDebtInt_{instfd,t}(debtStk_{instfd,t} - debtNew_{instfd,t})$	(84)
$debtPrincipal_{instfd,t} = rDebt_t^{StkT} rDebt_{instfd,t}^{Stk} debtStk_{instfd,t-1}$	(85)
$avgDebtInt_{instfd,t} = \frac{\sum^{t-1} \max(0, debtNew_{instfd,t}(1 - rDebt_{instfd,t}^{Stk})) interest_t riskPrem_{instfd,t}}{\sum^{t-1} \max(0, debtNew_{instfd,t}(1 - rDebt_{instfd,t}^{Stk}))}$	(86)
$debtNewT_t = \sum_{instfd} debtNew_{instfd,t}$	(87)

Equation (88) is the ubiquitous investment equals savings (I=S) equation. Nominal productive investment $yf_{inv,t}$ is equal to $fundP_t$, the amount of households and enterprise savings not used for new domestic government debt, government savings (in case of a positive government balance) and FDI (the difference between total foreign savings $savF_t$ multiplied by the exchange eR_t and new foreign government debt $fundF_t$). In order to define the value of productive private investments, the value of stock building¹⁰ and non-productive private investment dedicated to climate adaptation and to finance R&D efforts under the endogenous technology extension are subtracted.

$yf_{inv,t} = fundP_t - \max(0, -govBalance_t) + eR_t savF_t - fundF_t - tdelst_t - \sum_{adpinv \in inv} yf_{adpinv,t} + xfdRD_{inv,t} pf_{inv,t}$	(88)
---	------

Demographic equations

The following block of equations defines a set of demographic variables. Equations (89) and (90) define population by age $popAge_{age,t}$. $popAge_{age=1,t}$ depicts the birth-rate $brate_t$ times the last period working age population between 15 and 64 years old $pop_{15-64,t-1}$. $popAge_{age>1,t}$ is determined by the deathrate by age $drate_{age,t}$ and the population in previous age and climate damage $m_dmgAgg_{dmg,"age"}$. Equation (91) aggregates population by age into cohorts. Equation (92) computes the population of each household group, h where $t0$ implies the initial period and assumes that all households grow with the same factor.

$popAge_{1,t} = brate_t pop_{15-64,t-1}$	(89)
$popAge_{age,t} = popAge_{age-1,t-1} drate_{age,t} + m_dmgAgg_{dmg,"age"}$	(90)
$pop_{cohorts,t} = \sum_{age \in cohorts} popAge_{age,t}$	(91)

¹⁰The value of imports of stocks is the tariff-inclusive domestic price-excluding margin and other indirect tax wedges.

$pop_{h,t}^h = \frac{pop_{h,t0}^h}{pop_{total,t0}} pop_{total,t}$	(92)
---	------

Demand block

Household demand

The top-level nest of the households' demand system is depicted by a constant-difference-in-elasticity (CDE) utility function, the demand system found in the GTAP standard model. It defines the demand for single products or product bundles which are CES composite of commodities or other CES composites. This bundling of commodities reflects a flexible nesting structure available to final demanders. A commodity can be allocated to multiple nests so that the total demand for it needs to be summed over the multiple nests. For instance, part of energy demand might be allocated to a transport service and the remainder to a household (heating, cooking, lighting etc.,) bundle. The CDE is based on an implicit additive utility function and can collapse to a CES utility function under certain conditions. As it comprises three vectors of product (nest) specific parameters, it provides a fair amount of flexibility and is typically calibrated to external estimates of both income and own-price elasticities.

To ease the implementation of the CDE demand system, equation (93) defines an auxiliary variable $cdeInt_{h,k,t}$ for each household h and a product bundle k . Adding up $cdeInt_{h,k,t}$ over k as in equation (94) defines a scaling factor for each household h which is then used in the household's Marshallian demand $xak_{h,k,t}$ for each bundle in equation (95). Equation (96) determines household utility (on per capita basis) $uh_{h,t}$ and is an implicit function of the terms $cdeInt_{h,k,t}$ and the price related parameters $bh_{h,k,t}$. Equation (97) is an implicit function of money metric utility $mmuh_{h,t}$ which can be used to quantify the equivalent variation to estimate welfare effects for each household in monetary terms. Equation (98) defines the budget share $xakShr_{oa,k,t}$ for all final demand agents (oa), including households h , and for demand bundle k . Equations (99)-(101) describe alternative ways to define the household specific consumer price index (CPI) $pf_{h,t}$. In the default settings, CPI is computed using a Paasche index as in equation (99). Alternatively, the user could choose to use a Laspeyres price index in equation (100), or to compute CPI using a weighted sum of consumer goods prices in equation (101).

$cdeInt_{h,k,t} = akf_{h,k,t} bh_{h,k,t} uh_{h,t}^{eh_{h,k,t} bh_{h,k,t}} \left(\frac{pxak_{h,k,t}}{yf_{h,t}/pop_{h,t}^h} \right)^{bh_{h,k,t}}$	(93)
$consScale_{h,t} = \sum_k cdeInt_{h,k,t}$	(94)
$xak_{h,k,t} = \frac{cdeInt_{h,k,t}}{consScale_{h,t}} \frac{yf_{h,t}}{pxak_{h,k,t}}$	(95)
$\sum_k \frac{cdeInt_{h,k,t}}{bh_{h,k,t}} = 1$	(96)
$\sum_k akf_{h,k,t} uh_{h,t}^{eh_{h,k,t} bh_{h,k,t}} \left(\frac{1}{mmuh_{h,t}/pop_{h,t}^h} \right)^{bh_{h,k,t}} = 1$	(97)

$xakShr_{oa,k,t} = \frac{xak_{oa,k,t}pxak_{oa,k,t}}{yf_{oa,t}}$	(98)
$pf_{h,t} = \left(\sum_i xa_{i,h,t}paf_{i,h,t} \right) / \left(\sum_i xa_{i,h,t}paf_{i,h,t0} \right)$	(99)
$pf_{h,t} = \left(\sum_i xa_{i,h,t0}paf_{i,h,t} \right) / \left(\sum_i xa_{i,h,t0}paf_{i,t0} \right)$	(100)
$pf_{h,t} = \sum_k xakShr_{h,k,t}pkf_{h,k,t}$	(101)

Other final demand for bundles: government and investment

The demand decomposition of the other final demanders (i.e., government and investment) uses CES functions. Equation (102) determines the demands for a top-level bundle k for each other final demand agent f where $xdf_{f,t}$ is the aggregate quantity index for the agent, determined in specific closure rules typically as a share of real GDP. Equation (103) reflects the standard CES dual price aggregator function determining $pf_{f,t}$.

$xak_{f,k,t} = akf_{f,k,t}xdf_{f,t} \left(\frac{pf_{f,t}}{pxak_{f,k,t}} \right)^{\sigma_f^f}$	(102)
$pf_{f,t} = \left(\sum_k afk_{f,k,t}pxak_{f,k,t}^{1-\sigma_f^f} \right)^{\frac{1}{1-\sigma_f^f}}$	(103)

Final demand for sub-demand bundles and single products

The following set of equations refers to the nested CES structure for all final demand accounts oa (households, investment, governments). Demand decomposition of oa uses a flexible CES nesting structure in which $knest$ is a child and a parent and $knest1$ is its alias. Equation (104) determines the demand $xak_{oa,knest,t}$ for a child bundle $knest$ for each oa where $knest1$ are potentially multiple parent nests to which the child bundle is allocated in shares. Equation (105) shows the composite price $pxaf_{oa,knest,t}$ of a bundle which could include commodities captured by $paf_{i,oa,t}$ and bundles of commodities captured by $pxak_{oa,knest1,t}$ where $knest1$ depicts a child nest. $\lambda_{oa,i,knest}^{oa}$ and λ_t^{nrg} (which is zero for non-energy commodities) are efficiency variables capturing technological progress. The macro $m_dmgAgg_{dmgDem,"oa,i,knest"}$ captures Climate Change Damages to final demand efficiency.

$xak_{oa,knest,t} = \sum_{knest1} afk_{oa,knest,knest1,t} xak_{oa,knest1,t} \left(\frac{pxak_{oa,knest1,t}}{pxak_{oa,knest,t}} \right)^{\sigma_{oa,knest1}^c} \quad (104)$	
---	--

$pxaf_{oa,knest,t} = \left(\sum_i af_{oa,i,knest,t} \left(\frac{\chi_i^{pa} paf_{i,oa,t}}{m_dmgAgg_{dmgDem,"oa,i,knest"} \lambda_{oa,i,knest,t}^{oa} (1 + \lambda_t^{nrg})} \right)^{1-\sigma_{oa,knest}^c} \right. \\ \left. + \sum_{knest1 \in knest} afk_{oa,knest1,knest,t} pxak_{oa,knest1,t}^{1-\sigma_{oa,knest}^c} \right)^{\frac{1}{1-\sigma_{oa,knest}^c}} \quad (105)$	
--	--

Equation (106) defines the Armington demands $xa_{i,oa,t}$. The summation over $knest$ and indexing the share parameter $af_{oa,i,knest,t}$ both with i and $knest$ indicates that one product can be demanded through multiple bundles, for instance, fuel demand both by a transport bundle and a heating service bundle. The equation also comprises the option to introduce product and nest specific preference shifters $\lambda_{oa,i,knest}^{oa}$, and for all type of energy λ_t^{nrg} .

$xa_{i,oa,t} = \sum_{i \in knest} af_{oa,i,knest,t} xak_{h,knest,t} \left(m_dmgAgg_{dmgDem,"oa,i,knest"} \lambda_{oa,i,knest}^{oa} (1 + \lambda_t^{nrg}) \right)^{\sigma_{oa,knest}^c - 1} \left(\frac{pxak_{oa,knest,t}}{\chi_i^{pa} paf_{i,oa,t}} \right)^{\sigma_{oa,knest}^c} \quad (106)$	
---	--

The share parameters in investment demand can be dynamically updated on-demand depending on sector-specific investment vectors. In this case, the nesting for investment demand comprises solely one level called "top" with a zero-substitution elasticity in equation (104). The shifters for the investment demand $\lambda_{inv,i,top}^{oa}$ become endogenous and update the investment composition. The update first defines a volume index for total capital demand $newInv_t$ in equation (107). From the capital demands of the new vintages and a share of each old vintage $oldCapInvShare$ which by default is 2.5%, about half of a typical economy wide depreciation rate. Considering small shares of the *old* vintage in the volume index stabilizes the behavior in case where sectors shrink.

$newInv_t = \sum_a xFac_{a,cap,new,t} + oldCapInvShare_a \cdot xFac_{a,cap,old,t} \quad (107)$	
--	--

The update of investment shares is based on activity specific share parameters $\gamma_{i,a,t}^{inv}$, weighted according to their contribution to $newInv_t$. The normalization factor $\lambda_{inv,i,t0}^{inv}$ considers the differences between the volume index and total investment demand at the benchmark.

$\lambda_{inv,i,top}^{oa} \lambda_{inv,i,t0}^{oa} = \frac{1}{af_{inv,i,"top",t0}} \sum_a \gamma_{i,a,t}^{inv} (xFac_{a,cap,new,t} + oldCapInvShare_a xFac_{a,cap,old,t}) / newInv_t \quad (108)$	
--	--

The activity specific share parameters are driven by a CES demand function. The related price index $invPrice_{a,t}$ is defined in equation (109) and used to update the benchmark investment shares of the individual activities in equation (110).

$invPrice_{a,t} = \left[\sum_t \gamma_{i,a,t0}^{inv} \sum_{inv} (\chi_i^{pa} paf_{i,inv,t})^{1-\sigma_a^{inv}} \right]^{1/1-\sigma_a^{inv}}$	(109)
$\gamma_{i,a,t}^{inv} = \gamma_{i,a,t0}^{inv} \left(\frac{invPrice_{a,t}}{\sum_{inv} (\chi_i^{pa} paf_{i,inv,t})} \right)^{\sigma_a^{inv}}$	(110)

This mechanism is especially relevant for stock changes as part of investments which ensure that stock levels present a certain share of domestic output.

Demand for trade and transport margins

The final set of equations in the demand block refers to the optional demand for goods and services for the trade and transport margins. The model has three different nodes for margins: 1) from farm or factory gate to domestic markets ('d'); 2) from farm or factor gate to the domestic border for exports ('x'); and 3) from the border to domestic markets for imports ('m'). Though the wedges can differ across nodes, the unit cost of the margin for each commodity i , is the same across all nodes. This implies that the 'production' function of the trade and transport margins is identical across all nodes. Equation (111) expresses the demand for margin services for good j across all three nodes $xmargin_{j,t}$. This demand generates demand for 'Armington' goods, where an additive CES function is assumed for the volumes as in equation (112).¹¹ The price index for the additivity constraint $pmargin_{j,t}$ is in (113) where it is corrected to be unity in the initial period while the average cost of the trade margins is reflected in the exhaustion price $pmarg_{j,t}$ which is determined in equation (114). The differential cost structure for margins across goods will reflect, for instance, the different energy intensities as reflected in the cost shares, and the indirect demand for energy through the input output table.

$xmargin_{j,t} = tmg_{j,"d",t}xd_{j,t} + tmg_{j,"x",t}xe_{j,t} + tmg_{j,"m",t}xm_{j,t}$	(111)
$xa_{i,j,t} = amarg_{i,j}xmargin_{j,t} \left(\frac{pmargin_{j,t}}{\chi_i^{pa} pa_{i,t}} \right)^{\sigma_j^{mg}}$	(112)
$pmargin_{j,t} = \frac{\left(\sum_i amarg_{i,j} (\chi_i^{pa} pa_{i,t})^{-\sigma_j^{mg}} \right)^{\frac{1}{-\sigma_j^{mg}}}}{\left(\sum_i amarg_{i,j} (\chi_i^{pa} pa_{i,t0})^{-\sigma_j^{mg}} \right)^{\frac{1}{-\sigma_j^{mg}}}}$	(113)
$pmarg_{j,t} = \sum_i amarg_{i,j} (\chi_i^{pa} pa_{i,t})^{1-\sigma_j^{mg}} pmargin_{j,t}^{\sigma_j^{mg}}$	(114)

Equation (115) shows the price and volume split of total expenditure $yf_{oa,t}$ for the final demand agents. What variables these equations determine depends on specific closure rules to be discussed below.

¹¹ Margin services are not associated with specific indirect taxes – though this could be changed in future revisions.

$yf_{oa,t} = pf_{oa,t} xfd_{oa,t}$	(115)
------------------------------------	-------

Stock change behavior

Endogenous stock changes are introduced in the current version of the model only if the forward-looking option is selected where the model is solved simultaneously for the current and a number of future years. The first equation related to stock change is the usual stock-flow relation for each product i :

$st_{i,t} = st_{i,t-1} + delst_{i,t}$	(116)
---------------------------------------	-------

where $st_{i,t}$ is domestic stock level for i and $delst_{i,t}$ is the change in its stock. The second equation (117) in this blocks depicts the share of last year's stock added or released in the current year based on (a) the expected price change $psExpRel_{i,t}$ as in equation (118) and (b) a desired average stock level $\alpha_{i,t}^{st}$ relative to expected output $xsExp_{i,t}$:

$delst_{i,t} = st_{i,t-1} \left[\left\{ \max(0, -psExpRel_{i,t}) \right\}^{\rho_i^{pdelst}} - \left\{ \min(0, psExpRel_{i,t}) \right\}^{\rho_i^{pdelst}} \right] + xsExp_{i,t} \left\{ \frac{2\alpha_{i,t}^{st}}{(st_{i,t} + st_{i,t-1})/xsExp_{i,t}} \right\}^{\rho_i^{xdelst}}$	(117)
--	-------

Using both current and last year stock levels in the expression in the second line dampens the impact of current year prices induced changes on targeted stock levels. The *min* and *max* expressions in equation (117) imply a symmetric behavior of stock changes for positive and negative expected relative price changes. As *min* and *max* are not differentiable, they are implemented instead based on an NCP Veelken-Ulbrich smooth minimum, sinus-based function `ncpVU` in GAMS. $\alpha_{i,t}^{st}$ defines the desired stock level relative to domestic output, considering the elasticity ρ_i^{xdelst} , whereas ρ_i^{pdelst} defines how sensitive stock changes to react to the variable $psExpRel$ ¹², defined as

$psExpRel_{i,t} = \frac{psExp_{i,t} - ps_{i,t}}{psExp_{i,t}}$	(118)
---	-------

This variable thus defines differences between expected average and current year realized prices, relative to expected average ones. Price expectations are formed over a number of future years $tNext$ which are simultaneously solved for in each model solve:

$psExp_{i,t} = \frac{\sum_{tNext < t+n} ps_{i,tNext} expWgts_{tNext}}{\sum_{tNext < t+n} expWgts_{tNext}}$	(119)
--	-------

¹² Tests have shown that 0.75 is a reasonable level for ρ_i^{pdelst} . As $psExpRel$ measures relative deviations in prices, the related absolute changes are usually closer to zero than to unity. Using small value of ρ_i^{pdelst} , such as 0.1, implies that a difference between expected and realized prices of 1% will release 63% of previous stocks.

where n is the maximal foresight period and $expWgts$ are the weights for the future time points. Note here that for the final years of the simulation horizon, not all n future points can be considered due to the finite simulation horizon. The code therefore extends automatically the simulation period by n years beyond the last reported year. The same approach as for prices is used to define expected output levels:

$xsExp_{i,t} = \frac{\sum_{tNext < t+n} xs_{i,tNext} expWgts_{tNext}}{\sum_{tNext < t+n} expWgts_{tNext}}$	(120)
--	-------

An alternative to define desired stock levels relative to domestic output would relate them to total market appearances, i.e., production plus imports minus exports. This could be an appropriate option if, for instance, stochastic world market price shocks for fossil fuel importers are analyzed and it is assumed that the country analyzed maintains stocks of these fuels.

Armington decomposition: Import and domestic demand

The demand equations as discussed above determine the so-called Armington demand for goods $xa_{i,t}$ as the composite of import and domestic demand for different agents: activities (a), households (h), other final demand (f) and for margin delivery (j). The union of these sets defines all Armington agents aa which are assumed to share the same preferences for domestic and imported goods¹³. Each Armington good i is thus assumed homogenous across agents and aggregated linearly across them in volume terms, see equation (123).

Equations (121) and (122) define the import and domestic prices $pma_{i,t}$ and $pmd_{i,t}$, respectively. In equation (121), defining $pma_{i,t}$, the CIF price is reflected in the variable $pWm_{i,t}$, multiplied by the exchange rate eR_t and adjusted by the import tariff $\tau_{i,t}^{Imp}$ and trade and transport margins from the border to the point of sale $pMarg_{i,t}tmg_{i',m',t}$. Finally, an import specific value added tax, $\tau_{i,t}^{vat,m}$, is added to the import price after tariffs to become the agent's price of imported goods $pma_{i,t}$. Similarly, according to equation (122), the agents' price of domestically produced goods $pda_{i,t}$ is equal to the domestic producer price $pd_{i,t}$ augmented by domestic trade and transport margins $pMarg_{i,t}tmg_{i',d',t}$ and the domestic value added tax $\tau_{i,t}^{vat,d}$. The terms χ_i^{pma} , χ_i^{pda} and χ_i^{pd} are the price normalization factors and $tmg_{i',m',t}$ and $tmg_{i',d',t}$ are the margin rates. Notice that these agents' prices are uniform across the Armington agents and do not include further agent specific sales and emission taxes.

$\chi_i^{pma} pma_{i,t} = (eR_t(1 + \tau_{i,t}^{Imp})pWm_{i,t} + pMarg_{i,t}tmg_{i',m',t})(1 + \tau_{i,t}^{vat,m})$	(121)
$\chi_i^{pda} pda_{i,t} = (\chi_i^{pd} pd_{i,t} + pMarg_{i,t}tmg_{i',d',t})(1 + \tau_{i,t}^{vat,d})$	(122)

Equation (123) aggregates Armington demand at the national level $xaT_{i,t}$ as the sum across all agents. Aggregate national demand for domestic goods $xd_{i,t}$ is a fraction of $xaT_{i,t}$ which is sensitive to the

¹³ Some national SAMs and the GTAP database allow for the decomposition of Armington demand into its domestic and import components by agent. Even more detailed databases, such as a multi-regional input-output table (MRIO), allow for agent-specific preferences for imports by region of origin.

price of the domestically produced good $pda_{i,t}$ relative to the Armington composite good price $pa_{i,t}$ as shown in equation (124). The key parameter, known as the Armington substitution elasticity, is σ_i^m . Similarly, equation (125) determines the demand for imports $xm_{i,t}$. Note that equations (124) and (125) are derived from a normal CES function while equations (126) and (127) are from an additive volume-preserving CES form.

$xaT_{i,t} = \sum_{aa} xa_{i,aa,t}$	(123)
$xd_{i,t} = ad_{i,t} \gamma_{i,t}^{md(\sigma_i^m-1)} \left(\frac{pa_{i,t}}{pda_{i,t}} \right)^{\sigma_i^m} xaT_{i,t}$	(124)
$xm_{i,t} = am_{i,t} \gamma_{i,t}^{mm(\sigma_i^m-1)} \left(\frac{pa_{i,t}}{pma_{i,t}} \right)^{\sigma_i^m} xaT_{i,t}$	(125)
$xd_{i,t} = ad_{i,t} \gamma_{i,t}^{md(-\sigma_i^m)} \left(\frac{paInd_{i,t}}{pda_{i,t}} \right)^{\sigma_i^m} xaT_{i,t}$	(126)
$xm_{i,t} = am_{i,t} \gamma_{i,t}^{mm(-\sigma_i^m)} \left(\frac{palm_{i,t}}{pma_{i,t}} \right)^{\sigma_i^m} xaT_{i,t}$	(127)

Equations (128) and (129) define the Armington composite price of commodity i in the cases of the normal and additive CES forms, respectively.

$pa_{i,t} = \left(ad_{i,t} (pda_{i,t} / \gamma_{i,t}^{md})^{1-\sigma_i^m} + am_{i,t} (pma_{i,t} / \gamma_{i,t}^{mm})^{1-\sigma_i^m} \right)^{\frac{1}{1-\sigma_i^m}}$	(128)
$paInd_{i,t} = \left(ad_{i,t} (pda_{i,t} / \gamma_{i,t}^{md})^{-\sigma_i^m} + am_{i,t} (pma_{i,t} / \gamma_{i,t}^{mm})^{-\sigma_i^m} \right)^{\frac{1}{-\sigma_i^m}}$	(129)

If the additive CES form is chosen for some commodities, the exhaustion price $pa_{i,t}$ needs to be defined as follows.

$pa_{i,t} = \left(ad_{i,t} (pda_{i,t} / \gamma_{i,t}^{md})^{1-\sigma_i^m} + am_{i,t} (pma_{i,t} / \gamma_{i,t}^{mm})^{1-\sigma_i^m} \right) paInd_{i,t}^{\sigma_i^m}$	(130)
--	-------

Tax rates and tax-inclusive Armington price

Equations (131)-(141) define various tax rates as a function of a base tax rate (indicated by a superscript 0), and a set of additive (superscript D) and multiplicative (superscript M) shifters. The τ_t^{Mall} is a multiplicate shifter that affects all the tax rates.

Factor specific taxes for each activity $\tau_{a,fp,t}^{Fac}$ are defined in equation (131). Departing from the base rate $\tau_{a,fp,t}^{Fac0}$, it considers general factor tax shifters as well as factor-specific (labor, capital and land) shifters.

$\tau_{a,fp,t}^{Fac} = \left(\tau_{a,fp,t}^{Fac0} + \tau_t^{FacD0} + \tau_{a,t}^{FacD1} + \tau_{fp,t}^{FacD2} \right) \left(\tau_t^{FacMLab} \tau_t^{FacMLabA} + \tau_t^{FacMCap} + \tau_t^{FacMlnd} \right) \tau_t^{FacM0} \tau_{a,t}^{FacM1} \tau_{fp,t}^{FacM2} \left(\text{sign}(\tau_{a,fp,t-1}^{Fac}) (t_t^{Mall} - 1) + 1 \right)$	(131)
--	-------

Equation (132) defines the activity specific production subsidies $\tau_{a,t}^{Psb}$. Shifters can apply to all activities ($\tau_t^{PsbD0}, \tau_t^{PsbM0}$) or can be activity specific ($\tau_{a,t}^{PsbD1}, \tau_{a,t}^{PsbM1}$):

$\tau_{a,t}^{Psb} = (\tau_{a,t}^{Psb0} + \tau_t^{PsbD0} + \tau_{a,t}^{PsbD1}) \tau_t^{PsbM0} \tau_{a,t}^{PsbM1} \tau_t^{Mall}$	(132)
--	-------

Equation (133) shows the activity-specific production tax rates. Phantom tax rates (activity-specific) are given in equation (134) where *aagr* is a set of aggregated activities.

$\tau_{a,t}^{Prd} = (\tau_{a,t}^{Prd0} + \tau_t^{PrdD0} + \tau_{a,t}^{PrdD1}) \tau_t^{PrdM0} \tau_{a,t}^{PrdM1} \tau_t^{Mall}$	(133)
--	-------

$\tau_{a,t}^{Phn} = \tau_{a,t}^{Phn0} + \tau_t^{PhnD0} + \tau_{a,t}^{PhnD1} + \sum_{a \in aagr} \tau_{aagr,t}^{PhnD2}$	(134)
--	-------

Commodity-specific export and import tax rates are specified in equations (135) and (136). Symmetric to the activity tax rates above, shifters can apply to all commodities or can be commodity specific.

$\tau_{i,t}^{Exp} = (\tau_{i,t}^{Exp0} + \tau_t^{ExpD0} + \tau_{i,t}^{ExpD1}) \tau_t^{ExpM0} \tau_{i,t}^{ExpM1} \tau_t^{Mall}$	(135)
--	-------

$\tau_{i,t}^{Imp} = (\tau_{i,t}^{Imp0} + \tau_t^{ImpD0} + \tau_{i,t}^{ImpD1}) \tau_t^{ImpM0} \tau_{i,t}^{ImpM1} \tau_t^{Mall}$	(136)
--	-------

Equations (137) defines the emission tax rates for emission type (*em*) and the Armington agents (*aa*). Here, shifters can apply to all emissions and agents (index 0), to specific emissions for all agents (index 1) or to specific agents for all emissions (index 2).

$\tau_{em,aa,t}^{Emi} = (\tau_{em,aa,t}^{Emi0} + \tau_t^{EmiD0} + \tau_{em,t}^{EmiD1} + \tau_{aa,t}^{EmiD2}) \tau_t^{EmiM0} \tau_{em,t}^{EmiM1} \tau_{aa,t}^{EmiM2}$	(137)
--	-------

Equations (138) and (139) specify the income tax rates for households (*h*) and enterprises (*entr*) respectively. They can relate to all households or enterprises (index 0) or be households or enterprise specific (index 1).

$\tau_{h,t}^{Pit} = (\tau_{h,t}^{Pit0} + \tau_t^{PitD0} + \tau_{h,t}^{PitD1}) \tau_t^{PitM0} \tau_{h,t}^{PitM1} \tau_t^{Mall}$	(138)
--	-------

$\tau_{entr,t}^{Cit} = (\tau_{entr,t}^{Cit0} + \tau_t^{CitD0} + \tau_{entr,t}^{CitD1}) \tau_t^{CitM0} \tau_{entr,t}^{CitM1} \tau_t^{Mall}$	(139)
--	-------

General value-added tax rates are given in equation (140) while the Armington agent-specific sales tax rates are in equation (141). Similar to above, index 0 is a general shifter for all commodities and agents, index 1 refers to commodity specific shifters and 2 to agent specific ones.

$\tau_{i,n,t}^{Vat} = (\tau_{i,n,t}^{Vat0} + \tau_t^{VatD0} + \tau_{i,t}^{VatD1} + \tau_{n,t}^{VatD2}) \tau_t^{VatM0} \tau_{i,t}^{VatM1} \tau_{n,t}^{VatM2} \tau_t^{Mall}$	(140)
--	-------

$\tau_{i,aa,t}^{Paf} = (\tau_{i,aa,t}^{Paf0} + \tau_t^{PafD0} + \tau_{i,t}^{PafD1} + \tau_{aa,t}^{PafD2}) \tau_t^{PafM0} \tau_{i,t}^{PafM1} \tau_{aa,t}^{PafM2} \tau_t^{Mall}$	(141)
--	-------

Equation (142) defines the end-user price for each Armington agent $paf_{i,aa,t}$. In the absence of a carbon tax, the end-user price is equal to the basic Armington price adjusted by a user-specific sales tax, $\tau_{i,aa,t}^{Paf}$, as defined above. The additive carbon tax is composed of several parts. The

parameter $\rho_{em,i,aa}^{emi}$ measures the quantity of emissions per unit of absorption at the benchmark. It can be adjusted over time by the factor $\chi_{em,i,aa,t}^{emi}$. The carbon tax is given by $\tau_{em,t}^{Emi}$ and is in local currency per unit of emission. The parameter $\phi_{em,i,aa}^{emi}$ allows for variable participation rates across end-users to the tax scheme. If it is set to 1, the participant is subject to the full tax, if set to 0, the agent is tax exempt. The carbon tax is scaled by the emi_t^{cal} scale factor for accounting purposes.

$paf_{i,aa,t} = (1 + \tau_{i,aa,t}^{Paf})pa_{i,t} + \frac{\sum_{em} \tau_{em,t}^{Emi} emi_t^{cal} \chi_{em,i,aa,t}^{emi} \rho_{em,i,aa}^{emi} \phi_{em,i,aa}^{emi}}{\chi_i^{pa}}$	(142)
---	-------

Export supply and domestic sales, import and export prices

Analogous to the Armington specification described above, the model allows allocating domestic output to domestic and export markets based on a CET formulation. Alternatively, an infinite elasticity of transformation can be assumed which implies the law-of-one-price and physical aggregation of volumes of domestic sales and exports.

Equations (143) and (144) represent the revenue maximal supply to domestic $xd_{i,t}$ and export $xe_{i,t}$ markets, respectively, under the normal CET function. They drive implicitly the related prices $pd_{i,t}$ and $pe_{i,t}$. With finite transformation, these conditions are the standard CET first order conditions at given supply $xs_{i,t}$ (less stock building/variation $delst_{i,t}$). With perfect transformation, the implicitly driven prices are replaced with the law-of-one-price such that the domestic $pd_{i,t}$ and export $pe_{i,t}$ prices received by producers are set equal to the aggregate supply price $ps_{i,t}$ where γ_i^{xd} and γ_i^{xe} are price shifters.

$\left\{ \begin{array}{ll} \gamma_{i,t}^{xd} pd_{i,t} = ps_{i,t} & \text{if } \omega_i^x = \infty \\ xd_{i,t} = g_{i,t}^d \gamma_{i,t}^{xd(-1-\omega_i^x)} \left(\frac{pd_{i,t}}{ps_{i,t}} \right)^{\omega_i^x} (xs_{i,t} - delst_{i,t}) & \text{if } \omega_i^x \neq \infty \end{array} \right\}$	(143)
$\left\{ \begin{array}{ll} \gamma_{i,t}^{xe} pe_{i,t} = ps_{i,t} & \text{if } \omega_i^x = \infty \\ xe_{i,t} = m_dmgAgg_{dmgTrd,"i"} g_{i,t}^e \gamma_{i,t}^{xe(-1-\omega_i^x)} \left(\frac{pe_{i,t}}{ps_{i,t}} \right)^{\omega_i^x} (xs_{i,t} - delst_{i,t}) & \text{if } \omega_i^x \neq \infty \end{array} \right\}$	(144)

In the case of the additive CET form, the supply functions are written as in (145) and (146). In this case, the functional form guarantees that $xd_{i,t}$ and $xe_{i,t}$ add up to $(xs_{i,t} - delst_{i,t})$. :

$\left\{ \begin{array}{ll} \gamma_{i,t}^{xd} pd_{i,t} = ps_{i,t} & \text{if } \omega_i^x = \infty \\ xd_{i,t} = g_{i,t}^d \gamma_{i,t}^{xd} \left(\frac{pd_{i,t}}{psInd_{i,t}} \right)^{\omega_i^x} (xs_{i,t} - delst_{i,t}) & \text{if } \omega_i^x \neq \infty \end{array} \right\}$	(145)
$\left\{ \begin{array}{ll} \gamma_i^{xe} pe_i = ps_i & \text{if } \omega_i^x = \infty \\ xe_{i,t} = m_dmgAgg_{dmgTrd,"i"} g_{i,t}^e \gamma_{i,t}^{xe} \left(\frac{pe_{i,t}}{psInd_{i,t}} \right)^{\omega_i^x} (xs_{i,t} - delst_{i,t}) & \text{if } \omega_i^x \neq \infty \end{array} \right\}$	(146)

Equation (147) represents the market equilibrium. With perfect transformation (where all prices are uniform) domestic supply is equal to domestic sales and exports. The price aggregators $ps_{i,t}$ and $psInd_{i,t}$ correspond to the normal and additive CET forms.

$\left\{ \begin{array}{ll} xs_{i,t} = xd_{i,t} + xe_{i,t} + delst_{i,t} & \text{if } \omega_i^x = \infty \\ ps_{i,t} = \left(g_{i,t}^d \left(\frac{pd_{i,t}}{\gamma_{i,t}^{xd}} \right)^{1+\omega_i^x} + m_dmgAgg_{dmgTrd,"i"} g_{i,t}^e \left(\frac{pe_{i,t}}{\gamma_{i,t}^{xe}} \right)^{1+\omega_i^x} \right)^{\frac{1}{1+\omega_i^x}} & \text{if } \omega_i^x \neq \infty \\ psInd_{i,t} = \left(g_{i,t}^d (\gamma_{i,t}^{xd} pd_{i,t})^{\omega_i^x} + m_dmgAgg_{dmgTrd,"i"} g_{i,t}^e (\gamma_{i,t}^{xe} pe_{i,t})^{\omega_i^x} \right)^{\frac{1}{\omega_i^x}} & \text{if } \omega_i^x \neq \infty \end{array} \right\}$	(147)
---	-------

Under the additive CET form, the price aggregator differs from the revenue exhaustion price which is written as follows:

$ps_{i,t} = \left(g_{i,t}^d (\gamma_{i,t}^{xd})^{\omega_i^x} pd_{i,t}^{\omega_i^x+1} + g_{i,t}^e (\gamma_{i,t}^{xe})^{\omega_i^x} pe_{i,t}^{\omega_i^x+1} \right) psInd_{i,t}^{-\omega_i^x}$	(148)
---	-------

Equation (149) defines the exports price $pe_{i,t}$ from given world market price. The FOB price of exports $pWe_{i,t}$ is multiplied by the exchange rate eR_t to convert into local currency terms on the left-hand side. There are two wedges that affect the domestic producer price of exports: an ad-valorem export tax $\tau_{i,t}^{Exp}$ and the domestic trade and transport margin $pMarg_{i,t}tmg_{i,e',t}$ from factory gate to the border.

$\chi_i^{pe} pe_{i,t} (1 + \tau_{i,t}^{Exp}) + pMarg_{i,t}tmg_{i,e',t} = eR_t \chi_i^{pWe} pWe_{i,t}$	(149)
---	-------

The model allows for both a downward sloping export demand curve and an upward sloping import supply curve. Equation (150) reflects the export demand curve, with the possibility of an infinite demand elasticity in which case the FOB price of exports is fully exogenous. The variable $pWe_{i,t}^{ROW}$ represents the exogenous world market price. If the export demand elasticity η_i^e is finite, export demand declines as the price of exports increases as a function of home country exports. Equation (151) provides an analogous treatment of import supply where Import supply increases with a rise in the price of imports into the home country (for example with a reduction in import tariffs). With an infinite supply elasticity ω_i^m , the price of imports is fixed at $pWm_{i,t}^{ROW}$.

$\left\{ \begin{array}{ll} pWe_{i,t} = \gamma_{i,t}^{PWE} pWe_{i,t}^{ROW} & \text{if } \eta_i^e = \infty \\ xe_{i,t} = \chi_{i,t}^e \left(\frac{\gamma_{i,t}^{PWE} pWe_{i,t}^{ROW}}{pWe_{i,t}} \right)^{\eta_i^e} & \text{if } \eta_i^e \neq \infty \end{array} \right\}$	(150)
--	-------

$\left\{ \begin{array}{ll} pWm_{i,t} = pWm_{i,t}^{ROW} & \text{if } \omega_i^m = \infty \\ xm_{i,t} + mdelst_{i,t} = \chi_{i,t}^m \left(\frac{pWm_{i,t}}{pWm_{i,t}^{ROW}} \right)^{\omega_i^m} & \text{if } \omega_i^m \neq \infty \end{array} \right\}$	(151)
---	-------

There are potentially three goods market equilibrium conditions as follows.

$$xd_{i,t}^s = xd_{i,t}^d$$

$$xe_{i,t}^s = xe_{i,t}^d$$

$$xm_{i,t}^s = xm_{i,t}^d$$

These equations are substituted away, and the model implementation only carries a single variable for each of the equilibrium variables $xd_{i,t}$, $xe_{i,t}$ and $xm_{i,t}$.

Factor market closure

Flexible nesting

Similar to the nested structure in the final demand and in the production function, MANAGE-WB depicts factor markets based on a flexible nesting structure. Exempt from this structure is the supply of capital when capital vintages are active, which is discussed below.

The factor allocation system considers the different factors fmp and supply by different institutions depicted by the set $instTot$. In the default case, one economy-wide aggregate agent $sumInst$ manages the allocation of economy-wide factor stocks to production activities. Factor prices received by the institutions (households, enterprises, government) are identical in this case, and their income from each factor is a share of the economy-wide factor income, according to the distribution factors $\chi_{inst,fmp,t}^{Fac}$.

Alternatively, if information of the sourcing of each institution's income for a factor by activity is available and used, institutions manage their own factor stocks, such that, for instance, average wages by labor type can differ across households, while they are still identical at the level of the individual activity. This additional detail makes sense for sluggish factor markets, only. Moreover, the factor stocks can be broken into different age vintages captured by the set vf . This allows, for instance, to render the allocation of new entrants into the labor force more flexible compared to the existing labor stock.

Pretax factor prices for primary factors are identical across vintages and only assigned to the old vintage, see (152). Technically, these equalities are introduced by a macro-operator in GAMS which replaces $pfmSup_{fma,fmp,instTot,vf,t}$ with $pxFac_preTax_{fmp,fma,old,t}$ in the equations relating to the factor supply.

$pxFac_preTax_{fmp,fma,old,t} = pfmSup_{fma,fmp,instTot,vf,t}$	(152)
---	-------

The link between factor supply and factor demand for each activity is defined by the equation (153) which adds-up factor demand over the capital vintages on the left-hand-side and factor supply over potentially multiple supply institutions and factor vintages on the right hand-side, changed by short-run changes in factor utilization rates driven by elasticities $\varepsilon_{fmp,a}^{fmpIntens}$ and changes in factor prices before taxes $pxFac_preTax_{fmp,a,t}$ in current compared to previous years. The elasticity, for instance, affects the capital utilization rates (which then impacts the activity specific depreciation rates) or changes in hours worked in specific sectors, while the number of people employed is driven by long-run elasticities.

$\sum_v xFac_{fmp,fma,v,t} = \sum_{instTot,vf} xfmSup_{fma,fmp,instTot,vf,t} \left(1 + \varepsilon_{fmp,a}^{fmpIntens} \left(\frac{pxFac_preTax_{fmp,a,t}}{pxFac_preTax_{fmp,a,t}} - 1 \right) \right)$	(153)
---	-------

The total supply of a mobile or sluggish, but not immobile primary factor $xfmSup_{fmp,fmp,instTot,vf,t}$ is defined by an iso-elastic supply schedule, where $atFmSup_{fmp,instTot,vf,t}$ is a shifter variable and $m_dmgAgg_{dmgSup,fmp}$ is an expression which adds-up potential shifters due to multiple Climate Change damage channels. The additional set fmp indicates the factor market, such as the labor

category to which a labor sub-nest belongs. That eases the implementation as the same sub-nest names can be used for multiple markets.

$x_{fmSup_{fmp,fmp,instTot,vf,t}} = m_{dmgAgg_{dmgSup,fmp}} atFmSup_{fmp,instTot,vf,t} \left(\frac{p_{fmSup_{fmp,fmp,instTot,vf,t}}}{p_{GDPmp_t}} \right)^{\varepsilon_{fmp}^{fmp}}$	(154)
---	-------

In case of immobile resources such as oil fields or fish stocks or other natural resources, the supply to individual activities, depicted by the set fma , is based on an iso-elastic supply schedule in case of increasing prices but exhibits strongly growing elasticities in case of dropping prices based on a sigmoid expression. The use rate of natural resources is hence flexibly adjusted if returns fall (lower extraction quantities of oil etc.).

$x_{fmSup_{fma,fmp,instTot,vf,t}} = \min \left[\left(\frac{p_{fmSup_{fma,fmp,instTot,vf,t}}}{p_{GDPmp_t}} \right)^{\varepsilon_{fma,fmp}^{fma}}, \left(\text{sigmoid} \left(\left(\frac{p_{fmSup_{fma,fmp,instTot,vf,t}}}{p_{GDPmp_t}} \right) - 0.75 \right) * 25 \right) - 0.001 \right] * 10 \left[atFmSup_{fmp,t} Y_{fma,fmp,instTot,vf,t}^{fmp} m_{dmgAgg_{dmg,fmp}} \right]$	(155)
--	-------

For fully mobile or immobile (sub)-markets, the total supply is based on the adding-up condition as shown in equation (156):

$x_{fmSup_{fmPar,fmp,instTot,vf,t}} = \sum_{fmChl \in fmPar} x_{fmSup_{fmChl,fmp,instTot,vf,t}}$	(156)
--	-------

Supplies to sub-nests, including to individual activities $fmChl$ under a sluggish parent nest $fmPar$ are a share of the parent factor nest, updated with the price relation between the child and parent nest:

$x_{fmSup_{fmChl,fmp,instTot,vf,t}} = Y_{fmChl,fmp,instTot,vf,t}^{fmp} x_{fmSup_{fmPar,fmp,instTot,vf,t}} \left(\frac{p_{fmSup_{fmChl,fmp,instTot,vf,t}}}{p_{fmSup_{fmPar,fmp,instTot,vf,t}}} \right)^{\omega_{fmPar}^{fmp}}$	(157)
--	-------

In this fully mobile case, component prices are equal to parent ones:

$p_{fmSup_{fmChl,fmp,instTot,vf,t}} = \sum_{fmChl \in fmPar} p_{fmSup_{fmPar,fmp,instTot,vf,t}}$	(158)
--	-------

In case of the sticky wage mechanism, wages in the current year are equal to last year's wage plus the sticky wage parameter times the relation between current and long run supply of the labor factor minus unity:

$ \begin{aligned} & pfmSup_{fmpAll,fmp,instTot,vf,t} \\ & = pfmSup_{fmpAll,fmp,instTot,vf,t-1} (1 \\ & + stickyWage_{fmp} \left(\frac{xfmSup_{fmpAll,fmp,instTot,vf,t}}{m_{dmgAgg} dmgSup_{fmp} atFmSup_{fmp,instTot,vf,t} \left(\frac{pfmSup_{fmpAll,fmp,instTot,vf,t}}{pGDPmp_t} \right)^{\epsilon_{fmp}^{fmp}}} \right. \\ & \left. - 1 \right) \end{aligned} $	(159)
--	-------

The economic-wide average price for immobile resources is defined by the exhaustion price:

$pfmSup_{fmpPar,fmp,instTot,vf,t} = \frac{\sum_{fmpChl} pfmSup_{fmpChl,fmp,instTot,vf,t} xfmSup_{fmpChl,fmp,instTot,vf,t}}{xfmSup_{fmpPar,fmp,instTot,vf,t}}$	(160)
---	-------

The exhaustion price indices for the sluggish case and non-additive CET cases are defined as:

$ \begin{aligned} & pfmSup_{fmpPar,fmp,instTot,vf,t} \\ & = \left(\sum_{fmpChl} \gamma_{fmpChl,fmp,instTot,vf,t}^{fmp} pfmSup_{fmpChl,fmp,instTot,vf,t}^{1+\omega_{fmpPar}^{fmp}} \right)^{\frac{1}{1+\omega_{fmpPar}^{fmp}}} \end{aligned} $	(161)
--	-------

The additive case, which guarantees adding up, defines the dual price aggregator on a similar formula which differs in the exponent and is used instead of $pfmSup_{fmpPar,fmp,instTot,vf,t}$ in equation (162).

$pfmSupD_{fmpPar,fmp,instTot,vf,t} = \left(\sum_{fmpChl} \gamma_{fmpChl,fmp,instTot,vf,t}^{fmp} pfmSup_{fmpChl,fmp,instTot,vf,t}^{\omega_{fmpPar}^{fmp}} \right)^{\frac{1}{\omega_{fmpPar}^{fmp}}}$	(162)
--	-------

For the additive case, the exhaustion price $pfmSup_{fmpPar,fmp,instTot,vf,t}$ is defined as follows and is used where this price index enters the equation above for a child:

$ \begin{aligned} & pfmSup_{fmpPar,fmp,instTot,vf,t} \\ & = \left(\sum_{fmpChl} \gamma_{fmpChl,fmp,instTot,vf,t}^{fmp} pfmSup_{fmpChl,fmp,instTot,vf,t}^{1+\omega_{fmpPar}^{fmp}} \right) pfmSupD_{fmpPar,fmp,instTot,vf,t}^{-\omega_{fmpPar}^{fmp}} \end{aligned} $	(163)
---	-------

In case that factor supply is defined at level of individual institutions, physical aggregation over institutions is assumed:

$xfmSup_{fmpChl,fmp,"sumInst",vf,t} = \sum_{inst} xfmSup_{fmpChl,fmp,inst,vf,t}$	(164)
--	-------

Differences from defining factor income at the level of individual activities and from revenues by institutions are captured by a correction factor. It captures, for instance, the impact of intensive margins elasticities (see also equation (28))???

$corrFac_{actfp,t} = \frac{facty_{actfp,t}}{\sum_{inst,vf} xfmUpdateFac_{actfp,inst,t} xfmSup_{actfp,actfp,inst,vf,t} pfmSup_{actfp,actfo,inst,vf,t}}$	(165)
--	-------

Capital markets without vintage capital specification

If no vintage differentiation is present for the production activities, the equations shown above for the allocation of factor supply apply also to capital markets. The link to the capital stock is based on the following equality which renders the shifter $atmSup_{cap,instTot,old,t}$ in equation (154) endogenous.

$kapS_t = xfmUpdateFac_{cap,instTot,t} atmSup_{cap,instTot,old,t}$	(166)
--	-------

The corresponding rental rate of capital $trent_t$ is determined as:

$trent_t = pfmSup_{cap,inst,t}$	(167)
---------------------------------	-------

The average price for capital in sum over institutions is defined as:

$pfmSup_{cap,inst,sumInst,t} = \frac{\sum_{inst} xfmSup_{cap,inst,t} pfmSup_{cap,inst,t}}{xfmSup_{cap,inst,sumInst,t}}$	(168)
---	-------

Capital markets with the vintage capital specification

This section describes the capital allocation mechanism among the production activities under the assumption of two vintages for each activity. This specification is an on-demand option. Capital market equilibrium under the vintage capital framework assumes the following:

- *new* capital is perfectly mobile across production activities a which ensures an economy-wide uniform rate of return to capital $trent_t$. *new* capital is used in the *new* vintage only.
- *old* capital in expanding sectors – i.e., sectors which attract *new* capital – has the same productivity as *new* capital. Accordingly, the rate of return to *old* capital in the expanding sectors is the same as the economy-wide rate of return to *new* capital.
- declining sectors which do not attract *new* capital release *old* capital. The released *old* capital is added to the economy-wide stock of *new* capital. The assumption here is that declining sectors will first release the most mobile types of capital which is, as being mobile, comparable to *new* capital, such as transportation equipment.
- The rate of return to capital in the declining sectors reflects sector-specific supply and demand conditions and drives the share of released *old* capital.
- *old* capital is used in the *old* vintage (technology).

The result of these assumptions is that if there are no declining sectors, there is one single economy-wide rate of return to capital, identical to the pre-tax marginal return to capital in all sectors. Otherwise, there will be a sector-specific rate of return to *old* capital in each declining sector.

The economy-wide rate of return on *new* capital $trent_t$ is derived from the market equilibrium condition in equation (169):

$kapS_t = \sum_{cap,a,v} xFac_{cap,a,v,t}$	(169)
--	-------

The stock of installed or *old* capital $oldCap_{a,t}$ at the beginning of the year for each sector is determined as the sum of capital use in the *old* and *new* vintages in the previous year corrected for depreciation:

$oldCap_{a,t} = (1 - depr_{a,t} m_{dmgAgg} dmgDepr_{a,t}) (xFac_{cap,a,old,t-1} + xFac_{cap,a,new,t-1})$	(170)
--	-------

Equation (171) defines the quantity of *old* capital used by the *old* vintage as a share $rrat_{a,t}$ of the total *old* capital stock of this activity. This share is upper bounded by unity as the *old* vintage cannot attract *new* capital.

$axFac_{cap,a,old,t} = rrat_{a,t} oldCap_{a,t}$	(171)
---	-------

Equation (172) shows that this share $rrat_{a,t}$ depends on an activity specific supply schedule of *old* capital driven by the ratio between the returns to capital in the *old* vintage $pxFac_{preTax_{cap,a,old,t}}$ and the economy-wide return to capital $trent_t$. The lower the return to capital in a declining sector, the more of its capital is released and absorbed by expanding sectors, according to the elasticity $invElas_a$. Released *old* capital thus adds to *new* capital generated from net investment.

$rrat_{a,t} = (pxFac_{preTax_{cap,a,old,t}} / trent_t)^{invElas_a}$	(172)
---	-------

The condition for the *new* vintage to be active is determined as follows. As $rrat_{a,t}$ reaches unity, $pxFac_{preTax_{cap,a,old,t}}$ must be equal to $trent_t$ in the supply schedule. Equally, if the *new* vintage is not active for an activity, its production can stem solely from the *old* vintage and only a part of *old* capital is used. In order to implement these two conditions, first an intermediate variable $rrat1_{a,t}$ is defined as the total capital demand by the activity over *old* capital:

$rrat1_{a,t} = \frac{\sum_{cap,v} xFac_{cap,a,v,t}}{oldCap_{a,t}}$	(173)
--	-------

The variable $rrat_{a,t}$ is then the minimum of 1 and $rrat1_{a,t}$:

$rrat_{a,t} = \min(1, rrat1_{a,t})$	(174)
-------------------------------------	-------

The condition above is either implemented with an upper bound on $rrat_{a,t}$ if the model is solved as a MCP or by a differentiable fudging function for the *min* operator when the model is solved as CNS or NLP.

Assuming that outputs of vintages are homogenous, output of the new vintage is implicitly defined by adding up total activity output over the two vintages:

$xp_{a,t} = xFac_{ftop,a,old,t} + xFac_{ftop,a,new,t}$	(175)
--	-------

The vintage specification does not capture "stranded assets" in the sense that *old* capital no longer contributes to production. Capital no longer in use in declining sectors is released and contributes with the same marginal productivity to the overall productive capacity of the economy as new investments in the specification above. The economic-wide costs of shrinking sectors are captured by reduced returns to the part of their *old* capital which stays in their own production. The size of these costs depends first on the parameter $invElas_a$. The larger this elasticity, the faster *old* capital will be released if returns to capital drop in a sector and the smaller the economy-wide cost is. Second, the more flexibly sectors can substitute capital against other production factors, the lower the cost of adjustment

as the necessary drop in returns to capital to reduce output becomes smaller. The usual assumption is that due to lock-in effects into specific technologies at the time of investments, capital substitution against other production factors is less flexible for the *old* vintage compared to the *new* vintage.

As *old* vintage might shed some of their capital at a lower price compared to the one of new capital, the average rent $trentAvg_t$ received is defined from total factor income and the total capital stock.

$trentAvg_t = \frac{facy_{cap,t}}{kapS_t}$	(176)
--	-------

This average rent $trentAvg_t$ defines also the price received by the owner of the capital stocks.

$trentAvg_t = pmfSup_{cap, cap, instTot, vf, t}$	(177)
--	-------

For each activity, the supply of capital aggregated over the vintages must be equal to the capital supplied:

$\sum_v xfac_{cap, a, v, t} = xfmSup_{a, cap, "suminst", "old" t}$	(178)
--	-------

Endogenous technical progress

On demand, the model can employ an extension on top of the capital vintages which assumes monopolistic competition. Resulting markup profits of firms are invested into R&D efforts which generate new varieties with higher productivity. A stepping on toes effect depicts that the chance to find new, more productive varieties decreases with higher R&D efforts. The number of new varieties $einn_{a,t}$ is accordingly depending on the R&D intensity in a sector $xRD_{a,t}$, exponent the stepping on toes effects $\varphi_{a,t}$ times the number of varieties at the benchmark $einn0_a$. If $\varphi_{a,t}$ is zero, there is no change in the number of varieties over time.

$einn_{a,t} = einn_{a,t0} xRD_{a,t}^{\varphi_{a,t}}$	(179)
--	-------

The R&D intensity $xRD_{a,t}$ is determined in equation (180) by the markup income, i.e., the sector specific $p_markup_{a,t}$, which depends on the strength of monopolistic competition, times the sales revenues of the new vintage $xFac_{fptop, a, new, t}$ $pxFac_{fptop, a, new, t}$ as depicted on the RHS. It is corrected by a number of terms which besides the number of expected innovations in the current year $einn_{a,t}$ are all exogenous.

$xRD_{a,t}^{1-\varphi_{a,t}} xRD0_a pfrd_t frontBGP_{a,t} frontier_{a,t}$ $= \frac{p_{markup_{a,t}} xFac_{fptop, a, new, t} pxFac_{fptop, a, new, t} * 1 / \frac{einn_{a,t}}{einn_{a,t0}}}{(1 + interestRD_t) / (1 + egrowth_{a-t}) - (1 - (1 + 2\rho_{en})einn_{a,t})}$	(180)
--	-------

The total R&D demand $xfRD_{f,t}$ is allocated to final demanders f in shares $chi_{f,t}$ (currently to f =investments with $chi_{f,t} = 1$). It is defined as the sum of the R&D intensity of the different sectors $xRD_{a,t}$, multiplied with their benchmark R&D demand $xRD0_a$, and the two frontier growth terms

$frontBGP_{a,t}$ $frontier_{a,t}$, see the similarity to the LHS in (171) which differs by the exponent on the R&D intensity.

$xfRD_{f,t} = \chi_{f,t} \sum_a xRD_{a,t} xRD0_a frontBGP_{a,t} frontier_{a,t}$	(181)
---	-------

Total R&D expenditures $yfRD_t$ are defined by multiplying the total quantity demanded $xfRD_{f,t}$ with the price indices for the final demand categories $pf_{f,t}$, using the allocated demands $xfRD_{f,t}$ as weights:

$yfRD_t = \sum_f xfRD_{f,t} pf_{f,t}$	(182)
---------------------------------------	-------

The price index for R&D services $pf_{f,t}$ is defined by exhaustion, i.e., total expenditures for R&D services $yfRD_t$ divided by the sum of the allocated final demands $xfRD_{f,t}$:

$pfRD_t = yfRD_t \sum_f xfRD_{f,t}$	(183)
-------------------------------------	-------

The distribution of total output $xp_{a,t}$ to the new and old vintages' v total production output $xFac_{fptop,a,v,t}$ is defined as follows based on CES cost minimization problem:

$xFac_{fptop,a,v,t} \lambda_{a,v,t}^{enCr} = (inn_{a,t}(1 + \rho_{en}) \forall new + (1 - obsol_{a,t}) \forall old) \frac{\sum_v xFac_{fptop,a,v,t0} xp_{a,t0}}{xp_{a,t0}} xp_{a,t}$ $\left(\frac{px_{a,t}/px_{a,t0} \lambda_{a,v,t}^{enCr}}{(1 + markup_{a,t}) pxFac_{fptop,a,new,t} / (1 + markup_{a,t0}) pxFac_{fptop,a,new,t0}} \right)^{\sigma_{xpvt}} \frac{variety_{a,t-1}}{1 - \rho_{en} e inn_{a,t0}}$	(184)
--	-------

The first line of the RHS depicts the share parameters. The old vintage's share is $(1 - obsol_{a,t})$ the share of varieties getting obsolete $obsol_{a,t}$. The new vintage's share is equal to the new varieties becoming available in current year time $inn_{a,t}$, times $(1 + \rho_{en})$. The term $\frac{1 - \rho_{en} e inn_{a,t0}}{variety_{a,t-1}}$ acts as one of two productivity shifters. Note that an increase in number of varieties $variety_{a,t-1}$ compared to the benchmark, will lead to a reduction in $px_{a,t}$ and, through this channel, a reduction in overall factor demand of both vintages required to produce their combined output $xp_{a,t}$. The second productivity shifter $\lambda_{a,v,t}^{enCr}$ is vintage specific and defined below. The term $\frac{\sum_v xFac_{fptop,a,v,t0}}{xp_{a,t0}}$ acts as a scaling factor to account for differences in summed up physical factor demand and output at the benchmark. This is necessary as both inputs (for instance, labor or land) and outputs (for instance, in the case of energy carriers) can be measured in physical units and not in constant currency. The last line in equation (184) depicts the impact of changes in unit costs of the vintages augmented by the markup $(1 + markup_{a,t}) pxFac_{fptop,a,new,t}$ on the output composition, relative to the average price $px_{a,t}$. To account for quantities measured in physical units where prices differ from unity, prices in t are scaled with their benchmark values $t0$.

The vintage specific productivity shifters $\lambda_{a,v,t}^{enCr}$ are not driven by model equations but dynamically determined in between simulation points. For the new vintages, they are defined from two frontier growth terms: the growth frontier _{a,t} – currently identical for all activities – and the activity

differentiated growth term $frontCr_{a,t}$. They determine the productivity growth of the new vintage in (185):

$lambda_{a,new,t}^{enCr} = \frac{frontier_{a,t}}{frontCr_{a,t}}$	(185)
--	-------

For the old vintage, the first term in the productivity shifter $lambda_{a,old,t}^{en}$ is a weighted average of past growth embedded in the old vintage stemming from frontier and last year's growth of this frontier $frontier_{a,t-1}$ times the number of new innovations $inn_{a,t-1}$, corrected for $(1 + \rho_{en})$. The growth for the old vintage is hence accelerated for activities with above average new innovations becoming available:

$lambda_{a,old,t}^{en} = \left(\frac{\left[\left((1 - inn_{a,t-1} (1 + 2\rho_{en})) \right) lambda_{a,old,t}^{en} \sigma_{ftop,a,old}^{-1} + (1 + \rho_{en}) inn_{a,t-1} frontier_{a,t-1} \sigma_{ftop,a,new}^{-1} \right]}{(1 + \rho_{en}) inn_{a,t-1}} \right)^{1/\sigma_{ftop,a,old}}$	(186)
--	-------

A similar weighting is done for $lambda_{a,old,t}^{cr}$ which captures the growth from $frontCr$:

$lambda_{a,old,t}^{cr} = \left(\frac{\left[\left((1 - inn_{a,t-1} (1 + 2\rho_{en})) \right) lambda_{a,old,t}^{en} \sigma_{ftop,a,old}^{-1} + (1 + \rho_{en}) inn_{a,t-1} frontCr_{a,t-1} \sigma_{ftop,a,new}^{-1} \right]}{(1 + \rho_{en}) inn_{a,t-1}} \right)^{1/\sigma_{ftop,a,old}}$	(187)
---	-------

The two shifters, symmetric to (188) define the overall productivity shifter $lambda_{a,old,t}^{enCr}$ for the old vintage in (184):

$lambda_{a,old,t}^{enCr} = \frac{lambda_{a,old,t}^{en}}{lambda_{a,old,t}^{cr}}$	(188)
---	-------

Finally, the number of innovations becoming active in t $inn_{a,t}$ is equal to the expected ones in $t - 1$:

$inn_{a,t} = e inn_{a,t-1}$	(189)
-----------------------------	-------

Which also drives the number of obsolete varieties $obso_{a,t}$:

$obso_{a,t} = (1 + 2\rho_{en}) e inn_{a,t-1}$	(190)
---	-------

Accordingly, a higher rate of new innovations drives up the replacement of old, less productive varieties by new ones. Note that besides replacing equation (169) by equation (174), the remaining equations of the standard vintage extension, i.e., (163) to (168), are also used with the endogenous technical progress module.

To understand the endogenous technological progress module, it is helpful to consider the case where all prices stay constant, and a sector's output grows according to rate $frontBGP_{a,t}$ $frontier_{a,t}$ in equation (181). In this case, the sectors' R&D intensity $xRD_{a,t}$ is kept constant at unity and the number of expected new innovations $enn_{a,t}$ does not change compared to the benchmark. Following (189) and (190), the relative share of obsolete and new varieties is also constant. The productivity shifters in (185) and (188) will also remain at benchmark levels, letting the sector's productivity grow with $lambda_{a,new,t}^{enCr} = \frac{frontier_{a,t}}{frontCr_{a,t}}$ for both vintages. If in contrast, changes in demand or other factors increase the output growth of a sector above the economy-wide average, its markup income will increase, driving up its R&D intensity. This will let the change in the number of new varieties exceed the economy-wide average, such that unit cost of production of the activity will decrease stronger than for others. These productivity gains might increase demand further. Besides the usual factors which prevent exploding demands for a product, such as inelastic final demands, the stepping on toes-effect will reduce the number of new varieties with increasing R&D efforts of a sector and thus prevent unrealistic high productivity gains.

User cost of factors of production

Equation (191) converts the market price of factors of production $pxFac_Pretax_{fp,a,v,t}$ to the user-cost of factors of production $pxFac_{fp,a,v,t}$ by sectors by adding the activity specific ad-valorem factor tax or subsidy. Equation (192) adjusts the factor tax by the GDP price level for reporting purposes.

$pxFac_{fp,a,v,t} = pxFac_preTax_{fp,a,v,t} (1 + \tau_{a,fp,t}^{Fac}) + \sum_{em} \tau_{em,a,t}^{Emi} emi_t^{cal} \chi_{em,fp,a,t}^{emiFac} \rho_{em,fp,a}^{emiFac} \phi_{em,fp,a}^{emiFac}$	(191)
$\tau_{a,fp,t}^{rFac} = \frac{\tau_{a,fp,t}^{Fac}}{pGDPmp_t}$	(192)

Foreign savings and balance of payments

The balance of payments (BOP) in MANAGE-WB consists of four elements: (1) the trade balance (2) foreign savings contributing to investments, (3) foreign debt related flows $fundF_t$, see chapter "New foreign government debt" and (4) transfers between the ROW and domestic institutions (enterprises, households, government), see chapter "Transfers".

The share of the sum of foreign contribution to savings and investments relative to GDP $savfGDP_t$ is defined in equation (193), taking the exchange rate eR_t into account as foreign savings $savf_t$ are defined in international currency:

$savfGDP_t = \frac{savf_t eR_t}{GDPmp_t}$	(193)
---	-------

For the sum of foreign contribution to savings and investments $savF_t$, different closures are available, driven by the scalar $capAccClosure$ and pre-defined acronyms. This offers a flexibility to analysts as different rules apply in different applications. The default case in equation (194) fixes the relation $savfGDP_t$, active if $capAccClosure$ is set to $savfEndoERFix$ and the relation between B.o.T. and GDP (variable $botGdp$) is not fixed.

$savF_t = savfGDP_t \frac{GDPmp_t}{eR_t}$	(194)
---	-------

The second closure (*capAccClosure* is set to *savfEndoGTAP*) is the one found in the GTAP model and equilibrates the normally exogenous global expected return to capital $rorg_t$ to the domestic one $rore_t$ such that the changes in the foreign contribution to domestic investment follow a given path of $rorg_t$:

$rore_t = rorg_t$	(195)
-------------------	-------

The third rule (*capAccClosure* is set to *savfEndoGTEM*) defines foreign savings relative to GDP as a share γ_t^{sf2} of depreciated capital, the depreciation rate $depr_t$ times the capital stock $kstock_t$, driven by a term depending on the relation between $rore_t$ and $rorg_t$ dampened by a term which considers the debt-to-GDP ratio $debtGDP_t$, the exchange rate eR_t and the GDP price index $pGDPmp_t$:

$savF_t/GDPmp_t = \gamma_t^{sf2} depr_t kStock_t e^{\rho_t^{savf} (rore_t - rorg_t) - \omega_t^{savf} \frac{eR_t}{pGDPmp_t} debtGDP_t}$	(196)
---	-------

A fourth rule (*capAccClosure* is set to *savfEndoResRegSavings*) gives the user flexibility in how the share of foreign savings on GDP reacts to the changes in capital rent $trent_t$ compared to a given global one $retROW_t$ and to the domestic per capita GDP growth $gGDPpc_t$ and the global one $gGDPpcROW_t$, by choosing appropriate parameters (γ_t^{sf1} , θ^{1a} , θ^{2a}):

$savF_t/GDPmp_t = \gamma_t^{sf1} \left(\frac{1 + trent_t}{1 + retROW_t} \right)^{\theta^{1a}} \left(\frac{1 + gGDPpc_t}{1 + gGDPpcROW_t} \right)^{\theta^{2a}}$	(197)
---	-------

A last rule, inspired by the USAGE model (*capAccClosure* is set to *savfEndoUSAGE*), drives the growth rate of the capital stock grk_t . It considers with the variable $devRor_t$ defined in equation (199) as the difference between domestic and global rate of returns to capital $rore_t$ and $rorg_t$ corrected for the country risk premium $rord_t$. This variable lets the growth rate of capital fluctuate in a corridor between grk_t^{min} and grk_t^{max} driven by a trend in capital growth grk_t^{trend} . The smaller χ_t^{grk} , the closer grk_t will follow the trend. This growth rate is defined in equation (200)

$grk_t = \frac{grk_t^{max} e^{\chi_t^{grk} devRor_t} + grk_t^{min} (grk_t^{max} - grk_t^{trend}) / (grk_t^{trend} - grk_t^{min})}{e^{\chi_t^{grk} devRor_t} + (grk_t^{max} - grk_t^{trend}) / (grk_t^{trend} - grk_t^{min})}$	(198)
$devRor_t = rore_t - rord_t - rorg_t$	(199)
$prodInv_t = kStock_t (grk_t + depr_t)$	(200)

Due to Walras' Law, one equation in the model can be dropped. The current implementation selects the balance of payments defined in equation (201) as the equation to remove. This equation does not restrict the model solution. Instead of forcing RHS to zero, the LHS defines an additional variable $Walras_t$ which represents the equilibrium of the balance of payments. This variable should be zero (considering the numerical accuracy achieved by the solver) if the simulation has succeeded – i.e., Walras' law holds. This provides a model consistency check for the various exhaustion conditions. The code will throw an error during simulations if the absolute value of this test variable exceeds a certain

quite small share of GDP. Such an error typically points to logical errors in the equation structure. Note that the balance of payments equation is evaluated in foreign currency units.

$ \begin{aligned} walras_t = & \sum_i \underbrace{\chi_i^{pWe} pwe_{i,t} xe_{i,t}}_{Exports} + \underbrace{\frac{savF_t eR_t}{Net\ investment\ flows}}_{Net\ investment\ flows} \\ & + \sum_{inst} \underbrace{transfers_{inst,row,t}}_{Inbound\ transfers} - eR_t \sum_t \underbrace{pWm_{i,t} (xm_{i,t} + mdelt_{i,t})}_{Imports} \\ & - \sum_{inst} \underbrace{transfers_{row,inst,t}}_{Outbound\ transfers} - \sum_{instf} debtPay_{instf,t} \\ & - \sum_{fmp,vf} xfmSup_{fmp,fmp,row,vf,t} pfmSup_{fmp,fmp,row,vf,t} corrFac_{fp,t} \end{aligned} $	(201)
--	-------

Macro identities

The following block of equations provides the main macroeconomic identities. Equations (202)-(204) determine respectively nominal aggregate stock building, real aggregate stock building and the stock building aggregate price index.

$Tdelst_t = \sum_i (\chi_i^{ps} ps_{i,t} delst_{i,t} + eR_t pWm_{i,t} (1 + \tau_{i,t}^{Imp}) mdelt_{i,t})$	(202)
$rTdelst_t = \sum_i (\chi_i^{ps} ps_{i,t0} delst_{i,t} + eR_{t0} pWm_{i,t0} (1 + \tau_{i,t}^{Imp}) mdelt_{i,t})$	(203)
$pdelst_t = \frac{Tdelst_t}{rTdelst_t}$	(204)

Equations (205) – (207) determine the nominal and real values for aggregate exports and the aggregate export price index. Aggregate exports are evaluated at FOB prices in domestic currency units.

$TotExp_t = eR_t \sum_i (\chi_i^{pWe} pWe_{i,t} xe_{i,t})$	(205)
$rTotExp_t = eR_{t0} \sum_i (\chi_i^{pWe} pWe_{i,t0} xe_{i,t})$	(206)
$pExp_t = \frac{TotExp_t}{rTotExp_t}$	(207)

Equations (208)-(210) determine respectively the nominal and real values for aggregate imports and the aggregate import price index. Aggregate imports are evaluated at CIF prices in domestic currency units.

$TotImp_t = eR_t \sum_i pWm_{i,t}(xm_{i,t} + mdelst_{i,t})$	(208)
$rTotImp_t = eR_{t0} \sum_i pWm_{i,t0}(xm_{i,t} + mdelst_{i,t})$	(209)
$pImp_t = \frac{TotImp_t}{rTotImp_t}$	(210)

Equation (211) defines nominal GDP at market prices, where the index fd in the sum covers the final demand accounts – household expenditures (h), government current expenditures (gov), private investment expenditures (inv) and government investment expenditures ($ginv$). Equations (212) and (213) define real GDP at market prices and the GDP deflator at market prices. Per capita real GDP is defined in equation (214) where $pop_{total,t}$ is total population. Equation (215) is relevant in dynamic simulations. In the baseline simulation, the growth in real GDP per capita $gGDPpc_t$ may be exogenous (i.e., targeted) and another variable would be endogenous to achieve the growth target, typically some factor productivity parameter (discussed further below). In policy or alternative scenarios, the growth in real per capita GDP would be endogenous.

$GDPmp_t = \sum_{fd} yf_{fd,t} + Tdelst_t + TotExp_t - TotImp_t$	(211)
$rGDPmp_t = \sum_{fd} xfd_{fd,t} + rTdelst_t + rTotExp_t - rTotImp_t$	(212)
$pGDPmp_t = \frac{GDPmp_t}{rGDPmp_t}$	(213)
$rGDPpc = \frac{rGDPmp_t}{pop_{total,t}}$	(214)
$gGDPpc_t = \frac{rGDPpc_t}{rGDPpc_{t-1}} - 1$	(215)

Equation (216) defines nominal GDP at factor cost as the sum of factor remunerations across all factors and activities of production – i.e., it does not include indirect taxes. Equation (217) defines real GDP at factor cost. It is a linearization of the true GDP function defined as the weighted sum of the factors of production in efficiency units where the weights are given by the relevant base year prices. The GDP at factor cost price deflator is defined in equation (218).

$GDPfc_t = \sum_a \sum_v \sum_{fp} pxFac_{fp,a,old,t} xFac_{fp,a,v,t}$	(216)
$rGDPfc_t = \sum_a \sum_v \sum_{fp} pxFac_{fp,a,old,t0} xFac_{fp,a,v,t}$	(217)
$pGDPfc_t = \frac{GDPfc_t}{rGDPfc_t}$	(218)

Additional macro identities provide more flexibility to define closure rules. Key macroeconomic shares, real and nominal government and investment expenditures as shares of GDP, are described in equations (219) and (220) respectively.

$rshrGDP_{oa,t} = \frac{xf_{d_{oa,t}}}{rGDPmp_t}$	(219)
$nshrGDP_{oa,t} = \frac{yF_{oa,t}}{GDPmp_t}$	(220)

If the fiscal responsibility option is chosen, the required changes are reflected in $fiscalRespDrv_t$ as in equation (221) where $debtGDPTgtUp_t$ and $debtGDPTgtLo_t$ are target debt-to-GDP ratios.

$fiscalRespDrv_t = \min \left(0, - \left\{ \text{sigmoid} \left(\frac{debtGDP_t - debtGDPTgtUp_t}{debtGDPTgt_t} \right)^{0.5} - 0.5 \right\}^2 \right) - \min \left(0, - \left\{ \text{sigmoid} \left(\frac{debtGDPTgtLo_t - debtGDP_t}{debtGDPTgtLo_t} \right)^{0.5} - 0.5 \right\}^2 \right)$	(221)
---	-------

If $fiscalRespDrv_t < 0$, the normally exogenous shares on GDP of final demand categories $rshrGDP_{f,t}$ for $f = \{gov, ginv\}$, i.e., government consumption and investments, become endogenous and are determined by equation (222). The term $relExpInc$ determines whether $fiscalRespDrv_t$ is solely affecting government expenditures for final consumption or also tax revenue. If $relExpInc \neq 100$, it affects the overall tax shifter $tmall_t$ as in equation (223).

$\frac{xf_{d_{f,t}}}{rGDPmp_t} = \frac{xf_{d_{f,t0}}}{rGDPmp_{t0}} + \frac{relExpInc}{100} fiscalRespDrv_t$	(222)
$tmall_t = tmall_{t0} - fiscalRespDrv_t$	(223)

Equation (224) defines the balance of payments as the difference between the value of export at f.o.b. price and imports valued a c.i.f. Its share on nominal GDP is defined in (225).

$bot_t = totexp_t - totimp_t$	(224)
$botGDP_t = \frac{bot_t}{GDPmp_t}$	(225)

Aggregate NIPA shares on real GDP are defined in equations (226) - (228), the last two defining export and import shares.

$gdpCompShr_{nipa,t} = \sum_{fd \in nipa} \frac{xf d_t}{rGDPmp_t}$	(226)
$gdpCompShr_{expo,t} = \frac{rTotExp_t}{rGDPmp_t}$	(227)
$gdpCompShr_{impo,t} = \frac{rTotImp_t}{rGDPmp_t}$	(228)

The consumer price index is defined in (229) from the households' Armington demand and related prices, where $pafIni_{i,h}$ are the benchmark prices:

$cpi_t = \sum_{i,h} \frac{xa_{i,h,t} paf_{i,h,t}}{xa_{i,h,t} pafIni_{i,h}}$	(229)
---	-------

Default closure rules

The default closure rules of the model are as follows:

- Aggregate real government expenditures on consumption and investment are fixed as a share of real GDP.
- The government balance is endogenous, if negative, it is closed by new debt, if positive, it contributes to investment.
- The foreign savings are fixed as a share of GDP. Transfers from the ROW are fixed relative to GDP, transfers to ROW from domestic institutions are driven by changes in their income as detailed above. This implies that BOT is a fixed share of GDP corrected for changes in transfer to ROW relative to GDP. The latter correction is usually quite small.
- The contribution of foreign savings to investment depends on the new government's foreign debt minus principal and interest payments on foreign debt.
- Investment is savings driven. Household, enterprise and government savings were discussed above. Household and enterprise savings are usually parameterized to react to changes in returns to capital. As foreign savings are fixed relative to real GDP as a default, investments and capital accumulation are largely influenced through household and enterprise savings and new domestic government debt¹⁴
- The exchange rate is fixed to benchmark level and serves as the model numeraire.
- In the baseline simulation, the default closure fixes real GDP and endogenizes a uniform productivity shifter for all types of labor, or for labor in service sectors to which labor productivity changes in other sectors are linked. The labor productivity changes can also be linked with a multiplicative factor to drive changes in land, natural resource and capital productivity.

¹⁴ Alternative closures are conceivable, for example targeting investment (as a share of GDP) and allowing the household savings schedule to adjust to achieve the target.

- In alternative (counterfactual or policy) simulations, the endogenous productivity shifters defined in the baseline are fixed at baseline level and real GDP becomes endogenous.

Note that additional productivity shifters might be introduced both in the baseline and counterfactuals, either as exogenous or based on closure swaps. The code allows to add in shock files equations to the set documented in here to target additional variables or to implement specific closure swaps.

Emissions and land cover module

The model is set up to include any number of emissions, indexed by em , generated by final product use and input use (both intermediate and factor demand – e.g., herd-size or land), and by output (e.g., the case of cement, or methane emissions from landfills). Related emission factors are in most cases taken from the GTAP database, based on mapping products and activities in the project specific SAM to GTAP products and activities. GHG related emissions reported in GTAP are first scaled to match given emission inventories (energy, transport, etc.) at the benchmark which typically stems from CAIT, before being assigned to the SAM accounts. Other emissions, such as air emissions, are typically scaled with the same factor as the GHG emissions.

Equation (230) determines the level of emissions $EMI_{em,emSrc,aa,t}^{comm}$ by emission type em from the (Armington) consumption of emission source commodity $emSrc$ (a subset of i) by Armington agent aa (intermediate and final demand). The basic coefficient is $\rho_{em,emSrc,aa}^{emi}$, representing the benchmark level of emissions per unit of consumption, for example, tons of CO₂ per ton of oil equivalent. The $\chi_{emi,emSrc,aa,t}^{emi}$ parameters allow for (exogenous) changes in the emissions coefficients that could be brought about by autonomous improvements in emission efficiency. If needed, emissions can be scaled by the emi_t^{cal} scale factor for accounting purposes. Equation (231) determines the level of emissions associated with the use of each factor of production for each activity. In livestock this could pertain to the size of the herds, for example for methane emissions, and for crops, it could be linked to land use. Equation (204) determines the level of emissions generated by overall output – i.e., linked to the volume of production, $EMI_{em,a,t}^{xp}$ for each activity. This can be used to assess the level of emissions, so-called process emissions, from certain sectors such as emissions in cement, around 50 percent of which are not linked to the use of fuels, or methane emissions from landfills. Equation (203) determines emissions due to land use change for each activity.

$EMI_{em,emSrc,aa,t}^{comm} = emi_t^{cal} \chi_{emi,i,aa,t}^{emi} \rho_{em,i,aa}^{emi} \chi_{a,i,aa,t}$	(230)
$EMI_{em,fp,a,t}^{fact} = emi_t^{cal} \chi_{em,fp,a,t}^{emiFac} \rho_{em,fp,a}^{emiFac} \sum_v xFac_{fp,a,v,t}$	(231)
$EMI_{em,a,t}^{xp} = emi_t^{cal} \chi_{em,a,t}^{emiXP} \rho_{em,a}^{emiXP} xp_{a,t}$	(232)
$EMI_{em,a,t}^{LUCF} = - \sum_{lnd} \rho_{em,lnd,a}^{emiLnd} (xFac_{lnd,a,v,t} - xFac_{lnd,a,v,t-1})$	(233)

Equation (234) computes emissions by the Armington agents which are aggregated to so-called emission inventories in equation (235). Equation (236) computes the real carbon tax by dividing

nominal carbon tax by the GDP price deflator. Equation (227) determines the total level of emissions, for emission em , summing over all sources-intermediate and final consumption, and factor- and production-based emissions as well as emissions coming from the changes in land cover categories (defined in equation (238)) converted into CO₂ using a factor of 3.37. Land cover area $lCovArea_{lnd,lcov,t}$ is defined in equation (239) where $crops$, $anim$ and frs are sets of activities containing crops, livestock and forestry, respectively, $exolCovAreaChg_{lnd,lcov,t}$ is exogenous and the set $lCovDrv$ is land cover types that drive changes in unmanaged land according to the land transition matrix $lndTransMat_{lnd,lCovDrv,lCov}$. Equation (240) defines $lCovArea_{lnd,"builtupLand",t}$, one of $lCovDrv$, as a function of GDP per capita and population. It allows for an exogenous level of autonomous emissions – that could, for example, come from an external model. A subset of emissions is linked with greenhouse gases (GHG) such as methane, nitrous oxides and fluoridated gases. Equation (241) calculates the total of GHG emissions where the individual gases are weighted by the so-called global warming potentials $EmiConv_{em,polltype}$.¹⁵

$EMI_{em,aa,t}^{seca} = \sum_i EMI_{em,i,aa,t}^{comm} + \sum_{fp} EMI_{em,fp,aa,t}^{fact} + EMI_{em,aa,t}^{xp} + EMI_{em,aa,t}^{LUCF}$	(234)
$EMI_{ghgInv,em,t}^{inv} = \sum_{i,aa} EMI_{em,i,aa,t}^{comm} p_{ghgInvAlloc_{ghgInv,em,i,aa,t}}$ $+ \sum_{actFp,a} EMI_{em,actFp,a,t}^{Fact} p_{ghgInvAlloc_{ghgInv,em,actFp,a,t}}$ $+ \sum_a EMI_{em,a,t}^{XP} p_{ghgInvAlloc_{ghgInv,em,"emiXP",a,t}}$ $+ \left[\sum_{actHa,lCov} dCarbon_{act} \right] \nabla ghgInv = LUCF$	(235)
$\tau_{em,aa,t}^{rEmi} = \frac{\tau_{em,aa,t}^{Emi}}{PGDPmp_t}$	(236)

¹⁵ See IPCC (1996), page 22 and also the Global Warming Potentials (IPCC Second Assessment Report) at http://unfccc.int/ghg_data/items/3825.php.

$EMITot_{em,t} = \sum_{aa} EMI_{em,aa,t}^{seca} - 3.37 \sum_{lnd} \sum_{lcov} dCarbon_{lnd,lcov,t}$	(237)
$dCarbon_{lnd,lcov,t} = carbCnt_{lnd,lcov}(lCovArea_{lnd,lcov,t} - lCovArea_{lnd,lcov,t-1})$	(238)
$\begin{aligned} lCovArea_{lnd,lcov,t} = & lCovArea_{lnd,lcov,t-1} + \sum_{crops} \sum_v (xFac_{lnd,crops,v,t} - xFac_{lnd,crops,v,t-1}) \\ & + \sum_{anim} \sum_v (xFac_{lnd,anim,v,t} - xFac_{lnd,anim,v,t-1}) \\ & + \sum_{frs} \sum_v (xFac_{lnd,frs,v,t} - xFac_{lnd,frs,v,t-1}) + exolCovAreaChg_{lnd,lcov,t} \\ & - \sum_{lCovDrv} (lCovArea_{lnd,lCovDrv,t} - lCovArea_{lnd,lCovDrv,t-1}) lndTransMat_{lnd,lCovDrv,lCov} \end{aligned}$	(239)
$lCovArea_{lnd,"builtupLand",t} = \left(\frac{rGDPmp_t}{pop_{total,t}} / \frac{rGDPmp_{t0}}{pop_{total,t0}} \right)^{0.33503} \left(\frac{pop_{total,t}}{pop_{total,t0}} \right)^{1-0.17634}$	(240)
$EMI_{polltype,t}^{GHG} = \sum_{em} EmiConv_{em,polltype} EMITot_{em,t}$	(241)

Emission taxes can either be set exogenously by fixing the emissions tax variables, $\tau_{em,aa,t}^{Emi}$, or can be endogenous under an emission cap, for example by fixing $EMITot_{em,t}$. Specific parameters (see the section on taxes) allow to define which emission sources are included in the tax scheme. Emission taxes increase government revenues and can interact with other taxes depending on the chosen closure for the government account.

Equation (242) shows the emission intensity of pollution types with respect to GDP.

$EMI_{polltype,t}^{Int} = \frac{EMI_{polltype,t}^{GHG}}{rGDPmp_t}$	(242)
--	-------

Climate change damage module

Climate change damages are implemented through shifters $m_dmgAgg_{var,dims}$ into the model where *var* is a set comprising the following damage variables such as $dmgPrd_{fpNest,a,chnl,t}$ and *dims* are indexes of each variable showing the initial impact points such as *fpNest* and *oa*. Each damage variable carries the index *chnl* depicting different channels through which climate change impacts the economy. The set of active channels can be defined by the analyst. This setup provides a flexible framework to integrate climate change damage in a structured way into the model. For example, $m_dmgAgg_{var,dims} = \prod_{chnl} (1 + var_{dims,chnl,t})$. Technically, it is done by a macro-operator in GAMS for the following cases:

$dmgPrd_{fpNest,a,chnl,t}$	Acts as a productivity shifter
$dmgYld_{fpNest,a,chnl,t}$	Changes the yield (output over factor use)
$dmgSup_{fpNest,chnl,t}$	Changes factor supply, for instance, capital, labor or land supply
$dmgDem_{oa,i,knest,chnl,t}$	Preference shifter

$dmgTrd_{i,chnl,t}$	Export shifter
$dmgStk_{*,chnl,t}$	Changes capital stock or population size
$dmgDepr_{a,chnl,t}$	Depreciation rates

The damages are defined as potentially endogenous variables. This allows for an analysis to introduce additional equations in a shock file that drive the damage variables as a function of others, for instance, the amount of adaptative investments. Without such additional equations, all damage variables are exogenous and act as fixed parameters.

Adaptive investment to reduce CC damages are defined in (243):

$x d_{adpinv,t} = adpInvShift_{f,t} + adpInvShr_{adpinv,t} \sum_{f \in adpInv} x d_{f,t}$	(243)
---	-------

Air pollution emissions and related damages

The air emission intensity of each sector a is defined by summing up over the set of air emissions ema in equation (244); equation (245) defines a similar metric for each household h .

$emiIntSeca_{a,t} = \sum_{ema} \frac{emiSeca_{ema,a,t}}{xp_{a,t}}$	(244)
$emiIntHH_{h,t} = \sum_{ema} \frac{emiSeca_{ema,h,t}}{yf_{h,t}}$	(245)

The exposed air pollution is defined by the total air pollution emissions over the last five years in equation (246). It drives the damages to the labor stocks l in the different sectors a due to reduced health with increasing air emissions in (247):

$exposedAirPoll_t = \sum_{tf < t-5} emiGhG_{PMFT,t,f}$	(246)
$dmgAir_{l,a,t} = -p_AirPollEff_{l,a,t} \log \left(\frac{exposedAirPoll_t}{exposedAirPoll_{t_0}} \right)$	(247)

Model Dynamics

Model dynamics are driven by three factors, similar to most neo-classical growth models: changes in the labor stock, the capital stock and primary factor productivity. The labor force growth rate is typically equated to the growth rate of the working age population, assumed as the population aged between 15 and 64. Population growth and hence also labor stock changes are exogenous by default. However, in most applications, a labor supply function will render the active share of the labor force for each labor stock endogenous depending on real wages.

During baseline construction, shifters introduced by the analyst might also let labor stocks' growth differ from changes in the working age population, for instance, to changes in composition by skill level.

The aggregate capital stock in any given year for each type of capital $kStock_t$ is equated to the previous year's capital stock less depreciation at a rate of δ_t plus the current period volume of productive investments $prodInv_t$:

$kStock_t = m_dmgAgg_{dmgStk, "cap"}(1 - \delta_{t-1}) kStock_{t-1} + prodInv_t$	(248)
---	-------

Total unemployment is defined as the population in working age minus the total number of employed persons in equation (249).

$unemp_t = pop_{15-64,t} - \sum_l xfmSup_{l,l,"sumInst",vf,t}$	(249)
--	-------

The last factor contributing to growth is changes in productivity. The default choice in baseline construction is one uniform labor productivity shifter which affects all labor types and activities to recover a given GDP growth path. Alternatively, the labor productivity of service sectors can be endogenized and linked to labor productivity in other sectors by a quadratic function of GDP growth.

The supply of other primary factors (land, water, natural resources, if present) is typically based on an iso-elastic supply scheme driven by a return to these factors. Reasonable choices for these elasticities suggest moderate elasticity values where large increases in rents can result if the economy grows strongly. The model, therefore, allows linking productivity changes in these sectors multiplicatively to changes in labor productivity.

In baseline construction, it is common to target the composition by sectoral aggregates. To do so, the model defines the total value added in equation (250) and shares of sectoral aggregates in total value added in (251).

$sectTot_t = \sum_{fp,a,v} xfac_{fp,a,v,t} pfac_pretax_{fp,a,v,t}$	(250)
---	-------

$sectShr_{agr,t} = \sum_{fp,a,v,a \in agr} xfac_{fp,a,v,t} pfac_pretax_{fp,a,v,t} / sectTot_t$	(251)
---	-------

Similarly, baselines might target the energy or electricity intensity of GDP, defined in equation (252) and (253) as the aggregate Armington demands for energy product e or electricity products $iElec$ relative to real GDP:

$nrgInt_t = \sum_e xat_{e,t} / rgdpmpt_t$	(252)
---	-------

$elyInt_t = \sum_{iElec} xat_{iElec,t} / rgdpmpt_t$	(253)
---	-------

Recursive dynamics versus limited forward-looking behavior

In the recursive-dynamic version, agents have rational expectations for product and factor prices and other variables relevant to their decisions taken in the current year. Optimal decisions, for instance, with regard to use and allocation of resources, are based on realized prices for products and factors

derived from market clearance. There are hence no surprises about market outcomes, planned is equal to realized production, including the consideration of stochastic shocks to production in the current (but not future) period. First order conditions hold at realized prices. But current year decision-taking is myopic as it does not consider expectations of future developments. This assumption can be challenged in cases where agents are informed about future changes in markets such as under announced upcoming tax reforms or expected future impacts of climate change.

The model, therefore, supports a modular extension where expectations for some key variables in MANAGE-WB are based on a simultaneous solution for a limited number of future years to overcome the limitation of the myopic solution. It establishes an intermediate option between a fully dynamic case where all future years are considered simultaneously and the recursive dynamic case which looks at one single year, only.

Considering solely a number of future years compared to perfect foresight over the full simulation horizon can be defended based on three arguments. First, with the increasing distance of a future time point to the current one, uncertainty on available technologies, preferences, policy instruments etc. increases. Simulated future outcomes become hence less certain the larger the distance to the current year which renders their information value for current decision taking less valuable, compared to outcomes for the near future. Second, expectations are mostly important for decisions with a lock-in effect which impacts the future decision space of an agent such as in the case of investments. But lock-in effects are not infinite but soften over time, for instance, due to depreciation. Accordingly, the impact of current decisions on future flexibility also decreases over time which gives a further argument to consider only a limited number of future years when forming expectations. Lastly, time preference for money implies that impacts of current decision on future outcomes receive lower weights the farther away the future time point.

The solution is based on “rolling window” approach where a number of future years besides the current year are solved for simultaneously, and before the current year is moved forward by one period and the considered future period is prolonged by one year as well. The module introduced stock changes, see chapter “Stock change behavior”.

3. Model Implementation

The code for MANAGE-WB is structured into different files and folders to support structured model development and maintenance as well as hosting the code on a version control system. The main folder can be named freely as all folder references in the code are relative to the main (= root) folder.

To simplify the use of one model version with several country specific projects, country specific input files are stored in project-specific sub-folders under the **reg** sub-folder. The project specific folders comprise the (potentially multiple) bridge files, implemented in EXCEL, which bundle all necessary information for the model: the SAM, additional data, parameters and sets. The file **load_base.inc**, when present, is read by the code to add non-default parameters and sets from the bridge file required in the shock file. In a sub-folder called **sim**, shock files are hosted in such a project folder. Shock files can also comprise declarations and definitions of additional variables and equations which are added to

the core model. This allows project specific changes to the model structure without changes to the core model code under version control.

The subfolders in a MANAGE-WB installation host specific type of files while the main or root level is reserved for **.gms** files that are the start point of GAMS project (called **main files** from here on). All files that are included in such a main file have the extension **.inc** (for include file) and are hosted in one of the sub-folders.

These sub-folders that organize the model files according to their type and use:

- \dat folder hosts the country independent data files
- \inc folder hosts the GAMS include files that are called by model code at various stages
- \inc\cal host the files related to benchmarking
- \inc\solve host files related to solving the model
- \inc\sensStoch host files related to sensitivity analysis and stochastic runs
- \inc\samest host files related to SAM balancing
- \inc\outout host files related to post-model output generation
- \res comprises sub-folders for the different countries which host the result **.gdx** files
- \sav file keeps all temporary files (e.g., debug **.gdx** files, saved model instances)
- \opt comprises solver option files
- \reg folder hosts sub-folders for the different projects
- \gui folders hosts files and sub-folders needed for the Graphical User Interface
- \jrexxx hosts a Java-Run-Time Engine needed for the Graphical User Interface

Model runs to construct a baseline or to run a counterfactual are started with **run.gms**. A counterfactual inherits the time series of variables (i.e., the model solutions) and parameters from a counterfactual, as well as all set definitions etc. Starting a counterfactual with an empty shock file will hence replicate the baseline results, however, different from baseline mode with GDP turned endogenous and exogenous productivity and preference shifters. Accordingly, shock files define changes in exogenous variables compared to the underlying baseline to render them transparent and should not repeat shocks already implemented in the baseline.

Figure 3.1: Folder Structure

Name	Date modified	Type ^	Size
 .git	1/4/2022 11:57 AM	File folder	
 dat	1/25/2022 11:15 AM	File folder	
 inc	1/25/2022 11:24 AM	File folder	
 res	1/26/2022 7:47 AM	File folder	
 sav	1/4/2022 11:58 AM	File folder	
 sim	1/4/2022 11:58 AM	File folder	
 irencgedf.gpr	1/26/2022 9:31 PM	GAMS IDE project	125 KB
 .gitignore	12/12/2021 4:28 PM	GITIGNORE File	1 KB
 bau.gms	1/25/2022 8:59 AM	GMS File	3 KB
 COMP.GMS	1/25/2022 8:54 AM	GMS File	4 KB
 Maketab.gms	1/25/2022 10:11 AM	GMS File	2 KB
 report.gms	12/15/2021 9:28 AM	GMS File	24 KB
 result.gms	1/25/2022 9:06 AM	GMS File	10 KB
 result0.gms	12/12/2021 4:28 PM	GMS File	10 KB
 sim.gms	12/15/2021 10:58 AM	GMS File	3 KB
 EqVaMatch_Reg.xlsm	12/12/2021 4:28 PM	Microsoft Excel M...	7,387 KB
 EqVaMatch_v2_3.xlsm	12/12/2021 4:28 PM	Microsoft Excel M...	22,573 KB
 path.op2	12/12/2021 4:28 PM	OP2 File	1 KB
 path.op3	12/12/2021 4:28 PM	OP3 File	1 KB
 path.op4	12/12/2021 4:28 PM	OP4 File	1 KB
 path.opt	12/12/2021 4:28 PM	OPT File	1 KB
 varList.txt	12/12/2021 4:28 PM	TXT File	7 KB
 runall.cmd	12/15/2021 9:30 AM	Windows Comma...	1 KB
 vscode.cmd	12/13/2021 4:33 AM	Windows Comma...	1 KB

Folder and File Structure

Table 3.1 lists the core files used for model implementation.

Table 3.1: Subfolders and files under main model folder

Name	Description
./	Main Folder
run.gms	Runs business as usual scenario and simulations
run_stoch.gms	Runs a set of stochastic counterfactuals
Runs_sens.gms	Runs sensitivity analysis
samest.gms	SAM updating, balancing and splitting
gtapSAMv6.gms	Generates as SAM and other data from the GTAP Data Base
fromGuiToRun.gms	Includes a file generated by the GUI, overwritten with each run
.gitignore	List files, folders and extension to be ignored by GitHub (defines files not under version control)
.gitattributes	Internal files used by GitHub
manage.bat	Batch file which opens the GUI
loadGtap.gms	Converts a zip container from the GTAP website with a GTAP data base in GDX format to a zip container in MANGE-WB conventions
/dat	Subfolder for project independent data files
BridgeFile_GTAP_POW_template.xlsx	Template bridge file for GTAP derived SAMS (used by gtapSAMv6.gms)
BridgeFile_GTAP_CE_template.xlsx	Template bridge file for GTAP derived SAMS using the GTAP-CE data base (used by gtapSAMv6.gms)
BridgeFile_GTAP_9.xlsx	Template bridge file for GTAP 9 derived SAMS (used by gtapSAMv6.gms). Thought for testing with the freely available GTAP 9 version which is shipped with the model

cc_damages.gdx	CC damage estimates by Roson and Sartori (2016) converted into polynomial functions based on a regression analysis
debtData.gdx	Data on foreign debt stocks and related payments (used by gtapSAMv6.gms)
faoProd.gdx	Physical production output in 1000 tons for selected products from FAO production statistics (added to bridge-file by gtapSAMv6.gms)
gdld_2014_V10.gdx	Gender Differentiated labor data by sector and GTAP country, matching GTAP V10 (used by gtapSAMv6.gms)
gdld_2017_V11.gdx	Gender Differentiated labor data by sector and GTAP country, matching GTAP V11 (used by gtapSAMv6.gms)
ghg.gdx	GHG emission inventories from CAIT
gtapData9.gdx	Basic data from the now freely available GTAP database 9 with benchmark year used by gtapSAMv6.gms. run.gms uses the file to define physical land use, emission inventories and carbon stocks. GTAP data requiring a license can be brought in the expected format with "loadGtap.gms"
mfmodGdp.gdx	Macro-economic projections from MFMOD, June 2023 (added to bridge-file by gtapSAMv6.gms)
nutrients_iso.gdx	Estimates calories, protein and fat contents per USD (added to bridge-file by gtapSAMv6.gms)
SAM_GTAP_POW_template.xlsx	Template SAM file for GTAP derived SAMS (used by SAMEST)
splitSharesWater.gdx	Shares to split up the land account in GTAP V11 crop sectors to land and irrigation water, based on CGEBox results (used by gtapSAMv6.gms)
unpop.gdx	Demographic projections by the UN
/inc	Subfolder for files included in main files
base.inc	Declares sets and parameters; reads data from excel file, part of run.gms and run_stoch.gms
fromFrEndExcel.in	Reads in shocks from EXCEL sheet, part of run.gms and run_stoch.gms

fromFrEndReadExcelSheet.inc	Reads in shocks from EXCEL sheet, part of run.gms and run_stoch.gms
fromInvest.inc	Read shocks from INVEST
load_base.inc	Example file on how to add parameters to bridge-file, can be copied to reg folder
model.inc	Model sets, equations, variables and parameters, part of run.gms and run_stoch.gms, run_sens.gms
model_def.inc	List of paired equations. Converted with Python code to model_def_copy.inc for unpaired version.
opt.inc	Some core definitions such as folders and globals related to model extensions, part of run.gms, run_stoch.gms a
readIEC.gms	Example of a main reading CC damage information from CSV files provided by the IEC group
RSCCDamages.inc	Processes CC damages by Roson and Sartori (2016), part of run.gms (optional on CC damages on)
title.inc	Helper code to update title bar in the GUI, used by all programs
title_def.inc	Definition for the title helper code, part of run.gms
toInvest.inc	Generate CSV file for INVEST (option on INVEST link), part of run.gms
cal/calCDE.inc	Calibrates the parameters for the CDE demand system
cal/inical.inc	Includes initial calibration of parameters and variables, part of run.gms and run_stoch.gms
cal/iniEmis.inc	defines emissions from GTAP data base, emission inventories and distribution factor from emissions to inventories
cal/rebench.inc	re-calibrate some constant terms in equations in sensitivity analysis where elasticities are updated, part of run.gms
cal/rescale.inc	scale variables back to original scaling of SAM, part of run.gms
ouput/generateSets.inc	Generate GDX with sets for HTML documentation, part of run.gms

output/reportDecl.inc	Declaration of sets and parameters used in toGui.inc, part of run.gms (optional on GUI output is on)
output/samCalc.inc	Calculates SAM from model results for each period, part of run.gms (optional on SAM output is on)
Output/toGui.inc	Calculates the result parameter for the reporting backend, part of run.gms (optional on GUI output is on=
output/toExcelFrEnd.inc	Prepare for output to EXCEL, part of run.gms
output/toExcelFrEnd2.inc	Output parts of model solution which are subject to standardized closures swap to EXDEL, part of run.gms
samest/samStat.inc	Defines statistics on the SAM during balancing, part of samest.gms
samest/macroSam3.inc	Aggregates SAM from Macro-SAM, part of samest.gms
samest/mactotal_eg_agg.inc	Aggregation of macro-SAM, part of samest.gms
sensStoch/genPut.inc	Generate code to apply LHS draws to specific parameters, part of run_sens.gms
senstoch/iniStoch.inc	Generate nodes for stochastic experiments, part of run_stoch.gms
sensStoch/lhs.exe	External program for LHS draws, part of run_sens.gms
sensStoch/Lhs_to_GDX.gms	GAMS program called by run_sens.gms to convert output from lhs.exe to GDX, part of run_sens.gms
sensStoch/LHSManual2004.pdf	Manual for lhs.exe
sensStoch/runStoch.inc	Helper code when running stochastic simulations, part of run_stock.gms
sensStoch/sens_proc	Program which defines relations between linearized factors in LHS and actual symbols shocked, part of run_sens.gms
shock/balgrowth.inc	Balanced growth check
shock/ccdr.inc	Master shock file for CC related ready-to use shocks
shock/ccdmglEC.inc	Example shock file to introduce CC damages from IEC

shock/empty.inc	Completely empty shock, used by tests to check if no-shock recovers the benchmark
shock/fixEnergy.inc	Part of ccdr.inc, fixes energy demands and defines necessary closure swaps
shock/fixSectSpecFac.inc	Example shock file how to fix factor demand in a sector and swap with an endogenous factor tax
shock/GHGMitigation.inc	Part of ccdr.inc, carbon taxation, carbon tax recycling and MACC mechanism
shock/none.inc	Empty shock file for tests or to start a new one
shock/numtest.inc	Shock exchange rate in all years and debt stocks / new debt in t0 with a 10% increase to check for no money illusion
shock/output_keyind.inc	Example file for defining an additional output file. Defines key indicators (can be used with run_stoch.gms)
shock/redGhg.inc	Code which fixes GHG emissions and swaps with an endogenous carbon tax, part of ccdr.inc
shock/shk_drm.inc	Example for stochastic damages, to be used with run_stoch.gms
solve/checkInfes.inc	Prints lists of infeasible equations and variables hitting their bounds, part of run.gms and run_stoch.gms
solve/checkModelSolution.inc	Checks and report on model solutions statistics
solve/iniVars.inc	Construct a start point from last year's results
solve/iterloop.inc	Drives model dynamics and closures, part of run.gms and run_stoch.gms
solve/kill_equ.inc	Removes equations symbols from past years to reduce memory load, part of run.gms and run_stoch.gms
solve/prepSolve.inc	Defines current years in model, runs iterloop.inc, and define standard model attributes
solve/resetNegatives.inc	In between repeated solves, resets variables back to plausible ranges, part of run.gms and run_stoch.gms
solve/resLastYear.inc	Resets all variables levels to the previous year values (used for the solution algorithm), part of run.gms

solve/runShortRunShock.inc	Set parameterization for short run shocks
solve/setBoundsForCns.inc	Sets bounds for repeated solves
solve/rescale.inc	Rescales quantities and values back to the level of the original SAM, part of run.gms
/opt	Subfolder with option files for the solvers (PATH, CONOPT4)
/res	Subfolder to write results, with regional specific folders (generated automatically)
/sav	Subfolder with temporary files that user does not need to access directly
/reg	Subfolder with country specific directories

Note: all files used by **run.gms** are also used by **run_stoch.gms** and **run_sens.gms** which as mother processes call multiple model runs as child processes based on **run.gms**.

Main Files

The **.gms** files in the main folder organize the model code for different purposes by calling files from other folders. **run.gms** runs the baseline and other scenarios. **run_stoch.gms** runs a set of stochastic scenarios on a grid. **run_sens.gms** runs a sensitivity analysis on a grid. **samest.gms** runs the code to update a SAM against macro data or a macro-SAM, re-balance it and split the SAM further on demand. **gtapSAMv6g.gms** generates bridge-files from the GTAP data to be inputted into **samest.gms** for further processing, such as splitting, or to used directly with **run.gms**, **run_stoch.gms** or **run_sens.gms**.

Data input files

All other input data, sets and maps for MANAGE-WB are read directly from different sheets in EXCEL file found in **reg\xxx\BaseBridge_xxx_v??\.xlsx** where **v??** denotes the version number and **xxx** denotes the country specific folder. Such an EXCEL file is called a bridge-file as it provides the bridge between the template, data driven structure of MANAGE-WB and the region and project specific model implementation. The bridge-file is hence the way to set up an actual implementation of MANAGE-WB for a specific country (and project). It must follow certain conventions as detailed next.

The **layout** sheet (Figure 3.2) in the bridge-file keeps the information on where each set, map or parameter to feed into MANAGE-WB is found. The first column in this sheet defines the type of GAMS symbol to read (*set* for multi-dimensional maps, *dset* for one-dimensional sets and *par* for parameters), the second column defines the name of the GAMS symbol, and the third column describes its location in the workbook. The three following columns define the number of rows, column and total dimensions of the symbol, the last being optional. Based on this information, a GAMS utility converts the content of the entries in the layout sheet into a GDX container from which they are inputted in **base.inc**. Wrong ranges or other errors in settings up the **layout** sheet can lead to not or partial input of symbols or

can let crash the conversion process. It is therefore recommended after changes to the layout sheet to carefully check the generated GDX container in the **sav** folder. In many cases, errors in there will throw errors at compile or run time, but some others, such as reading only a part of a parameter table might instead lead to faulty parameterization of the model and implausible results.

The user can add data on sheets or new sheets to read additional data, for instance into shock files. Such additional loads can be performed by placing a file **"load_base.inc"** in the project specific folder under **reg**. This requires adding rows for such new symbols in the **layout** sheet.

Figure 3.2: Layout sheet

Comment	Data Type	Name	Location	Row dimension rdim	Column dimension cdim	Total dimension dim
Mapping from activities to GTAP	set	mapaGTAP	Maps!A6	2		
Mapping from products to GTAP	set	mapcGTAP	Maps!D6	2		
All accounts (plus nests)	dset	is	Sets!B6	1		
Energy products	dset	e	Sets!G6	1		
Debt account	dset	debt	Sets!S6	1		
Labor accounts	dset	l	Sets!H6	1		
Capital account	dset	cap	Sets!I6	1		
Land accounts	dset	lnd	Sets!J6	1		
Natural resources	dset	natres	Sets!K6	1		
All investment account	dset	inv	Sets!L6	1		
Government investment account	dset	ginv	Sets!M6	1		
Adaptive investment account	dset	adpinv	Sets!N6	1		
Stoch changes account	dset	stb	Sets!O6	1		
Rest-of-the-World	dset	row	Sets!P6	1		
Energy shifter nest	dset	nrgFac	Sets!T6	1		
Maps from tax types to tax accounts	set	maptax	Maps!J6	2		
Aggregation from proto to final SAM	set	mapis	Maps!G6	2		
Demand bundle composition	par	mapkNestShr	mapk!B5	1		1
Substitutional elasticities in demand nests	par	sigmacNest0	mapk!C2			1
Map from aggregated to single sectors	set	mapaagr	Maps!M6	2		
Production nesting	par	pnest	pnest!B6	2		1
Substitution elasticities for production nests	par	subselas	pnest!C2	1		1
Factor market nestings	par	fmrNest	fmrNest!B9			
Parameters for factor market nests	par	fmrNestPar	fmrNest!B2			
Age cohorts	dset	cohorts	Sets!Q6	1		
Initial SAM	par	SAM0	SAM!A1	1		1
default parameter values	par	defPar	defPar!B2	1		
exceptions in activity parameters	par	actPar	actPar!A1	1		2
exceptions in commodity parameters	par	comPar	comPar!A1	1		1
exceptions in HH parameters	par	hohPar	hohPar!B9	1		2
exceptions in other parameters	par	othPar	othPar!A1	1		1
labor quantities for initial SAM, aggregated to model SAM	par	labor0	labor!B1	1		1
Emission inventories to overwrite CAIT	par	emiInventory_OFF	emiInv!A2	1		1
Power mix	par	powerMix	powerMix!A1	2		1
GHG mitigation	par	GHGMitigation	GHGMitigation!A1	2		1
Energy projections	par	energyProj	Energy!A1	3		1
Nutrient contents	par	nutCont	Nutrients!A1	1		1
Points on Marginal Abatement Cost Curves	par	macc	macc!A1	1		1
Regression parameter to render CDE income dependent	par	regparIncomeCDE	hohPar!A3	2		1
GDP composition (for targeting)	par	GDPComp	GDPCComposition!A1	2		1
Factor supply distribution across institutions	par	xfrmSupDistrInst	hohPar!B15	2		1
endogenous tech change parameters	par	EnTPar	EnTech!G5	1		
stepping on toes effect	par	PhiRDoes	EnTech!A5	1		
supernormal growth of sectors	par	Supfgrrowth	EnTech!D5	1		

The **sets** sheet contains all the one-dimensional sets that are read into the model, while the **maps** sheet comprises two- and three-dimensional ones. Some sets are directly read from columns or rows of maps and parameters and do not need to be defined a second time in the sheet. The file **bridgefile_GTAP_POW_template.xlsx** in the **dat** folder can be used as a starting point for developing a new bridge-file as it comprises all necessary sets, maps and parameters for a default baseline construction.

The second row in the **sets** sheet uses a formula to get the set name from the fifth row, while the third row shows the column name automatically, again using a formula. These two rows are used in the

Figure 3.3: The sets and maps sheets

The mapping of activities and commodities to GTAP, as shown in Figure 3.3, serves three purposes. First, it allows the GAMS code to derive factors for GHG and air pollution emissions from the global GTAP data if available for the country. Second, the maps can be used in SAM to split up the proto-SAM based on GTAP information. Third, it defines standard sector aggregates for reporting (agriculture, manufacturing, services).

The **defpar** sheet in Figure 3.4 comprises a longer list of core model parameters. Details on these parameters as listed below can be found in the related section of the model equation documentation.

Figure 3.4: defpar sheet

Elasticity between demand bundles other final Demand	sigmaf0	1.01
Elasticity between savings and total consumption	omegaSav0	0.2
Elasticity between savings and transfers to HHs	omegaSavEntr0	0.05
Elasticity between government and private bonds	omegaFund0	4
Elasticity of capital supply to capital stock	epsRor0	eps
Elasticity of expected return on capital to capital stock	epsRorE0	0.9
Share of investment that is not added to capital stock	PhiPcInv0	0.75
Depreciation rate	depr0	0.05
Capital stock GDP ratio	capoutRate0	5
Initial interest rate (one plus interest rate)	interest0	1.1
Elasticity of labor productivity to air pollution	AirpolEff0	0.025
Elasticity among commodities when an activity produces multiple commodities	omegap0	0.20
Disinvestment Elasticity for declining sectors	invElas0	0.7
Elasticity among activities when a commodity is produced by multiple activities	sigmas0	inf
Elasticity between imports and domestic demand	sigmam0	0.054
Elasticity between exports and domestic supply	omegax0	3
Elasticity between margin commodities	sigmamg0	0.01
Import supply Elasticity	omegam0	inf
Export demand Elasticity	etae0	inf
debt to gdp default threshold	omegasavf0	0.5
positive scaling parameter	rhosavf0	10
Country risk premium	riskPrem0	1
expected return rate in the row	retrow0	0.05
expected per capita GDP growth rate in the row	ggdppcrow0	0.02
efficiency param. returns	eta10	7
efficiency param. GDP p.c. growth	eta20	7
efficiency param. returns	eta1a0	7
efficiency param. GDP p.c. growth	eta2a0	7
Elasticity of foreign new public debt to interest rate	sigmaFInt0	1
Threshold after which gov consumption is dampended	debtGdpTgt0	0.5
Default income elasticity for aggregate commodity group (overwritten by hohpar)	incElas0	0.735
Default price elasticity for aggregate commodity group (overwritten by hohpar)	prcElas0	-0.54
Exchange rate Dom Currency per USD	USD	1
Foreign Debt Stock	DebtStkF0	0
Domestic Debt Stock	DebtStkD0	0
Interest on foreign debt	DebtIntF0	3.84
Interest on domestic debt	DebtIntD0	8.20E+00
SAM scale factor	inscale	1.00E-03
Population Scale factor	pscale	1.00E-03
Emissions scale factor	escale	1.00E+00
Elasticity of stock changes with regard to relative delta in expected price	rhoPDeISt0	0.5
Elasticity of stock changes to targeted stock over production	rhoXDeISt0	0.25
Target stock over production level	alphaSt0	eps

The **pnest** sheet defines the nested production structure and comprises two blocks. The first block defines the substitution elasticity for each parent nest. *f-top* is the fixed name for total output. The

second block defines the nesting shares. It comprises the child nests in the rows. The first column shows if the information along the row applies to all (= label all) or certain activities. The proposed default nesting for a GTAP-derived SAM assumes a Leontief-relation between the value added *va* and intermediate *nd* bundles as child of *f-top*. This is indicated by *eps* as a zero would not be read. Equally, Leontief relations are assumed inside the intermediate bundle *nd*. Child nests of this bundle (*crops*, *anim_meat*) allow instead for substitution between individual intermediate inputs. Note also in the example below that the value-added bundle *va* comprises the child nests *cap_ener* and *wtr_land* which combine primary factors with intermediates or bundles thereof.

Figure 3.5: Example of a part of a *pnest* sheet

The user enters non-default activity related input parameters in the **actpar** sheet. This allows to overwrite the defaults for three different parameters for each activity: *invElas0* (related to the release of old capital stock from the old vintage, in default **defpar**), *omegap0* (transformation of production to multiple outputs, default in **defpar**), and the substitution elasticities for the *old* and *new* vintages for the different nests (defaults in **pNest**). Empty cells do not imply zero elasticities, but rather that the defaults will be used. Entering *eps* will overwrite the default with a zero.

Figure 3.6: *actpar* sheet

	va	va	cap_labsk_ener	cap_labsk_ener	depr	ener	physProd	capital	labor	
	Old	New	Old	New				intensMarj	intensMarg	Elas
a-PDR	0.195804	0.522144	0.1	0.8	0.038662		2451.707	0.05	0.1	
a-WHT	0.195804	0.522144	0.1	0.8	0.038662		50.467	0.05	0.1	
a-GRO	0.195804	0.522144	0.1	0.8	0.038662		7873.18	0.05	0.1	
a-V_F	0.195804	0.522144	0.1	0.8	0.038662			0.05	0.1	
a-OSD	0.195804	0.522144	0.1	0.8	0.038662			0.05	0.1	
a-C_B	0.195804	0.522144	0.1	0.8	0.038662		3060.61	0.05	0.1	
a-PFB	0.195804	0.522144	0.1	0.8	0.038662		111.4687	0.05	0.1	
a-OCR	0.195804	0.522144	0.1	0.8	0.038662			0.05	0.1	
a-CTL	0.195804	0.522144	0.1	0.8	0.096654		8643.198	0.05	0.1	
a-OAP	0.195804	0.522144	0.1	0.8	0.057993		579.954	0.05	0.1	
a-RMK	0.195804	0.522144	0.1	0.8	0.096654		2301.638	0.05	0.1	
a-WOL	0.195804	0.522144	0.1	0.8	0.096654		7.309457	0.05	0.1	
a-FRS	0.15	0.4	0.1	0.8	0.038662			0.05	0.1	
a-FSH	0.15	0.4	0.1	0.8	0.038662			0.05	0.1	
a-COA	0.15	0.4	0.1	0.8	0.038662			0.05	0.1	

Figure 3.8: compar sheet structure

	sigmas0	sigmam0	omegax0	sigmamg0	omegam0	etae0	alphaSt0	rhoPdelSt0	rhoXdelSt0
c-pdr		5.05					0.4		
c-wht		4.45					0.4		
c-gro		1.3					0.4		
c-v_f		1.85					0.4		
c-osd		2.45					0.4		
c-c_b		2.7					0.4		
c-pfb		2.5					0.4		
c-ocr		3.25					0.4		
c-ctl		2							
c-oap		1.3							
c-rmk		3.65							
c-wol		6.45							
c-frs		2.5							
c-fsh		1.25							
c-coa		3.05	0.05						
c-oil		5.2	0.05						
c-gas		17.2	0.25						
c-oxt		0.9							

Table 3.3: User-specified commodity elasticities

Name	Description
sigmas0	The CES aggregation elasticity for the make matrix
sigmam0	CES (Armington) elasticity between domestic and imported goods
sigmax0	CET transformation elasticity between domestic and export markets
sigmamg0	The substitution elasticity for the 'production' of trade and transport services
etae0(i)	Export demand elasticity
omegam0(i)	Import supply elasticity
alphaSt0(i)	Target stock to output ratio
rhoXdelSt0(i)	Elasticity of stock changes to deviation from target stock level
rhoPdelSt0(i)	Elasticity of stock changes to price relative deviations between realized and expected prices

The **hohpar** sheet comprises income (*incElas0*) and own-price elasticities (*prcElas0*) for the households which are used to calibrate the CDE parameters, as well the elasticity which update the saving rate (*omegaSav0*). Note that the regularity conditions such as adding up can imply differences between calibrated and targeted elasticities as entered in the sheet. The sheet can also comprise elasticities to update the expansion terms of the income elasticities depending on income changes.

Figure 3.9: *hohpar* sheet structure

	incElas	incElas	incElas	incElas	incElas	prcElas	prcElas	prcElas	prcElas	prcElas	omegaSav0
	food	Manu	Ener	Serv	Tran	food	Manu	Ener	Serv	Tran	
h-hhold	0.885	1.161556	0.85	1.1	0.695	-0.5	-1.1	-0.3	-1.25	-1.25	

Table 3.4: User-specified household demand related elasticities

Name	Description
incelas0	Household and commodity bundle specific targeted income elasticity
incelas0	Household and commodity bundle specific targeted own-price elasticity
omegaSav0	CET transformation elasticity between savings and consumption

Similarly, the **mapk** sheet defines the shares of products and sub-demand nests for final demands to their parents and the related substitution elasticities. Parent nests are shown in the columns, child nests in the rows. Parents which are not also child nests will become the bundles present in the top-level CDE demand nest of the households for which parameters must be inputted in the **hohpar** sheet. For other final demand, a CES top level demand system is used and the share parameters for these bundles are calculated automatically in the benchmark code. Note that each child nest, including single products, needs to be assigned to at least one parent nest and that the sum of the shares must be one.

Figure 3.10: Example of part of a *mapk* sheet

	Crops	MeatDairy	FoodOther	Clothing	Manu	Hot_Trad	Serv	Tran	tranEner	TranRest	enerGas	enerFossil	enerEle	HousEner
	0.5	0.25	0.25	0.25	0.25	0.2	0.25	eps	eps	0.25	0.1	0.25	0.1	0.25
	Crops	MeatDairy	FoodOther	Clothing	Manu	Hot_Trad	Serv	Tran	tranEner	TranRest	enerGas	enerFossil	EnerEle	HousEner
c-pdr	1													
c-wht	1													
c-gro	1													
c-v_f	1													
c-osd	1													
c-c_b	1													
c-ocr	1													
c-ctl		1												
c-ctlRais		1												
c-oap		1												
c-rmk		1												
c-fsh		1												
c-cmt		1												
c-omt		1												
c-mil		1												
c-trd		0.05	0.1	0.1	0.05	0.6				0.1				
c-vol			1											
c-pcr			1											
c-sgr			1											
c-ofd			1											
c-b_t			1											
c-pfb				1										
c-wol				1										
c-tex				1										
c-lea				1										
c-wap				1										
c-frs					0.5									0.5
c-oxt					1									
c-lum					1									

A set of additional parameters is defined in the **othPar** sheet. The user can add columns to define non-default substitution parameters for demand nests.

Table 3.5: User-specified other parameters

Name	Description
sigmaf0(f)	Substitution elasticity between commodity bundles in other final demand (government, investments)

The **GDPComposition** sheet keeps inter alia the GDP growth projections for the baseline and can host projections of the VA composition by broad sectors (here Manu, Serv, agri, mine, cons, hema) as well as of GDP composition (Government consumption 'govCons', net exports 'netExp', foreign savings 'savf').

Figure 3.11: Example of part of a GDPComposition sheet

		2017	2018	2019	2020	2021	2022	2023	2024
bas	Manu	0.13022814	0.13022814	0.13022814	0.13022814	0.13022814	0.13022814	0.13022814	0.1
bas	Serv	0.40855384	0.40855384	0.40855384	0.40855384	0.40855384	0.40855384	0.40855384	0.4
bas	GovCons	0.0879181	0.0879181	0.0879181	0.0879181	0.0879181	0.0879181	0.0879181	0.
bas	netExp	-0.07015892	-0.07015892	-0.07015892	-0.07015892	-0.07015892	-0.07015892	-0.07015892	-0.0
bas	savf	0.05698296	0.05698296	0.05698296	0.05698296	0.05698296	0.05698296	0.05698296	0.0
bas	agri	0.26585163	0.26585163	0.26585163	0.26585163	0.26585163	0.26585163	0.26585163	0.2
bas	mine	0.01807101	0.01807101	0.01807101	0.01807101	0.01807101	0.01807101	0.01807101	0.0
bas	cons	0.14393702	0.14393702	0.14393702	0.14393702	0.14393702	0.14393702	0.14393702	0.1
bas	hema	0.03335834	0.03335834	0.03335834	0.03335834	0.03335834	0.03335834	0.03335834	0.0
bas	GDP	6.79	5.44	5.80	2.00	4.28	5.34	5.70	

The **SAM** sheet keeps the SAM. The SAM is a key element in the data driven concept underlying MANAGE-WB. The user can aggregate the SAM defined in this sheet based on the *mapis* map (in the maps sheet) to the products, sectors, factors, tax instruments etc. used in the project.

Figure 3.12: Example of part of a SAM sheet

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
SAM 2019, million LC																			
		a-agr0	a-frs0	a-coa0	a-oil0	a-ox0	a-fod0	a-b_r0	a-text0	a-wod0	a-p_c0	a-chm0	a-bph0	a-cm0	a-L_s0	a-fmp0	a-ele0	a-mvh0	a-met0
a-oh0																			
a-ins0																			
a-oxg0																			
a-edu0																			
a-hht0																			
a-crp0	73534565	305318	26.1817	13.347726	2.48167	18986623	1781639	1.4E+07	18512	12.876553	24487.96	48289.2	17625.464				5.69536	2710.26	118804
a-frs0	260664.02	1644040	10858.6	4.6467703	8813.897	7634.345	36.5284		739075	160.15249	297420.7	7.355	40713.219						41085.6
a-coa0	4242.4125	1770.22	159.954	4.7318754	278.4959	167.2382		3451.95			12181.96		17958.402	10004.57	13.7626				3.63326
a-oil0	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	8.488.6	#####	#####	#####	#####	#####	6.723.5	9.243.2
a-ox0	38252.686	2389.54	9481.77	623906.64	727799.7				43.3678	14127.651	400847.1	16.716	868102.29	6503588	9397.77				334.55
a-fod0	3512450.8	37478.3					6351279	568678	12669.6	21550291	24367.32	21.2234	889.67974	462.49126	232.605	10220.1	21.1054	6.83548	
a-b_r0	477010.35	81.3332					216282.8	824506	665.449	9.7565044	130.8222	10.0295	260.08276	51456589	12.4709	61.1564		98.64	
a-text0	77589.608	6588.68	2370.63	669.41703	14065.75	40372.94	10710	1.4E+07	718.206	4862.5801	44789.18	485.852	168287.84	102392.7	4980.74	14912.2	873.23	28682	
a-wod0	339823.8	7712.37	38771	1096.083	45341.6	516370.1	360800	222433	1195887	352.52476	61250.86	28332.5	965160.02	26327.78	8175.28	96569	7016.91	78363.6	
a-p_c0	2100146.3	114191	62006.7	174759.4	216328.8	93371.6	37021.7	80317.7	20680.8	30152619	484573.6	1845.35	210902.43	56382184	5544.51	75145.7	22323.3	87535.9	
a-chm0	2382307.6	57123.8	15185.5	39157.383	133258.7	222637.1	329402	1328959	184331	52484.447	8508093	64602.9	3023575.3	471412.79	68520.5	117318	96004.1	113674	
a-bph0	2714363.48	4143.03	47.0177	15.239804	5.742587	108851.7	4830.96	547.73	40.0145	436.2853	54963.33	747512	52421.963	719.1067	79.7734	43.3258	8.60294	64.9137	
a-cm0	1290221.8	26307.9	61250.8	17845.199	424387.5	388874.1	715894	804035	15397.7	20267.188	181508.1	15349	3728939.5	1704171.4	268561	176373	98899	1213665	
a-L_s0	253088.02	4174.43	11233	107880.52	444695	17773.39	1886.62	120359	852.467	5627.574	25194.92	1069.91	904887.62	12633769	2197545	1735965	1657481	1626517	
a-fmp0	1386174.1	28430.3	7988.38	166727.01	262386.8	145151.4	397442	148597	64628.4	3490.5371	72577.16	687.425	491574.88	527180.02	825246	204677	261721	570795	
a-ele0	509752.21	17003.6	14974.6	9630.2062	476643.4	129306	67630.5	79842.6	141802	14639.23	60040.45	1502.04	232352.55	474848.58	145614	3266080	116344	295967	
a-mvh0	906922.37	20013.4	17965.2	247523.46	233052.1	111261	153775	323634	74398.5	16678.294	462631.3	960.004	153505.87	660336.51	36022.3	93827.8	628265	259350	
a-met0	2247121.27	5289.49	2245.74	11301.939	63176.36	46824.74	12339.5	33448.6	9523.98	2074.785	20328.14		48647.781	397344.9	1664.33	14503.3	9116.71	2.2E+07	
a-Trn0	69467.115	2980.06	7942.12	356464.45	28626.44	73486.16	16647	108416	23752	69661.006	179826.9	7977.67	156443.53	424287.04	24338.5	68056.4	33639.9	32481.2	
a-ely0	202094.66	8787.74	23128.5	1050340	84175.32	215045.7	49221.3	316613	63950.6	205142.49	520872.9	23435.4	455705.51	1223351.8	71343.1	188662	98698	95178.1	
a-wtr0	16896.742	5531.8	594.895	263.59159	9850.643	32723.35	5085.51	50706.6	45340	694.32297	31361.8	2098.95	89072.823	462393.68	49130.6	21029.4	24176.1	7494.83	
a-cns0	150493.64	21719.1	3425.62	2470.432	32592.56	24102.2	23592.4	13463.1	181219	67722.641	36007.65	197.72	364795.63	43559.019	7267.93	8560.35	2365.65	18437.6	
a-trd0	96169.059	39850.9	578.313	38491.194	9401.698	115960	41623.3	16132.8	5544.11	1622.9745	17881.72	599.715	32080.334	82372.32	1359.36	27632.5	21431.6	44346.2	
a-otp0	35374.324	12823.6	9.98644	5978.452	73426.97	235708	48085.4	59451.7	11282.6	8364.0409	11917.3	1054.54	70950.73	51197.3	10620.8	19620.8	19416.9		
a-wp0	512.85475	76.3406	423.631	158.246	24.8168	259.9764	2649.85	490.897		324.62677	10492.14	264.516	4720.1647	1096.4015	117.025	444.362	3038.76		
a-afu0	63758.442	1239.65	0.85562	4466.195	1224.881	6242.49	10093.7	4969.23	58.4142	5393.4191	30603.7	225.775	7343.2777	6150.1008	309.01	2761.15	963.364	7412.83	
a-oxr0	1000694.8	67286.9	28481.3	245741.32	117631.7	318952.3	94491.3	188580	288888	8879.3144	489069.6	15081	238596.17	126263.79	63942.7	260633	71942.9	1020364	
a-cmn0	6673.1553	1678.87	85.5221	15600.861	12504.33	14624.91	4909.24	11259.1	2487.98	1698.7006	44035.86	720.672	23394.308	27343.291	12211.9	4073.66	10757.5	15427.2	
a-oil0	969654.38	22496.9	23957.1	1324694.6	348543.3	489810.1	71122.2	504425	97932.7	19980.752	362322.1	43080.8	550047.28	867105.76	112123	152318	103373	408812	
a-ins0	107779.84	2064.47	1902.18	147023.53	63629.63	60255.05	8322.41	29304.8	2462.01	2030.4693	32259.69	1711.81	58137.163	87467.618	3724.25	14122.5	8570.21	32804.9	
a-oxg0	1829.1777	91.041	505.294	592.1703	62.18138	4291.398	740.237	2954.63	324.666	700.20115	2232.756	1022.41	45916305	10991496	3024.48	2387.5	4943.66	6880.96	
a-edu0	3663.8129	594.857	809.4	5375.7786	48.31329	738.2904	1806.44	505.564	145.852	448.39236	3797.962	107.13	36214146	12791376	334.132	286.964	23.2372	2964.18	
a-hht0	4664.9531	96.0765	308.238	407.39353	198.512	2413.894	1436.42	423.397	16.3916	384.2828	4093.965	210011	1970.0661	5194.2105	250.885	451.729	19.8882	892.577	
f-lsk0	325706	9035	27504	208450	111524	82620	71089	419516	70200	74765	215435	41519	352617	539606	72228	146610	35141	244261	
f-ssk0	3435768	110086	135421	1110162	584149	988033	411644	2245548	368626	393948	1145894	146913	1864786	3226873	374699	520867	182344	1271478	
f-lusk0	487164.76	10461.5	3860.54	30287.806	15520.21	104671.6	13374.2	61182.3	9639.39	10426.587	31377.81	1124.79	50205.651	97826.245	101314	4021.31	4876.12	35624.1	
f-lsk0	37832.985	988.325	1752.22	13381551	7011.463	9136.535	4635.5	26988.4	4394.68	4694.0253	13849.6	1533.54	22410.834	40766.272	4564.86	5470.43	2208.38	15752.6	
f-lis0	5532166.1	128042	4757.72	386896.95	197037.8	1272032	176792	782410	123220	133454.95	400551	10680.7	639229.99	1249221.8	128181	38170.7	6189.5	451177	
f-lis0	187663.24	3694.72	2586.14	21198.948	10451.26	37711.63	7933.32	42596.4	6488.46	7053.6253	21982.1	186.518	34517.777	83987.334	6858.69	666.603	3270.17	24873.9	
f-capo0	11568423	4223369	136329	31149375	3396045	8110101	2396449	9107351	1218033	2149618.4	5829036	743227	7311865.6	28255582	1794808	1868391	838193	3677925	
f-Ind0	63957155																		
f-Ind0	34084447																		
t-tlsa0	541407.61	24656.7	4338.93	206936.33	48388.82	138185.7	47621.7	205337	29254.8	61800.399	113220.2	11627.7	134078.19	345395.04	33620	55798.8	26381.7	201780	

The **emiinv** sheet is intended to keep emission inventory data and used to scale GTAP emission factors to these inventories. The **nutrient** sheet stores calories, protein and fat supply per USD.

Include files

The **inc** folder comprises all non-project specific information that are called by the main files. Inc files thus host the generic parts of the model code which are not in the mains. Currently these files are organized to separate declaration of sets and parameters used in a run, code for model benchmarking and loading the baseline information for a counterfactual, variables and model equations, recursive dynamics and reporting.

Project specific changes to these files should be avoided to ensure a smooth use of the versioning system. Modifying the code of files under version control will easily provoke so-called conflicts when model developers commit code to extend the model or correct bugs. Changes to the equation structure of the model can be introduced via the shock file. Additional data can be loaded in shock files or by providing the **load_base.inc** file in the project folder.

Main settings

opt.inc file sets the main settings for the model such as activating optional modules, defining time periods etc. Global variables that manage the folder structure and defined names of output files are also declared in **opt.inc** file.

Files for benchmarking

Benchmarking is the process of defining certain model parameters such that the observed *ex post* data, mainly found in the SAM, are replicated by the model under a no-shock experiment in the base year. This model calibration starts with declaring the necessary sets, maps and parameters in the **base.inc** file. Afterwards, the data from the bridge-file as an Excel workbook is translated to a *gdx* file that is stored in the **sav** folder under the same name. Errors in the **layout** sheet, such as wrong cell references, will trigger errors in the conversion such that the *gdx* file is not generated. This typically provokes many follow-up errors as necessary symbols cannot be inputted. This *gdx* file derived from the bridge-file can be inspected, for instance, with GAMS Studio to check if the translation has worked properly. The code of **base.inc** defines then the sets and reads afterwards the parameters using so-called domain checking - i.e., it makes sure that the labels used to define elements in sets, maps and parameters are found stem in the sets used in the symbol declaration. This excludes, for instance, that typos in the bridge-file go unnoticed.

After loading the project specific data tables, each parameter is assigned to the corresponding initialization parameter in the model, data from generic global databases such as UN population projections are added. The user can add data to **base.inc** by providing a file called **load_base.inc** in the project specific *reg* folder.

After reading in the data, calibration of the parameters and assigning initial values to variables from the SAM and other satellite accounts is done in the **inical.inc** file. Like most CGE models, model parameters are calibrated such that the model implementation is able to re-produce the base dataset defined by the SAM and auxiliary data, such as on land use and emissions. The input to the calibration procedure is the base year SAM, potentially some satellite accounts such as energy balances, population, etc., and a set of key parameters (mostly elasticities). More formally the model can be written as:

$$F(y, x, \theta, \Omega) = 0$$

where y is a multi-dimensional set of endogenous variables, x is a multi-dimensional set of exogenous variables such as tax rates, and the parameter set is divided into two: θ contains parameters to be calibrated for benchmark and Ω contains fixed key (or user-specified) parameters. A typical simulation is then solving F for y with x , θ and Ω being fixed where x deviates from its base value, for instance, tax rates might have changed. In the calibration phase, y and x are taken from the SAM and other data, the parameters Ω are inputted via the bridge-file and function F is used to solve for θ :

$$F(y_0, x_0, \theta, \Omega) = 0$$

In practice, calibration can be mostly done recursively for block of equations without the need to invert the full equation system F formally. Exemptions are the calibration of the CDE demand (see **inc/cal/calCDE.inc**), balancing the GTAP emissions against given inventories (see **inc/cal/iniEmis.inc**) and calibrating factor supply shares.

The variables that are exogenous by default in the model, such as tax rates, emission and damage coefficients or productivities, are already fixed in the **inical.inc** file to keep the closure part of the **iterloop.inc** file simpler. If these variables need to become endogenous due to closure swaps specific

to a counterfactual, the lower and upper bounds need to be set to minus and plus infinite in the shock file by the user.

Repeating the benchmarking for normal simulation runs with unchanged elasticities is not necessary. Sensitivity analysis can however comprise changes in elasticities such that without repeated benchmarking of the related constant terms would keep the model from replicating the observed base year data. The code in **inc/cal/rebench.inc** addresses these cases.

Recursive Dynamic files

Most recursive dynamics in the model code are hosted in the **iterloop.inc** file. **iterloop.inc** is triggered only in the second and subsequent years of simulations. It first provides start values for the next solve by using some 'rule of the thumb' such as adding the GDP growth rate to previous year's result by calling the **resLastYear.inc** file. Start values should not change the simulation results, but can affect model convergence and speed, and in worse case, prevent the model from solving altogether. For counterfactuals, the user can alternatively use the given baseline results or the results from the last period as the starting point. Using baseline results might speed up solution time when the differences between the baseline run and the counterfactuals are limited over the full simulation horizon. **resLastYear.inc** also initializes the vintage capital structure for the first year by deriving the old capital from the benchmark capital use in each sector. Initial values, reflecting the SAM and satellite account for the base year, and lagged variables used in certain equations are also fixed in **iterloop.inc**. Capital accumulation and population dynamics together with some rules to update variables between periods exogenously are also done in the **iterloop.inc** file. Then different default closure options are implemented based on whether the model run is for the baseline or a counterfactual, depending on the **ifCal** parameter. **resLastYear.inc** is also used to reset the variable values in case of a model convergence failure, before attempting to solve the model with a different solver option file. **iterloop.inc** also updates on-demand the share parameters of the old vintages to represent a weighted average of the previous year's old and new vintage, similarly, updates share parameters of the old labor force stock and can shift parameters in the CDE demand system on-demand. It also comprises update of certain parameters in the endogenous technical progress extension. The user can on demand activate the code of **inivars.inc** which recursively defines from the start point values of certain values start values for other based on the model's equation structure. This can help in case of hard to solve shocks as the number of infeasible equations and the sum of their infeasibilities are reduced, but will often rather slow down solution.

If the shock file in use implements certain non-standard closure swaps, providing a file **release.inc** in the project specific folder might be needed to problems resulting from applying default closure rules in **iterloop.inc**. Such a file is automatically called if present in the project specific folder and should add the necessary statements to re-define bounds, such as freeing some variable instances. In case the first solve trial does not find a feasible solution, repeated trials when using PATH as a solver try different solver settings. In case that the model is solved as a constrained system of equations with CONOPT4, the second solve will switch to NLP mode by adding a dummy objective. This activates different algorithms in CONOPT4 which are usually slower, but able to deal better with numerical issues, for instance due to small pivots.

Model Files

The **model.inc** file contains variables and equation declarations, equation and model definitions and declarations of parameters and sets in the model. To help read the code, parameters are pre-fixed with **p_**. The listing of the model equations follows the same order as in this document and should allow for a relatively transparent comparison of the model write-up and its implementation in GAMS. The file is structured by blocks, such as production, income, government, consumption. Each equation is conditioned on the current solve year (i.e., if t is an element of ts) and other controls that drop equations in case where corresponding variables are not active. The **model_def.inc** file finally declares the core model as paired – i.e., each equation is matched to the corresponding variable. It can be used with a MCP solver to introduce complementarity conditions such that hitting a bound on a variable turns the related equation into an inequality. Using the MCP solver with the paired model version is also helpful in case programming errors lead to a so-called non-square model – i.e., the number of non-fixed variables differs from the number of equations. The pairing information is neglected when a non-MCP solver is used. Code in **model.inc** uses Python code to automatically generate from the paired a non-paired version in the model, stored in **model_def_copy.inc**. The non-paired version can be useful when the MCP solver is used with shock files using closure swaps.

Files used to solve the model

After a start point for the model is defined for the current year to solve and variables are fixed or freed according to the active closure swap, **run.gms** will attempt to solve the model. The code of **inc/solve/prepSolve.inc** defines the current years in the model (one for the default, multiple ones with the forward-looking extension) and sets some default properties for the first solve. As a default, the first solve will use CONOPT4 in CNS mode (Constrained Non-Linear System of Equations). This will immediately fail if the model is non-square – i.e., if the number of non-fixed (or endogenous) variables is not equal to the number of equations. If the model is square, the maximal wall-clock time in seconds and number of iterations defined on the GUI can be used by the solver to find a solution. If this fails, the code will try repeatedly to solve the model. If the sum of infeasibilities after the first try is small, the second try will start from this solution. Otherwise, the start point will be reset based on a call to **iterloop.inc**. Subsequent solves will switch to DNLP mode in CONOPT4 which is usually slower than CNS but numerically more stable, they will also use solver option files where the feasibility tolerance is slightly relaxed. These repeated solves will also use different sets of bounds on some model variables (see **inc/solve/setBoundsForCns.inc**) to keep the solver away from regions where zero divisions and other math trap errors could occur. This is supported by calling **inc/solve/resetNegatives.inc** between solves that moves implausible values from the last solve into a range where numerical issues are unlikely. If the model is solved as an MCP, different option files are tried in repeated attempts to solve the model.

If no solution is found after the last solve – i.e., if there are remaining infeasibilities or the Walras test fails, **run.gms** throws an error and aborts further execution. The related checks and reporting on model statistics are comprised in **inc/solve/checkModelSolution.inc**. Before the abort, the intermediate results are stored in a **gdx** file comprising the string “debug”. It can be inspected to find suspicious developments in endogenous variables and implausible levels for fixed ones. To help the analysis in case of infeasible equations, **inc/solve/checkinfes.inc** will report the first 100 infeasible equations and

the first 100 cases where non-fixed variables hit their bounds. The latter should not happen as bounds should solely stabilize the solution behavior and not restrict the model.

When the model is successfully solved, the code of **run.gms** will clean up the memory every third year by setting marginals and levels of equations back to defaults in **inc/solve/kill_equs.gms**. This file will also remove bounds which had been introduced by **run.gms** to stabilize the solution behavior. The next solve of the model will not use an in-memory call to the solver to trigger a re-organization of the internal memory of the GAMS base engine. This keeps the overall memory load of MANAGE-WB lower.

If the short-run extension is used, the model is solved twice in each year: first with the default parameterization and a second time with a different set of lower transformation and factor supply elasticities in factor markets, while elasticities for changes at the intensive margins (hours per worked, capital utilization rate) are introduced. The change in the parameterization is defined in **inc/solve/runShortRunShock.inc**.

Reporting Files

Model results are now mostly explored based on the reporting backend of the GUI. The necessary information for the GUI reporting is assembled in **inc/output/toGui.inc** which populates the parameters p_toGUI . It will also generate an XML file comprising the list of sectors, products, emissions etc., which is copied under the project name to the folder **gui/xml**. Relating parameters and sets defining are **inc/output/reportDecl.inc**. The **samCalc.inc** file builds a SAM from the model results using variable values for each year in a simulation. This helps in spotting possible errors, especially those that cause a non-zero Walras check.

Simulation Files

Simulations with the model use the same core GAMS file **run.gms** as employed for benchmarking. Simulation (or shock) files are generally project specific and therefore stored in the country specific folders in the **sim** folder. Such a shock file comprises of two parts, the declaration section which defines any additional symbols used in the shock file, and a transformation part which is called by the main file **run.gms** in each year. The following empty no-shock file shows this concept and gives an example on how a shock file can change the equation structure, here related to the pairing.

```
$iftheni.decl "%1"=="decl"

$setglobal modelVariant "rgdmp"

Model cge_%modelVariant%
/
  cge_
*
* --- implement BaU closure by changing pairing
*
  - rgdpmpeq
    rgdpmpeq.chixFacL
/

$else.decl

$endif.decl
```

Shock files can be used to cover a set of variants of a certain type of shock. For instance, a parameter read from a GDX container and inputted in the table in the declaration part could provide different sets of productivity changes, for instance, climate change damages for different future climate. The user can then via a global defined on the user interface (see next) determine which future climate to use for the specific counterfactual. Shock files are also regularly used during baseline generation to address cases which cannot be handled based on the features available from the GUI.

The model code comprises a set of shock files under **inc/shockFiles** which are copied to the project specific folder under reg. They relate partly to the CCDR interface, others can be used as examples.

4. Graphical User Interface

Introduction

The Graphical User Interface (GUI) of MANAGE-WB serves mainly two purposes. First, it allows users to set run specific options, such as which files to use, to avoid editing text files. Besides a touch and feel closer to modern software, this also avoids frequent changes to files under a version control. Second, the GUI generates reports to view and analyze model results, potentially comparing three or more scenarios which can represent baselines or counterfactuals.

The code underlying the GUI is the package GGIG (GAMS Graphical Interface Generator in Britz (2014) realized in Java of which license free versions portable across some operation systems are available. A free version of a Java run-time engine under the GNU license is shipped with the MANAGE-WB installation such that no additional Java installation is necessary. GGIG as a package has been used for more than a decade with different economic modeling systems, including at the OECD and the European Commission.

The MANAGE-WB GAMS code uses consequently forward slashes as folder separators to support at some point in the future a version running under native MAC¹⁶. Changes to the GUI layout require no knowledge of Java. Instead, the tasks and GUI controls are defined in **gui/manage.xml** and the reports in **gui/manageTables.xml**.

Baseline generation

Overview

The overview of the baseline process is depicted below. The user operates the controls on the GUI and once all settings are made, starts GAMS. Before the GUI executes the GAMS code for a baseline generation in **run.gms**, it generates the file **fromGuiToRun.gms** which maps the current control settings on the interface as chosen by the user into GAMS statements. This file is included by **run.gms**, the main file of MANAGE-WB. The user can also start **run.gms** without the GUI, for instance from GAMS Studio, and can manually change settings in **fromGuiToRun.gms**.

¹⁶ Currently, the code uses some utilities which are available under native Windows, only.

The code in **run.gms** defines some core information (for instance, directories to use, simulation horizon) in **opt.inc** and reads in the necessary data for benchmarking. This is based on two steps. First, the information from the selected base-bridge file in the country directory (**reg**) is converted into GAMS symbols in a GDX container.

Second, the sets and parameters in this GDX container are inputted into the GAMS process. These sets and parameters are mostly declared in **base.inc**. **base.inc** also comprises a longer list of data transformation statement which read in GTAP data on economic transactions and related emissions and maps them into emissions factors matching the structure of MANAGE-WB and the specific lists of products and activities in use. Country specific data input and transformation after reading and processing the default information from the base-bridge file can be done in a country specific file called **load_base.inc** which is automatically read if it exists. Users should refrain from changes to **base.inc** or **inical.inc** to the extent possible to keep the master files clean. Instead, move country specific code into **load_base.inc**.

When working on **load_base.inc** or shock files, users should avoid fixing all potential instances of a variable to zero (such as *someVariable.fx(a,j,v,t) = 0;*). Instead, the code should use available flags sets such as *flagXFac* to restrict such statements to the variable instances actually entering the model equations. Otherwise, GAMS will need to allocate memory for all potential records and all fields of a variable (.lo,.up,.l,.scale,.m). A previous version of the model comprised statements which fixed over a million records of which only a few ten thousand could actually enter some model equations. The more memory in use, the slower GAMS will get. Allocating a lot of memory might also mean that MANAGE-WB might fail to run on machines in training. It will also cost disk space when the results are saved.

The data read is then used to benchmark the model in **inical.inc**. The declarations of the variables are found in **model.inc** along with declarations of parameters used in the model. The benchmarking step defines starting values for all model variables and time points. Model parameters are either copied from symbols found in the base-bridge file (mainly substitution and transformation elasticities) or derived from them and the starting values of the variables, such as share parameters in CES and CET functions.

Before the solving loop over the years starts which generates the baseline, declarations in the shock file for the baseline (the BAU for Business-as-usual scenario) are included. Declarations inside a loop are not allowed. Any shock file must be structured as seen below, even if the declaration part is empty. Otherwise, the shock will be applied before the simulation loop with unpredictable results:

```
$ifthen.decl "%1"=="decl"
```

```
* --- here all declarations which are specific to the shock file:
```

```
$else.decl
```

```
* Here come the statements which are executed in each simulation year, declarations are not allowed
```

```
$endif.decl
```

For each year, the code in **resLastYear.inc** will first copy last year's variable levels to the current year to solve. Quantity and volume variables are additionally multiplied with next year's exogenous GDP growth. Some variables are then updated based on others (such as demography, debt stocks) and the

closures applied - i.e., some variable levels are fixed, such as real GDP per capita, and other ones which are exogenous in simulation runs are turned endogenous.

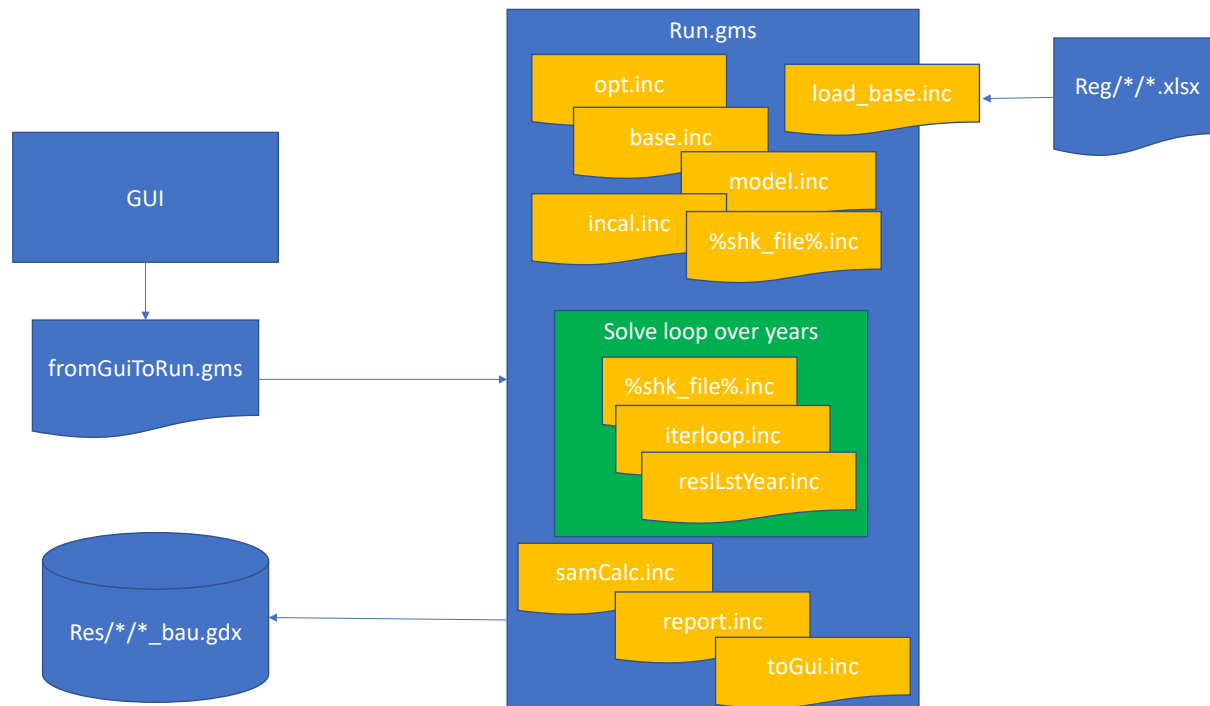
After setting the start values, the shock file is then included which introduces BAU specific changes, for instance, with regard to taxes or productivity shifters. The shock file can also define additional closure swaps or change the ones found in **iterloop.inc**. Afterwards, **iterloop.inc** is included again and fixes some tax rates and productivity shifters which do not depend on endogenous variables such that the related equations are not needed. This reduces the time for model generation and saves memory during model solving.

Should the solver not be able to solve the model in one go (too many iterations, too much time needed, declared infeasible), additional solving attempts are made. In MCP mode, different option files will be tested in repeated solve attempts. In CNS mode, subsequent solves will formally solve the model as a DNLP, using an empty objective function. This is useful as CONOPT4 applies a different solution strategy to square systems of (non)linear equations (CNS) and optimization problems (DNLP). The CNS mode is generally faster but might fail on certain model instances which can be solved as (pseudo) non-linear optimization problem. The tests in DNLP mode will introduce some bounds to avoid that the solver CONOPT4 spends a lot of time in finding defined equations and to stabilize the solution process.

Once all years are solved, the post-model reporting codes are executed according to the options chosen by the user. In order to use the reporting back-end of the GUI, **toGui.inc** must be called. It maps variables and parameters and transactions derived thereof into one multi-dimensional parameter. Additionally, the user can generate a SAM for each year and produce output for the EXCEL based front-end.

In baseline mode, all symbols (sets, parameters, variables) are stored in a GDX container under the **res** folder in a country specific sub-folder. The name of the GDX container is by default `countryName + "_" + "BAU"`. In order to support multiple baselines, the user can add a simulation name *postfix* on the GUI as shown in Figure 4.2. This container is then loaded as a start point for counterfactuals.

Figure 4.1: Flowchart of baseline generation



GUI controls in baseline mode

General steering

The GUI controls in “General settings” are shown below. The user will select the sub-directory for the country (“Region dir”, found in **reg** folder) and based on this, one of the XSLX bridge-files found there (“SAM input (xlsx)”. Equally, the last year of the simulation period is chosen, the first year is taken over from the baseline.

Figure 4.2: Screen shot “General settings” tab of the GUI in baseline mode



The second row of controls under the heading “Shock” allows to set a simulation name and to select the shock file used for the BAU run. Shock files are searched for under **/regname/sim/*.inc**. If multiple BAU runs are to be generated, the analyst can add a post-fix under “Postfix” next to “Simulation name”.

The content of the “Scen driver” fields will be written as entered by the user as additional lines in the generated include file. This content must hence comprise correct GAMS syntax. Most often, such additional lines provided by the user are used to introduce additional information, such as numerical

settings or the choice of a specific scenario into a shock file. For instance, the size of a productivity shock could be stored on a global via one of these fields:

```
Scen driver 1 $setglobal prodShockServ 1.01
```

And then used in the code of a shock file, for instance:

```
chixFacAgr.fx("serv","f-top",tt) = %prodShockServ%;
```

An appropriate use of such globals can help to keep the number of shocks files in a project small, especially to avoid repeating the same code block in many shock files. The user should however note that result files are by default named after the shock file such that either a post-fix must be set manually for results based on the same shock file, but using different settings, or the code in the shock file must define the postfix global based on the globals inputted by the user via the GUI.

The second last row of controls sets options such as “Air pollution damages” which refer to specific mechanism in MANAGE-WB and are detailed in the previous chapter. The last row comprises the option to choose the GDP composition and growth scenario from the “GDPComposition” sheet in the bridge-file and to target certain shares.

Options specific to the baseline mode are found under the “Baseline” tab:

Figure 4.3: Screen shot “Baseline” tab of the GUI

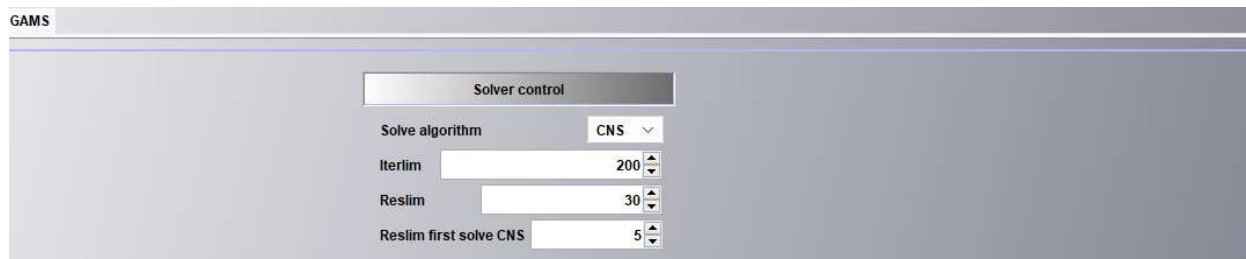


The different options available either are specific to the baseline, such as productivity changes implemented to recover a given growth path of real GDP found under “Productivity” tab or refer to model extensions which must be identically implemented in both the baseline and subsequent counterfactuals. The meaning of the different options to configure of the model is detailed in the methodological part of the documentation.

The option to switch on/off certain model mechanisms such as the *old* and *new* vintages can also help in cases when the model produces implausible results or gets infeasible to check if problems root in these optional model features.

Solver control

The last tab comprises options to change the solution algorithm



The available solve algorithms are shown below.



MCP (Mixed Complementary Programming) allows the solver to change equations defined as equality constraints into inequalities if the variable paired with the equation hits its lower or upper bound. For instance, if the emission tax under an emission ceiling becomes zero, the ceiling can become non-binding. Such options are currently hardly used in MANAGE-WB. MCP requires specialized solver such as PATH which support the “redefinitions” of equality to inequalities. As not all users of MANAGE-WB might have a PATH license, the GUI allows to use the NLPEC solver instead for MCP which is comprised for free in any GAMS license. It will reformulate the MCP as a CNS and delegate the solution of the CNS to CONOPT4. This is about as fast as PATH itself. Solving the model as a MCP is usually slower. Numerical issues are also more frequent under MCP, but there are also cases where the model can be solved in MCP but CNS mode.



The MANAGE-WB model is now defined such that it can be solved as a CNS (= square system of non-linear equations) as the default. This allows GAMS to use CONOPT4 which is usually somewhat faster than PATH.

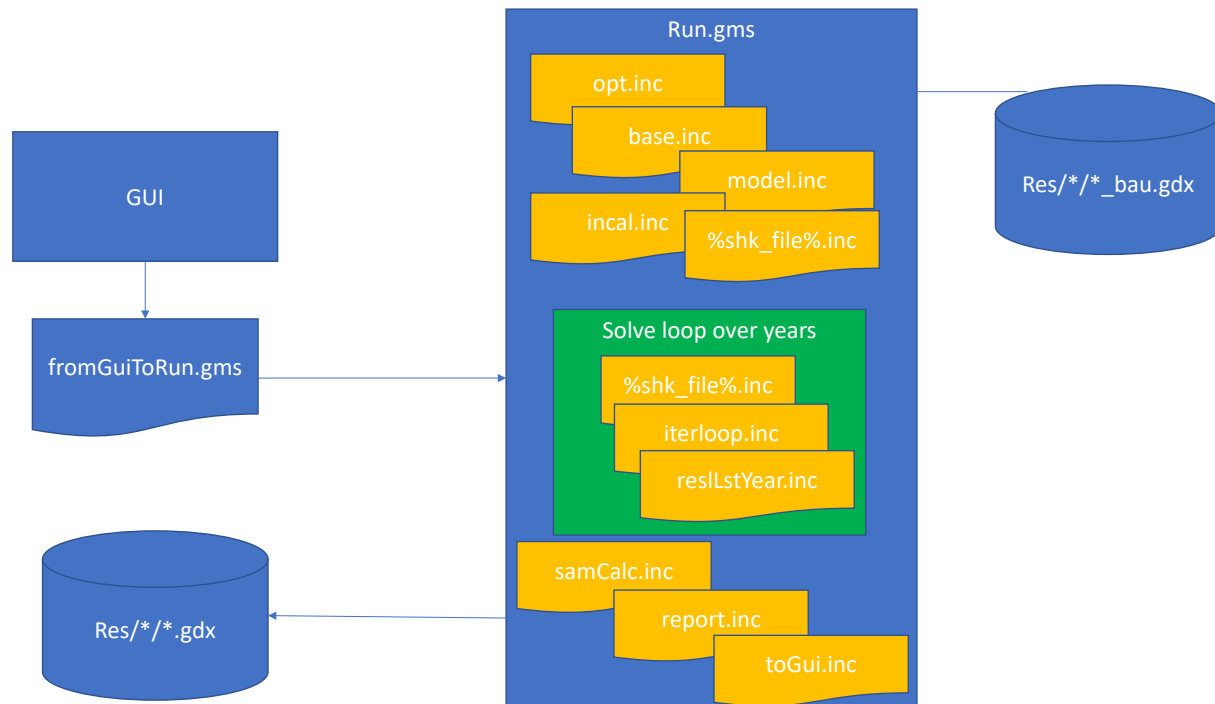
The “*Iterlim*” and “*reslim*” options define the maximum iterations and seconds allowed by the solver for one single model instance. If these limits are exceeded, the model will be declared infeasible. As discussed above, MANAGE-WB can make repeated attempts to solve the model in this case. The number of repeated attempts is defined by the “Maximum of repeated solves”. In MCP mode with PATH, repeated attempts will use different option files.

Simulations

Overview on simulation

The steps in a simulation are largely identical to what was described above for the baseline. The main difference is that the data import from the base-bridge XSLX file and the subsequent benchmarking

steps are skipped. Instead, **inc/base.inc** reads the necessary sets from the GDX container produced by a previous baseline run and **inc/cal/inical.inc** reads all parameters and start levels of variables from this GDX container as well. Instead of constructing a start point from last year's solution, **inc/solve/iterloop.inc** can use in simulations on demand the solution from the baseline for each year to solve. This can be faster, especially if the shock does not lead to larger deviations from the baseline solution.



GUI controls for simulation

The GUI options for the simulation are mostly identical to the baseline. The first and last year are loaded internally from the baseline GDX chosen by the user as they cannot be changed. If the simulation name is not left empty, it defines the name of the GDX with the simulation results. Otherwise, the simulation output is named after the chosen shock file. Entering a simulation name can also be used to pick a specific column from pre-defined shock parameters in the shock file.

Figure 4.5: GUI controls for simulation

As seen from the screen shot, five extensions can be added on top of those fixed during baseline generation: the damage modules, the fiscal responsibility mechanism, the INVEST link and short run shocks. Short run shocks do not change the equation set-up but allow the analyst to solve the model twice in each year and require a specific set-up of the shock file. During the first solve, the normal model set-up is used and solely *expected* exogenous changes by the agents should be introduced, such as changes in average temperature and precipitation. Production decisions will be based on these expected changes, and using the normal model logic, resulting adjustments in prices etc. are perfectly foresighted by all agents. The second solve will add unexpected changes, such as anormal high or low realizations of climate variables. Producers will have limited options to react to those and the second solve can therefore fix part of the factor allocation.

All other modular extensions beyond the five available in simulation mode imply changes in the model structure impacting the benchmarking step and hence require the construction of a new baseline before a counterfactual can be run where these extensions are active.

Extended options on user interface

Certain settings, especially technical one relating to the solution process, are set for a so-called “runner” in the background to reasonable defaults which work best in most cases. This avoids information overload. Model developers have additional options available on the interface. For baseline generation, the interface looks as follows and adds the Invest link which is currently under development:

The baseline tab opens up additional extensions:

Baseline Options for damages CCDR shock Invest Link **Output** GAMS

Data and checks	Model configuration	Productivity
Balanced growth check ☐	Forward looking ☐	Differentiated labor productivity ☒
Benchmark check ☐	Labor vintages ☐	Land productivity follows labor 1
Skip GTAP emission calculations ☐	Sticky wage 0.00	Natural resource productivity follows labor 1
	Vintages ☒	Capital productivity follows labor 1
	Update share parameters ☒	
	Endogenous technical progress ☒	
	Current Account Fixed as nominal GDP share	
	Transfers from ROW Driven by population growth	
	Sector specific investments ☐	
	Income dependent CDE parameters ☐	
	Capital income shares depend on saving ☐	

For simulation, the interface looks as follows:

Manage General settings

Country and time	Shock	Options	GDP Composition dynamics
Region dir vnm	Simulation name	Climate change damages ☐	GDP (composition) scenario BAU
BAU file (gdx) vnm_BAU	Shock file type GAMS	Invest link ☐	Aggregate sector shares on VA ☐
Use results as baseline ☐	Shock file (inc)	Air Pollution damages ☐	
Last year of simulation 2040	Scen driver 1	Fiscal Responsibility ☐	
	Scen driver 2	Short run shocks ☐	
	Scen driver 3		
	Scen driver 4		
	Scen driver 5		
	Scen driver 6		

Output controls

The Output tab is shown to developers, only, on the interface and allows to (de)select the reporting options. Especially for large-scale sensitivity analysis, reducing the output generated by the reporting back-end (de-selected below) and increasing the report frequency, such that results not for every year are generated, can save considerable disk-space and speed up processing. The additional output file can be provided by the user to define, for instance, additional indicators to be stored to disk.

Figure 4.4: Reporting options

Output

Output for reporting back-end	☒
Reduced output for back-end	☐
Report frequency for back-end	1
Generate SAM	☐
Output for EXCEL I/O similar to frEnd	☐
Additional output file	☒
Additional output file (inc)	output_keyind

Solver control and GAMS specific settings

The last tab comprises options for listing and profiling, the solver control and for solve outputs with extended options:

Figure 4.5: Options which affect output and the solution process

The screenshot shows the GAMS Options dialog box with four main tabs: Options for listing, Initialization, Solver control, and Options for solve outputs. The Options for listing tab is active, showing settings for Print gams code to listing (off), Symbol list (off), Symbol cross references (off), Profile (0), and ProfileTol (0.1). The Initialization tab shows Default initialization (unchecked) and Initialize factor demands (unchecked). The Solver control tab shows Solve algorithm (MCP), MCP Solver (PATH), Paired (checked), Scale model (checked), Automated scaling of SAM (checked), Base for auto-scale TE+ (5), Optfile (2), Iterlim (200), Reslim (30), and Maximum of repeated solves (6). The Options for solve outputs tab shows Solprint (On), Limrow (0.0), and Limcol (0.0).

During baseline generation, the solve for the next year will without the initialization options use the level of the endogenous variables from the last solve as the starting point, with volumes and values being updated according to the targeted GDP growth. For simulations, alternatively, the results from the baseline can be used. Activating “Default initialization” will try to improve on this by, for instance, using the demand system to capture income effects on household demands. It can be combined with initialization of the factor demands. In most cases, the default procedure works fine, and the initialization options can be skipped.

The first screenshot shows the Initialization tab with Start point (Baseline), Default initialization (unchecked), and Initialize factor demands (unchecked). The second screenshot shows the same tab with a dropdown menu for Start point open, showing options: Baseline, Baseline, and Last year's result. The third screenshot shows the Solver control tab with Solve algorithm (CNS), Scale model (checked), Automated scaling of SAM (checked), Iterlim (200), Reslim (30), Reslim first solve CNS (5), and Maximum of repeated solves (6).

Under MCP, GAMS is able to check if the pairing between variables and equations is set-up correctly and will report unmatched variables. This is helpful for debugging as adding new equations to the model structure or closure swaps in shock files can introduce errors which lead to non-square models. In such cases, the number of equations is not equal to the number of non-fixed variables. A fully paired

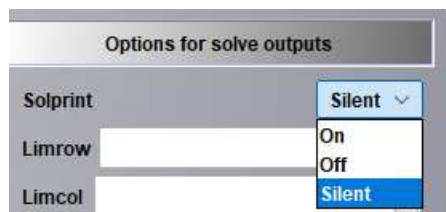
version of the model is available which eases finding variables not linked to equations or equations not linked to variables. This option is available under MCP, only, as the pairing information is neglected by GAMS if the model is not solved as an MCP. However, the default pairing defined in **inc/model.inc** might not work if shock files introduce closure swaps. Even a model with the same number of equations and non-fixed variables can provoke errors and follow-up program aborts if the new closure rules violate pairing rules. In these cases, switching the pairing off is necessary.

The most common case of pairing errors when using shock files is that GAMS will automatically remove equations which are paired to fixed variables. This behavior is consistent with the MCP rules: an equation is slack if the variable is at its lower or upper bound. If the variable is fixed, its lower and upper bound are equal. With both bounds being equal, the equation is slack by definition and cannot provide information on the model's solution space. If no redefinitions are needed, pairing can be switched off and the MCP solver used to simply neglect such problems.



The MANAGE-WB model is now defined such that it can be solved as a CNS (= square system of non-linear equations) as the default. This allows GAMS to use CONOPT4 which is usually somewhat faster than PATH.

The *"Iterlim"* and *"reslim"* options define the maximum iterations and seconds allowed by the solver to solve a single model instance. If these limits are exceeded, the model will be declared infeasible. As discussed above, MANAGE-WB can make repeated attempts to solve the model in this case. The number of repeated attempts is defined by the *"Maximum of repeated solves"*. In MCP mode with PATH, repeated attempts will use different option files.



The last block of controls under *"Options for solve outputs"* steers the amount of information which GAMS puts in the listing for each model solve. If *"Silent"* is chosen for *"Solprint"*, no output from model solves is produced and the listing has its minimum size. This also saves computing time.

"Off" will report a summary for each solved model instance (number of equations and variables, time to solve etc.) whereas *"On"* reports all equations and variables. If the model cannot be solved after repeated trials in a year, the last trial will always switch the generation of the full model listing on such that infeasible equations can be found. Generally, it is often more convenient to use GAMS Studio to inspect the variables from a model solution instead of using the equation listing. Setting *"Solprint"* to *"Silent"* or *"Off"* is therefore the recommended option.

Both with *"solprint"* set to *"off"* or *"on"*, *"limrow"* and *"limcol"* can be used to produce for the first n instances, as inputted in the text field, of each equation (*limrow*) or variable (*limcol*) a report with the record fields (level, lower and upper bound, marginal) and the Jacobian entries. Note that in case of non-linear expressions, the Jacobian entries are not constants and calculated at the given start values of all variables in an instance of an equation. Their value will hence change during the solution process. Switching *"limcol"* on by setting a non-zero number can be useful in case of unmatched variables to find out in which equations they occur. As the size of model listing increases substantially with *"limrow"* or *"limcol"* switched on, the two fields are by default set to zero.

CCDR simulation driver

Background

The work on a Climate Change Develop Report (CCDR) or other projects focusing on Climate Change (CC) related work with MANAGE-WB encompasses some typical shocks related to mitigation of CC emissions, introduction of CC-related damages and adaptation measures to reduce damages. To help an analyst to introduce such standard shocks, the CCDR simulation driver was developed and consists of the following components:

1. There is one central shock file called *"ccdr"* which combines different *"sub-shocks"*, such as fixing the power mix, fixing GHG emissions to define an endogenous carbon tax, implementing carbon tax recycling, integrating information of a transport model, introducing CC damages based on IEC input, introducing adaptation etc.
2. The different elements, such as fixing the power mix, can be activated by check boxes on the GUI.
3. Each sub-shock has its own separate shock file. This avoids one very long and complex shock file where each change to a single sub-shock would trigger commits.
4. The CCDR central shock files and the sub-shock files are stored as templates in the sub-folder *"inc/shockfiles"* from which they are automatically copied to the regional directory if SAMEST is run.
5. The project will use these copies in its regional folder which allows to introduce project specific adjustments without changes to the central templates, supporting a clean master version under version control.

Putting to a large extent choices on the interface eases structural sensitivity analysis in batch mode, such as assessing the macro-impact of specific damages; templates for such batch analysis can be standardized as well.

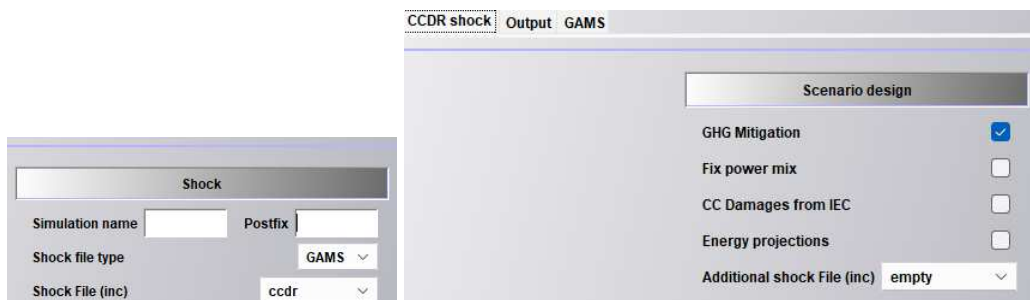
Figure 2: Current content of template inc/shockFiles subdirectory

 BalGrowth.inc	27.10.2023 14:33	INC-Datei	4 KB
 ccdmgIEC.inc	20.10.2023 10:42	INC-Datei	6 KB
 ccdr.inc	28.10.2023 13:27	INC-Datei	2 KB
 empty.inc	20.10.2023 12:56	INC-Datei	1 KB
 fixEnergy.inc	28.10.2023 13:16	INC-Datei	4 KB
 fixPowerMix.inc	20.10.2023 10:29	INC-Datei	2 KB
 GHGMitigation.inc	28.10.2023 13:16	INC-Datei	6 KB
 none.inc	20.10.2023 06:33	INC-Datei	1 KB
 numtest.inc	13.10.2023 10:07	INC-Datei	1 KB
 redGhg.inc	20.10.2023 06:33	INC-Datei	2 KB

The files *ccdr.inc*, *ccdmgIEC.inc*, *fixEnergy.inc*, *fixPowerMix.inc* and *GHGMitigation.inc* shown above are part of the CC DR templates.

As seen from the screen shot below (Figure 3: Screen shot of CC DR tab on the interface), choosing the CC DR shock file automatically adds a tab to the interface on which the different sub-shocks can be switched on/off by the user. Specific settings for the sub-shocks are handled there as well (such as, for instance, options for tax-recycling). These specific settings are discussed below.

Figure 3: Screen shot of CC DR tab on the interface



The CC DR shock file includes the different sub-shocks (see Figure 4: Declaration part of the *ccdr.inc*, the top-level shock file template), depending on the choice on the interface, and defines a variant of the model with some changes in the pairing to introduce required closure swaps and additional equations related to tax recycling.

Figure 4: Declaration part of the *ccdr.inc*, the top-level shock file template

```

$iftheni.decl "%1"=="decl"

  $$iftheni.baseline "%task%"=="Baseline"
  *
  *   --- in baseline mode, make sure that all symbols are loaded for subsequent
  *   use in counterfactuals
  *
  parameter GHGMitigation(*,*,t) "Relative emission ceilings and carbon taxes"
            powerMix(*,*,t) "Power mix and power price projections from basebridge"
            energyProj(*,*,*,t) "Projections of changes in demand for energy services and energy use";
  set aEner(a),shockedFpNest(fpNest),shockedI(i);
  $$gdxin %readFile%
    $$load    aEner<energyProj.dim2
    $$load    shockedFpNest<energyProj.dim3
    $$load    shockedI<energyProj.dim2
  $$gdxin

  $$endif.baseline
  *
  *   --- add declaration part (including definitions of additional variables and equations)
  *   for subshocks
  *
  $$onmulti
  $$ifi "%GHGMitigation%"=="on" $batinclude '%simf%/GHGMitigation.inc' decl

  $$ifi "%fixPowerMix%"=="on" $batinclude '%simf%/fixPowerMix.inc' decl

  $$ifi "%CCEnergy%"=="on" $batinclude '%simf%/fixEnergy.inc' decl
  $$offmulti

  $$ifi "%ccDamagesIEC%"=="on" $batinclude '%simf%/ccDmgIEC.inc' decl

  $$batinclude '%simf%/add_shk_file%.inc' decl

  $$setglobal modelVariant "ccdr"

  Model cge_%modelVariant%
  /
  cge_

  *
  *   --- add additional model defined by the user (redirections are similar)
  *
  $$ifi not set addModel $set addModel SomeUnknownModel
  $$ifi defined %addModel% %addModel%

  *
  *   --- remove the pairing for the power mix and the fixing of the electricity price
  *
  $$ifi "%fixPowerMix%"=="on" -e_elecMix -e_ps +e_elecMix +e_ps

  $$ifi "%CCEnergy%"=="on" -e_xat +e_xat -e_xap +e_xap -e_xFac +e_xFac e_xFacAgr.chixFacAgr + e_lambda0a.lambda0a

  $$iftheni.ghgMitigation "%GHGMitigation%"=="on"
    m_ghgMitigation -e_emigHG + e_emigHG
    $$ifi "%maccs%"=="on" -e_px + e_pxMac
  $$endif.ghgMitigation

  *
  $$iftheni.baseline "%task%"=="Baseline"
  *
  *   --- implement BaU closure by changing pairing
  *
  -e_rgdmp
  $$iftheni.diffLabProd "%diffLabProd%"=="on"
    e_rgdmp.labProd
  $$else.diffLabProd
    e_rgdmp.chixFacL
  $$endif.diffLabProd

  $$endif.baseline

  /;

```

Mitigation policies

The exogenous changes superimposed on the model related to GHG mitigation policies are stored in the bridge file and need to populate a three-dimensional parameter call GHGMitigation (Scenario, policy instruments, time). The GTAP Power template uses a sheet called "GHGMitigation" to do so:

Figure 5: Entering GHG mitigated related policy instruments in the bridge file

		2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
HAU	carbonTax		1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
HAU	ceilingBench	1	0.98	0.96	0.94	0.92	0.9	0.88	0.86	0.84	0.82	0.8	0.78
HAU	ceilingBaseline	1	0.98	0.96	0.94	0.92	0.9	0.88	0.86	0.84	0.82	0.8	0.78
HAU	ceilingAbsolute												

The emission ceiling is defined here in relative terms, in baseline mode solely the option where a multiplier relative to benchmark year $t0$ is employed can be used. In counterfactuals, the user can define the multiplier relative to a given baseline scenario or the benchmark point. The code also supports imposing an absolute ceiling.

The policy instrument (taxes or ceiling) and the definition of how the ceiling is defined can be chosen on the interface. Equally, the analyst can decide if carbon stock changes from land use should be considered in the calculation of emissions or not.

Figure 6: Interface options related to mitigation instruments

GHG mitigation

GHG mitigation scenario: BAU

Policy instrument: Carbon tax

MACCs ☒

Direct tax recycling: 0.10

Factor tax recycling: 0.10

Sales tax recycling: 0.10

All tax recycling: 0.10

Transfers recycling: 0.10

If carbon taxing is used, the code of the sub-shock file *GHGMitigation.inc* will fix the carbon tax to the values as entered in EXCEL in absolute levels. Otherwise, if a GHG ceiling is used as the policy instruments, the sum of the GHG emissions will be fixed and the emission tax becomes endogenous:

```

---- Simulation mode, code is called for each year

$$iftheni.polInstrument "%Policy_instrument%"=="carbon tax"

    --- exogenous carbon tax

    tEmiD0.fx(%tt%) = GHGMitigation("%GHGMitigationScenario%", "carbonTax", %tt%);
    hasGhgPolicy $ sum(t, GHGMitigation("%GHGMitigationScenario%", "carbonTax", t)) = 1;

$$else.polInstrument

    $$iftheni.baseline "%task%_Policy_instrument%"=="Baseline_Ceiling to baseline"

        $$abort "Ceiling to baseline cannot be used in baseline mode."

    $$endif.baseline

    --- emission ceiling (= emissions fixed), carbon tax is endogenous

    $$iftheni.ceiling "%Policy_instrument%"=="Ceiling to baseline"
        emiGHG.fx("GHGs", %tt%) $ GHGMitigation("%GHGMitigationScenario%", "ceilingBaseline", %tt%)
        = p_ghgs(%tt%) * GHGMitigation("%GHGMitigationScenario%", "ceilingBaseline", %tt%);
        hasGhgPolicy $ sum(t, GHGMitigation("%GHGMitigationScenario%", "ceilingBaseline", t)) = 1;
    $$elseifi.ceiling "%Policy_instrument%"=="Ceiling to benchmark"
        emiGHG.fx("GHGs", %tt%) $ GHGMitigation("%GHGMitigationScenario%", "ceilingBench", %tt%)
        = p_ghgs(t0) * GHGMitigation("%GHGMitigationScenario%", "ceilingBench", %tt%);
        hasGhgPolicy $ sum(t, GHGMitigation("%GHGMitigationScenario%", "ceilingBench", t)) = 1;
    $$else.ceiling
        emiGHG.fx("GHGs", %tt%) $ GHGMitigation("%GHGMitigationScenario%", "ceilingAbsolute", %tt%)
        = GHGMitigation("%GHGMitigationScenario%", "ceilingAbsolute", %tt%);
        hasGhgPolicy $ sum(t, GHGMitigation("%GHGMitigationScenario%", "ceilingAbsolute", t)) = 1;
    $$endif.ceiling

    if ( emiGHG.range("GHGs", tt) eq 0,
        tEmiD0.lo(ts(t)) = -inf;
        tEmiD0.up(ts(t)) = +inf;
        tEmiD0.l(%tt%) = tEmiD0.l(%tt1%);
        tEmiD0.l(ts(t)) $ (Not tEmiD0.l(t-1)) = 0.01;
    );
$$endif.polInstrument

```

The code also supports options to recycle share of the total emission tax revenues to lower other taxes.

If a counterfactual is run and **tax recycling of direct taxes** is used, the code sets the (nominal) sum of direct (*cit*, *pit*) tax revenues to the given direct tax revenue of the baseline (stored on *p_taxes*) plus a share of the carbon tax revenues (*emi*, *emiXp*). The global %dirTaxRec1% captures the share as entered on the interface:

```

reyclDirectTaxesEq(ts(t)) $ (tDr.range(t) ne 0) ..

    (yg("cit", t) + yg("pit", t))/max(1, (yg.l("cit", t-1) + yg.l("pit", t-1))*%scaleMult%)

    =E=
----- 8 line(s) not displayed -----
    --- counterfactual: increase direct tax revenues from baseline by carbon tax revenue

    p_taxes("pit", t)+p_taxes("cit", t) - (yg("emi", t)+ yg("emiXp", t))*%dirTaxRec1%
----- 2 line(s) not displayed -----
    ]/max(1, (yg.l("cit", t-1) + yg.l("pit", t-1))*%scaleMult%);

```

If tax recycling is used in the baseline, the meaning is not so well defined. The code currently let the targeted tax flows (direct taxes or all) increase with nominal GDP:

```

reyclDirectTaxesEq(ts(t)) $ (tDr.range(t) ne 0) ..
    (yg("cit",t) + yg("pit",t))/max(1, (yg.l("cit",t-1) + yg.l("pit",t-1))*%scaleMult%)
    =E=
[
----- 1 line(s) not displayed -----
*
*   --- baseline: let taxes increase with GDP at market prices plus carbon tax revenue
*
    ((yg.l("cit",t-1) + yg.l("pit",t-1))*gdpmp(t)/gdpmp.l(t-1)
    + (yg("emi",t) + yg("emiXp",t))-(yg.l("emi",t-1) + yg.l("emiXp",t-1)))*%dirTaxRec1%
----- 7 line(s) not displayed -----
]/max(1, (yg.l("cit",t-1) + yg.l("pit",t-1))*%scaleMult%);

```

Similar expressions are used to update taxes rates for factors, sales or all taxes. The template allows also to recycle the tax flow to update transfers, defining the variable *yTaxRedist* as part of the model template.

If the shares for carbon tax recycling are in sum lower than unity, the overall tax income is likely to increase which will reduce the borrowing needs of the government in case of a government deficit. If the government runs a surplus, the additional tax income will increase overall savings instead. In both cases, capital accumulation is likely to increase, fostering economic growth. Due to these mechanisms, it is not clear ex-ante if direct tax-recycling will make households better off in the long run. With tax recycling, there is an immediate utility gain of reducing direct taxes for households by higher spendings on consumption. Additionally, part of the additional household income (and enterprise income, if it is subject to direct taxation) will increase savings which implies higher investments and thus increased capital accumulation.

There is an also subtle accompanying effect on the interest rate on government debt in case of a government deficit. If tax recycling rate add up to unity, they should not directly affect the borrowing needs of the government (tax income should be unchanged). But tax reduction are likely to increase household income. At unchanged saving rates, this will increase household savings which in turns implies that the share of government bonds bought by households in their saving portfolio must drop at unchanged government borrowing. This requires that the interest on government bonds falls relative to expected returns to capital. A lower interest will imply lower future interest payments and thus lower future borrowing needs and debt stocks.

Without tax recycling, the additional tax revenue will not trigger additional household consumption (not immediate utility gain) and instead fully reduce borrowing needs (or increase government savings). The dynamic effects of increased capital accumulation or lower debt stocks (also due to lower interest rates) are hence likely higher compared to tax recycling.

Given these observations, it is recommended to use sensitivity analysis with regard to tax recycling. The possibility to allocate shares of the emission tax revenues to different taxes allows to finetune the related scenarios.

Power mix and electricity price fixing

If the fix power mix option is activated on the CCDD interface tab, the user can define the name of the power mix scenario in a text field:

Figure 7: Options related to power mix and power price fixing

Power mix and power price projections for different scenarios are inputted in the bridge-file. The GTAP-Power template outputs a constant benchmark mix and zero price changes in a sheet called "powermix" for testing to populate a three-dimensional parameter powerMix (scenario, power generation sector, year):

Figure 8: Power mix and power price information in the bridge-file

	A	B	C	D	E	F	G	H	I	J	K	L
1			2018	2019	2020	2021	2022	2023	2024	2025	2026	2027
2	bas	a-CoalBL	0.40066655	0.40066655	0.40066655	0.40066655	0.40066655	0.40066655	0.40066655	0.40066655	0.40066655	0.40066655
3	bas	a-WindBL	0.2630471	0.2630471	0.2630471	0.2630471	0.2630471	0.2630471	0.2630471	0.2630471	0.2630471	0.2630471
4	bas	a-OtherBL	0.23287681	0.23287681	0.23287681	0.23287681	0.23287681	0.23287681	0.23287681	0.23287681	0.23287681	0.23287681
5	bas	a-GasP	0.06603905	0.06603905	0.06603905	0.06603905	0.06603905	0.06603905	0.06603905	0.06603905	0.06603905	0.06603905
6	bas	a-HydroP	0.00074244	0.00074244	0.00074244	0.00074244	0.00074244	0.00074244	0.00074244	0.00074244	0.00074244	0.00074244
7	bas	a-OilP	0.00160889	0.00160889	0.00160889	0.00160889	0.00160889	0.00160889	0.00160889	0.00160889	0.00160889	0.00160889
8	bas	a-SolarP	0.03501915	0.03501915	0.03501915	0.03501915	0.03501915	0.03501915	0.03501915	0.03501915	0.03501915	0.03501915
9	bas	elecPrice	1	1	1	1	1	1	1	1	1	1

This data is used by the power-mix shock file and loaded in "base.inc" during baseline generation if found. The code raises an error if the option to fix the power mix is activated the data are not available and searches the power activity with the biggest share at the benchmark:

```

$iftheni.decl "%1"=="decl"

$ifnot defined PowerMix $abort "Bridge-file missing powerMix projections, in file: %system.fn%, line: %system.incline%"

*
* --- largest electricity activity in t0 (higher share on powermix)
*
set aElecFst(a);
alias(aElec,aElec1);
aElecFst(aElec) $ (powerMix("%powerScen%",aElec,t0) eq smax(aElec1, powerMix("%powerScen%",aElec1,t0))) = YES;
set elecPrice / elecPrice /;

```

During simulation, the share of each power activity in total power output, as defined by the variable *elecmmix* in *model.inc*, is fixed for n-1 power activities (excluding the largest one). To keep the model square, total productivity shifters are released:

```

*
* --- fix n-1 power mix shares
*
elecmmix.fx(aElec,iElec,tt) $ (not aElecMax(aElec)) = powerMix("%powerScen%",aElec,tt);

*
* --- release prod shifters for n-1 power activities
*
chixFacFa.lo("f-top",aElec,tt) $ (powerMix("%powerScen%",aElec,tt) and (not aElecMax(aElec))) = -inf;
chixFacFa.up("f-top",aElec,tt) $ (powerMix("%powerScen%",aElec,tt) and (not aElecMax(aElec))) = +inf;

```

Fixing solely n-1 shares avoids potential numerical problems with adding up.

If also a projection for the electricity price is given, this price becomes exogenous and the total productivity shifter for the largest power producing activity is additionally released:

```

*
* --- fix electricity price (implicitly defines the productivity shift for the n-1 share)
*
$if(Lhs1,fixElecPrice,"$fixElecPrice$"--"on"
  ps.fx(iRinc,LL) $ powerMix("$powerScen$","elecPrice",LL) = ps.f(iRinc,LL) * powerMix("$powerScen$","elecPrice",LL);
  chxFacTa.to("f-lop",aRincPst,LL) $ powerMix("$powerScen$","elecPrice",LL) = -inf;
  chxFacTa.up("f-lop",aRincPst,LL) $ powerMix("$powerScen$","elecPrice",LL) = +inf;
$endif,fixElecPrice

```

As using MCP will automatically remove equations paired to fixed variables, the ccdr template uses the following at first glance curious statement:

```

*
  $if1 "%fixPowerMix$=="on" -elecMixEq -pseq +elecMixEq +psEq

```

The effect of this is that the original pairing of the two equations *elecMixEq* and *psEq* defined in *model.inc* is removed, and the two equations are not paired any longer to a variable. This keeps the model square in case it is solved as an MCP. This is called a re-direction in GAMS syntax.

IEC damages

Converting the IEC CSV files to a GDX container

The shock file "*ccDmgIEC.gms*" assumes that a matrix of CC damage shocks is found in a GDX container called "*ddmg_shocks_IEC.gdx*". Such a container can be produced with (a variant of) the GAMS main "*readIEC.gms*" which is found in the *inc* directory. For this main program to work, the csv containers for different damage channels and potentially also scenarios must be found in sub-directories.

The following code snippet shows how CSV-files from IEC are first converted to intermediate GDX containers (these statements are commented out in the shown file version as the conversion step only needs to be done once and is time consuming). The data from the CSV are then loaded into GAMS symbols with GDXIN at compile time to make the labels available to the GAMS code. At run time, the damage matrix for the specific channel and scenario is then read into one damage matrix *p_shocks*.

```

*execute_load 'csv2GDX' "inland_flooding/RTM_inland_flooding_capital_loss.csv" in=p_flood_capital index=1,2 values=3 useheader=y'
*execute_load 'csv2GDX' "inland_flooding/RTM_inland_flooding_capital_loss/adapt.csv" in=p_flood_capital index=1,2 values=3 useheader=y'

$onmulti
sol years,SSP;
$gdxin RTM_inland_flooding_capital_loss
  $load years,p_flood_capital.sim1
  $load SSP,p_flood_capital.sim2
$gdxin
$offmulti

parameter p_flood_capital(*,*)

execute_load "RTM_inland_flooding_capital_loss/adapt" p_flood_capital;
p_shocks(years,ssp,"","inland_flood","adapt") = p_flood_capital(years,SSP);

execute_load "RTM_inland_flooding_capital_loss" p_flood_capital;
p_shocks(years,ssp,"","inland_flood","no_adapt") = p_flood_capital(years,SSP);

```

Once all CSV containers are processed, the damage matrix is stored in a GDX container for use with MANAGE-WB:

```

* --- output summary results

execute_unload "shocks" p_shocks;

```

Note that our colleagues from IEC have the tendency to rename CSV files, shocks etc. from release to release. It is therefore important to check carefully the content of the generated GDX container for completeness by comparing it, for instance, with GAMS Studio, against the data found in the CSV files.

IEC damages as part of the CCDR shock template

The interface options for the IEC damages are more evolved compared to power fixing. They allow us to define the IEC scenario, climate, climate type and selected one, several or all damage channels.

Figure 9: Interface options related to IEC damages

The screenshot shows the CCDR shock interface with two main panels: 'Scenario design' and 'IEC Damages'. In the 'Scenario design' panel, 'Fix power mix' and 'CC Damages from IEC' are checked, and 'Power mix scenario' is set to 'bas'. In the 'IEC Damages' panel, 'IEC scenario' is 'CG', 'IEC climate' is 'Wet/Warm mean', and 'IEC climate type' is 'Wet/Warm'. The 'Damage channels' section lists 'health', 'lab', 'livestock', 'urban_flood', 'hydro', 'road_bridge', and 'inland_flood' with checkboxes.

The damage matrix is read first by the related sub-shock file "*ccDmgIEC.inc*". This is a two-step procedure where first the labels used on the matrix are loaded at compile time into GAMS sets. This allows for domain checking on the matrix and other parts of the code. Based on this information, the matrix itself is loaded.

```
*
* --- read the labels from the IEC shock matrix from the GDX
*
set clim,clim_type,scen,chnl,ccShock;
$$gdxin '%simf%/ccdmg_shocks_IEC.gdx'
  $$load clim < p_shocks.dim2
  $$load clim_type < p_shocks.dim3
  $$load scen < p_shocks.dim4
  $$onmulti
  $$load chnl < p_shocks.dim5
  $$offmulti
  $$load ccShock < p_shocks.dim6
  parameter p_shocks(t,clim,clim_type,scen,chnl,ccShock) "IEC Shock matrix";
  $$load p_shocks
  $$gdxin
```

The damages for the different channels are assigned to the *dmg* variables of MANAGE-WB. For instance, productivity gains for labor from better sanitation ("wash") are assigned to the damage shifter variable *dmgSup*, increasing the labor stock.

```
*
* --- improved labour productivity from WASH under SR scenario
* (only for CG)
*
dmgSup.fx(1,"lab",LL) = curChnl("wash") - p_shocks(LL,"STECCLimS","STECCLimTypeS","STECscenS","health","pol_change_supply");
```

In other cases, such as for changes in livestock productivity, productivity is shocked:

```
*
* damage to livestock productivity. Only available for CG
*
dmgProd.fx("f top",anim,"livestock",tt) = curChnl("livestock")
  - p_shocks(tt,"IECCLimS","IECCLimTypeS","CG","livestock","");
```

The damages are introduced via the macro *m_dmAgg* into various model equations:

```
* --- calculates cumulative damage from different channels by including only the non-zero variables
$$macro m_dmAgg(var,dims) prod(chnl,1+var(&&dims&&,chnl,t) $ ((var.1(&&dims&&,chnl,t) ne 0) $ ifdmg))
```

This macro links damages from multiple channels *chnl* for the damage variable *var* in a multiplicative way. Damages variables are shifters in percentage terms, i.e., a 0.05 implies a shift of 5%. The impact of the damage variables is introduced in the model only if the flag *ifdmg* is non-zero.

The following code snippet shows how the climate change damage shifters are used to change (potentially together with other shifter variables) the productivity multiplier *lambdaXfac*. Shifters stored on the variable *dmgPrd* must hence be defined for specific technology nest *pNest*, activity *and* year *t*.

```
* --- Productivity factor shifter equation

lambdaXfacEq(flagXfac(fpNest,a,v),ts(t)) $ ((lambdaXfac.range(fpNest,a,v,t) ne 0) $ (lambdaXfac(fpNest,a,v,t)/max(1,lambdaXfac.1(fpNest,a,v,t-1)*%scaleMult%)
    =e= [ chixFacF(fpNest,t) * chixFacA(a,t) * chixFacV(v,t)
        * chixFacFA(fpNest,a,t)
* --- air pollution impact on labor productivity
        * (1 + dmgAir(fpnest,a,t) $ ifAirPolDmg $ 1(fpnest))
* --- shifter for all labor types
        * (1 + (chixFacL(t)) $ 1(fpnest))
* --- climate damages
        * m dmgAgg(dmgsPrd,'fpnest,a')
* --- shifter for capital
        * (1 + chixFacK(t) $ cap(fpnest))
```

A second damage variable called *dmgStk* is used to shift the capital stock and population by age. The third damage variable *dmgTrd* allows to shift the export supply for products, while the *dmgSup* variable affects the factor supply equations (capital, labor, land, natural resources). The damage variable *dmgDem* allows to shift demand for all final demanders *oa*, products *i* and demands nests *kNests*.

The damage variables are used as well by the CC damages from Roson and Sartori.

Energy projections

A sheet call "Energy" in the bridge-file populates the parameter "energyProj" with data for which an example is shown below:

1				2017	2018	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029
2	BAU	c-OIL	a-OTP	1	1.06936037	1.14289384	1.1703767	1.1937573	1.25265586	1.32961062	1.41254639	1.50155414	1.59267696	1.68492858	1.77893667	1.87577272
3	BAU	c-OIL	a-WTP	1	1.0697217	1.14378198	1.17337815	1.19922666	1.25965047	1.33783196	1.42208386	1.51251552	1.60529928	1.6994917	1.79571479	1.89503067
4	BAU	c-OIL	a-ATP	1	1.07341783	1.15168556	1.18390066	1.2123068	1.27701815	1.36064213	1.45102592	1.54833775	1.64857018	1.75073276	1.85549387	1.96400158
5	BAU	c-P C	a-OTP	1	1.06936037	1.14289384	1.1703767	1.1937573	1.25265586	1.32961062	1.41254639	1.50155414	1.59267696	1.68492858	1.77893667	1.87577272
6	BAU	erier	a-OTP	1	1.07473404	1.15451008	1.18810938	1.2179338	1.28444754	1.3702061	1.46288931	1.56289055	1.66617253	1.77153893	1.87977816	1.99206382
7	BAU	erier	a-WTP	1	1.07186543	1.18037389	1.18044667	1.2088663	1.27232291	1.35399893	1.44215327	1.53693503	1.63448576	1.73385843	1.83569873	1.94110822
8	BAU	erier	a-ATP	1	1.07419232	1.15399238	1.18746248	1.21716818	1.28342245	1.36883163	1.46122385	1.56078038	1.66348177	1.76933676	1.8760273	1.98772351
9	BAU	erier	a-oupEle	1	1.07476272	1.15457152	1.18818616	1.21802478	1.28456922	1.37036925	1.46319893	1.56325299	1.66649208	1.77192038	1.8802237	1.99257943
10	BAU	gas pldt	a-OTP	208845.403	224453.263	241114.122	248131.181	254359.895	268250.964	286161.307	305538.592	326423.39	347972.473	369977.97	392583.026	416033.37
11	BAU	gas pldt	a-WTP	380.807711	408.174621	437.309631	449.523194	460.346197	484.510374	515.613232	549.183086	585.276713	622.42478	660.266661	699.04823	739.188978
12	BAU	gas pldt	a-ATP	221.387732	237.879418	255.479757	262.889626	269.466103	284.133881	303.043195	323.497035	345.537628	368.274457	391.488064	415.329428	440.057599
13	BAU	gas pldt	a-oupEle	1	1.07476272	1.15457152	1.18818616	1.21802478	1.28456922	1.37036925	1.46319893	1.56325299	1.66649208	1.77192038	1.8802237	1.99257943
14	BAU	ely lnd	a-OTP	208845.403	224453.263	241114.122	248131.181	254359.895	268250.964	286161.307	305538.592	326423.39	347972.473	369977.97	392583.026	416033.37
15	BAU	ely lnd	a-WTP	380.807711	408.174621	437.309631	449.523194	460.346197	484.510374	515.613232	549.183086	585.276713	622.42478	660.266661	699.04823	739.188978
16	BAU	ely lnd	a-ATP	221.387732	237.879418	255.479757	262.889626	269.466103	284.133881	303.043195	323.497035	345.537628	368.274457	391.488064	415.329428	440.057599
17	BAU	ely lnd	a-oupEle	1	1.07476272	1.15457152	1.18818616	1.21802478	1.28456922	1.37036925	1.46319893	1.56325299	1.66649208	1.77192038	1.8802237	1.99257943
18	BAU	demand	a-OTP	1	1.07742761	1.16030435	1.19706494	1.22018874	1.30062426	1.39094054	1.48884958	1.59460498	1.70413435	1.81644354	1.93225594	2.0528083
19	BAU	demand	a-WTP	1	1.07455181	1.15413736	1.18934447	1.22103325	1.28834892	1.37448781	1.46764523	1.56882245	1.67172563	1.77780691	1.88694594	2.00022989
20	BAU	trpyFac	a-ISR	1	1.005	1.010025	1.01507513	1.0201505	1.02525125	1.03037751	1.0355294	1.04070704	1.04591058	1.05114013	1.05639583	1.06167781
21	BAU	trpyFac	a-OMN	1	1.005	1.010025	1.01507513	1.0201505	1.02525125	1.03037751	1.0355294	1.04070704	1.04591058	1.05114013	1.05639583	1.06167781
22	BAU	power	c-COA	67.8	67.8	67.8	67.8	147.9	344.9	231.7	200.8	181.6	162.5	143.3	124.2	105
23	BAU	power	c-OIL	47.2	47.2	47.2	47.2	75.5	99.8	88.8	75.7	77.7	79.7	81.7	83.8	85.8
24	BAU	power	c-GAS	2.2	2.2	2.2	2.2	4.1	6.1	6	5.7	5.4	5.2	5	4.7	4.5
25	BAU	power	c-P C	47.2	47.2	47.2	47.2	75.5	99.8	88.8	75.7	77.7	79.7	81.7	83.8	85.8
26	NZ	power	c-COA	67.8	67.8	67.8	67.8	147.9	344.9	231.7	200.8	177.4	154	130.5	107.1	83.7

The first column depicts the energy scenario, allowing to store multiple ones in one shock file. This data can be used with the "fixEnergy.inc" shock file, related options are set on the "Energy" panel as shown below. The energy scenario is selected there, as well as the first year where projections are

introduced. Projections can be entered in absolute terms or as multipliers. Absolute values are internally converted in the multipliers as shown below. This allows the analyst to enter external projections, for instance, on world market prices or energy demands as reported directly in the sheet without the need to re-base them on the SAM or price level derived from it.

Figure 10: Interface options related to fixing of variables in energy markets

The screenshot shows a software interface titled "Energy". It contains several settings:

- Energy scenario:** A text box containing "BAU".
- Projection start year:** A dropdown menu showing "2018".
- Total demand projections:** A checkbox that is checked (indicated by a blue checkmark).
- Activity specific projections:** An unchecked checkbox.
- Energy efficiency projections:** An unchecked checkbox.
- World market price projections:** An unchecked checkbox.

Total demand projections can be firstly entered for an activity. The related label is "demand" in the second column, examples are shown above. This fixes the total Armington demand for the product produced and swaps it with a total productivity shifter for the largest activity producing this product:

```
*
* --- fix the economy wide demand for energy services according to projection
* (here, the use of an activity indicates that the prod. of the activity producing the output should be targeted)
*
* xat.fx(1,0,0) $ (sum(shner $ sum(shner,1),energyProj("EnergyScen0","Demand",shner,0,0)) $ (years("GenerProStartYear0") 1e years(0,0)))
*   = xat.1(1,0) * sum(shner $ (sum(shner,1) $ energyProj("EnergyScen0","Demand",shner,0,0)),
*     energyProj("EnergyScen0","Demand",shner,0,0)/energyProj("EnergyScen0","Demand",shner,0,0))
*
* --- shift productivity for energy activities to recover fixed demand forecast
*
* chixacta.1("I-top",shner,0,0) $ (energyProj("EnergyScen0","Demand",shner,0,0) $ (years("GenerProStartYear0") 1e years(0,0))) = chixacta.1("I-top",shner,0,0):
* chixacta.1o("I-top",shner,0,0) $ (energyProj("EnergyScen0","Demand",shner,0,0) $ (years("GenerProStartYear0") 1e years(0,0))) = -inf:
* chixacta.1p("I-top",shner,0,0) $ (energyProj("EnergyScen0","Demand",shner,0,0) $ (years("GenerProStartYear0") 1e years(0,0))) = -inf:
*
* -- if multiple activities produce i, only fix the largest one
*
* chixacta.fx("I-top",shner,0,0) $ (no. largestProducers(shner,"")) = chixacta.1("I-top",shner,0,0):
```

Alternatively, the demand for a product can be targeted directly, in which a case uniform shifter affects all intermediate demand and final demand of the product:

```
energy demand is targeted directly (not via activity)

xat.fx(iRner,0,0) $ (energyProj("EnergyScen0","Demand",iRner,0,0) $ (years("GenerProStartYear0") 1e years(0,0)))
= xat.1(iRner,0,0) * energyProj("EnergyScen0","Demand",iRner,0,0)/energyProj("EnergyScen0","Demand",iRner,0,0)

shift productivity for intermediate demand of the product

chixFacP.1oactPpNest,0,0) $ (sum(sameas(iRner,actPpNest),energyProj("EnergyScen0","Demand",iRner,0,0)) $ (years("GenerProStartYear0") 1e years(0,0))) =
chixFacP.1oactPpNest,0,0)
chixFacP.1oactPpNest,0,0) $ (sum(sameas(iRner,actPpNest),energyProj("EnergyScen0","Demand",iRner,0,0)) $ (years("GenerProStartYear0") 1e years(0,0))) = -inf:
chixFacP.1pactPpNest,0,0) $ (sum(sameas(iRner,actPpNest),energyProj("EnergyScen0","Demand",iRner,0,0)) $ (years("GenerProStartYear0") 1e years(0,0))) = -inf:

and use the same shifter for the preferences in final demand

lambda0a.1o(i,0,0,0,0,0) $ (energyProj("EnergyScen0","Demand",i,0,0) $ p_0f(i,0,0,0,0,0))
= lambda0a.1o(i,0,0,0,0,0)
lambda0a.1o(i,0,0,0,0,0) $ (energyProj("EnergyScen0","Demand",i,0,0) $ p_0f(i,0,0,0,0,0)) = -inf:
lambda0a.1p(i,0,0,0,0,0) $ (energyProj("EnergyScen0","Demand",i,0,0) $ p_0f(i,0,0,0,0,0)) = -inf:
```

The common shifter is introduced with an additional equation:

```
= lambda0a.1o(i,0,0,0,0,0) $ (energyProj("EnergyScen0","Demand",i,0,0) $ (lambda0a.range(0a,i,knest,0) ne 0) $ p_0f(i,0,0,0,0,0)) ..
lambda0a.1o(i,0,0,0,0,0) =L- sum(sameas(actPpNest,i),chixacta(actPpNest,i)):
```

The template allows to **target the intermediate product, endowment and tech nest demand for an activity**, examples are shown in the EXCEL example above. It swaps the fixed demand to the same change compared to the base for both vintages by endogenizing the shifter *lambdaxFac*. Setting the scale field to 999 removes the equation *e_lambdaXFac* which by default would determine the variable based on a set of other shifter variables.

```

--- fix factor demands according to projection (can be, for instance, total energy demand
or individual intermediate demands)

xtac.fx(shockedFpNest,sRnor,v,kstk) $ (energyProj("KenergyScen",shockedFpNest,sRnor,kstk) $ (years("KenerProjStartYear") 1e years(kstk))
$ xtac.l(shockedFpNest,sRnor,v,t0))
= xtac.l(shockedFpNest,sRnor,v,t0) * energyProj("KenergyScen",shockedFpNest,sRnor,kstk)
/energyProj("KenergyScen",shockedFpNest,sRnor,t0);

vander prod shifters endogenous for energy related nests with fixed demands

lambdaxFac.scale(shockedFpNest,sRnor,v,kstk) $ (xtac.range(shockedFpNest,sRnor,v,kstk) ng 0) = 999;
lambdaxFac.lo(shockedFpNest,sRnor,v,kstk) $ (xtac.range(shockedFpNest,sRnor,v,kstk) ng 0) = -int;
lambdaxFac.up(shockedFpNest,sRnor,v,kstk) $ (xtac.range(shockedFpNest,sRnor,v,kstk) ng 0) = +int;
lambdaxFac.l(shockedFpNest,sRnor,v,kstk) $ (xtac.range(shockedFpNest,sRnor,v,kstk) ng 0) = lambdaxFac.l(shockedFpNest,sRnor,v,kstk 1);

```

The template can also target the demand for a production factor (product, endowment, tech nest) at the level of activity aggregates:

```

--- intermediate, factor and composite demand for sector aggregation is targeted

xtacAgr.fx(actFpNest,aagr,kstk) $ (energyProj("KenergyScen",actFpNest,aagr,kstk) $ (years("KenerProjStartYear") 1e years(kstk)))
= sum(flagXFac(actFpNest,a,vp) $ mapAagr(aagr,a),
xtac.l(actFpNest,a,vp,t0)
* energyProj("KenergyScen",actFpNest,aagr,kstk)/energyProj("KenergyScen",actFpNest,aagr,t0));

shift productivity for intermediate demand of the product

chixFacAgr.l(aagr,actFpNest,kstk) $ (xtacAgr.range(actFpNest,aagr,kstk) ng 0) = chixFacAgr.l(aagr,actFpNest,kstk);
chixFacAgr.lo(aagr,actFpNest,kstk) $ (xtacAgr.range(actFpNest,aagr,kstk) ng 0) = -int;
chixFacAgr.up(aagr,actFpNest,kstk) $ (xtacAgr.range(actFpNest,aagr,kstk) ng 0) = +int;

```

This requires an additional equation:

```

--- the equation is paired with chixFacAgr and activated only if the shift factor is endogenous

e_xFacAgr(aagr,actFpNest,ts(t)) $ ( sum(mapAagr(aagr,a),1) $ (chixFacAgr.range(aagr,actFpNest,t) ne 0)
$ energyProj("%energyScen%",actFpNest,aagr,t)) ..

xFacAgr(actFpNest,aagr,t)
/ max(%minScale%,sum(flagXFac(actFpNest,a,vp) $ mapAagr(aagr,a),xfac.l(actFpNest,a,vp,t-1))*%scaleMult%)
= e= sum(flagXFac(actFpNest,a,vp) $ mapAagr(aagr,a),xfac(actFpNest,a,vp,t))
/ max(%minScale%,sum(flagXFac(actFpNest,a,vp) $ mapAagr(aagr,a),xfac.l(actFpNest,a,vp,t-1))*%scaleMult%);

```

Energy efficiency projections are implemented for the “ener” tech nests:

```

shift energy efficiency

lambdaxFac.scale("ener",sRnor,v,kstk) $ (energyProj("KenergyScen","engFac",sRnor,kstk) $ (years("KenerProjStartYear") 1e years(kstk))
$ xtac.l("ener",sRnor,old,t0))
= 999;
lambdaxFac.fx("ener",sRnor,v,kstk) $ (energyProj("KenergyScen","engFac",sRnor,kstk) $ (years("KenerProjStartYear") 1e years(kstk))
$ xtac.l("ener",sRnor,old,t0))
= energyProj("KenergyScen","engFac",sRnor,kstk);

```

Finally, **world import and export prices** can be driven by the template:

```

--- fix world market price projections to input

pwmRow.fx(iImp(shockedI),%tt%) $ (energyProj("%energyScen%", "pwm",shockedI,%tt%) $ (years("%enerProjStartYear") 1e years(%tt%)))
= energyProj("%energyScen%", "pwm",shockedI,%tt%)/energyProj("%energyScen%", "pwm",shockedI,t0);

pweRow.fx(iExp(shockedI),%tt%) $ (energyProj("%energyScen%", "pwm",shockedI,%tt%) $ (years("%enerProjStartYear") 1e years(%tt%)))
= energyProj("%energyScen%", "pwm",shockedI,%tt%)/energyProj("%energyScen%", "pwm",shockedI,t0);

```

Reporting back-end

Concept

The reporting back-end provides a larger set of pre-defined reports as tables, maps or graphs for model analysis. Its concept is summarized by the following points¹⁷:

- Each report combines a block of information helping to analyze specific aspects of the model solution.
- This often implies looking simultaneously at different model variables, say quantities, pre- and post-tax prices or values which is cumbersome with the model listing or GAMS Studio.
- The back end therefore removes the technical disaggregation of the model's solution space into GAMS variables, parameters and equations.
- Each report comprises at least the following information:
 - Which dimension (for instance, SAM row or column) from *p_toGUI* is shown in columns and rows, (where applicable column or row groups) of the user's viewport, and which are moved to drop down lists. This information is called the pivot. Maps and graphs used columns and rows to define axis respectively to find regional labels.
 - The selection of labels from each dimension in *p_toGUI*, for example, if sectors or products are shown. If no filter is given for a dimension, all found labels are included in the report. The selection can be based on a REGEX statement or selection lists defined by the GAMS code.
 - The view type (table, graph, map).

Further information can be added, such as the number of digits shown, if percentages changes should be additionally shown, if empty rows and columns should be hidden etc. The user can change the pivot, view type etc., but cannot add labels which are excluded by the filters which define the report. It is however possible to filter out additional labels.

To enable this concept, variables, parameters and slacks in equations are copied by **toGui.inc** into a n-dimensional parameters (region, item, SAM row, SAM column, time):

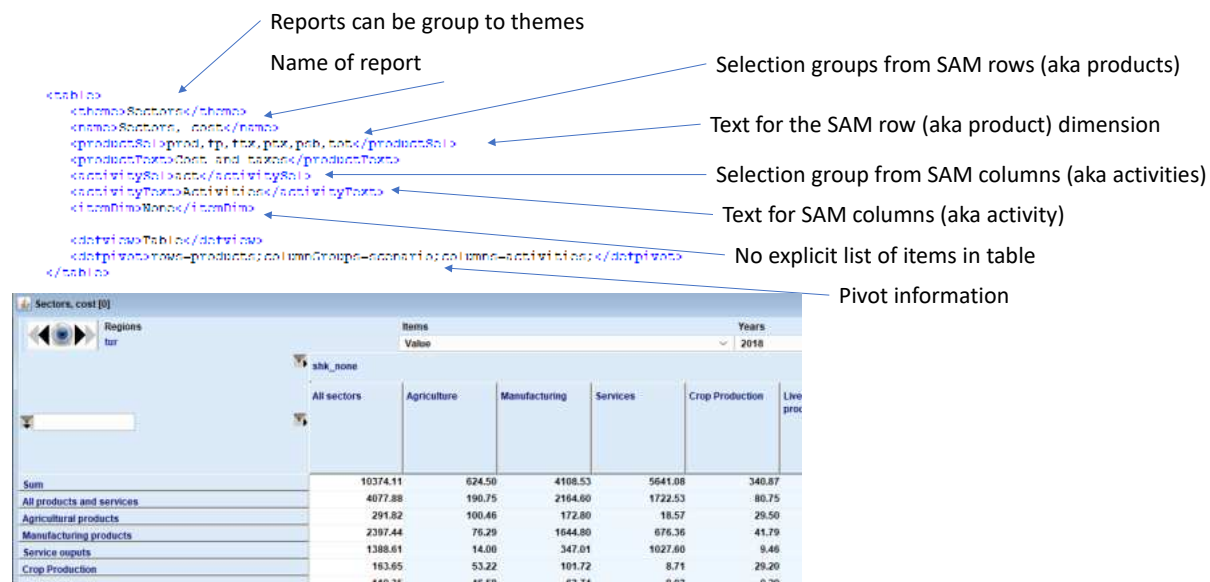
- The item dimension covers Q=quantity, P=Price, V=Value, T=tax rate, G=tax income, plus emissions (set *em*).
- The SAM row dimension also comprises additional variables not found in the SAM, such as macro totals, cohorts, debt related variables, emissions, etc.
- The SAM column also comprises additional detail on such variables, such as emission sources.

If new variables are added to the model and should be reported, they need to be added in the **toGui.inc** file.

An example of how reports are defined in **gui/manageTables.xml** is shown below.

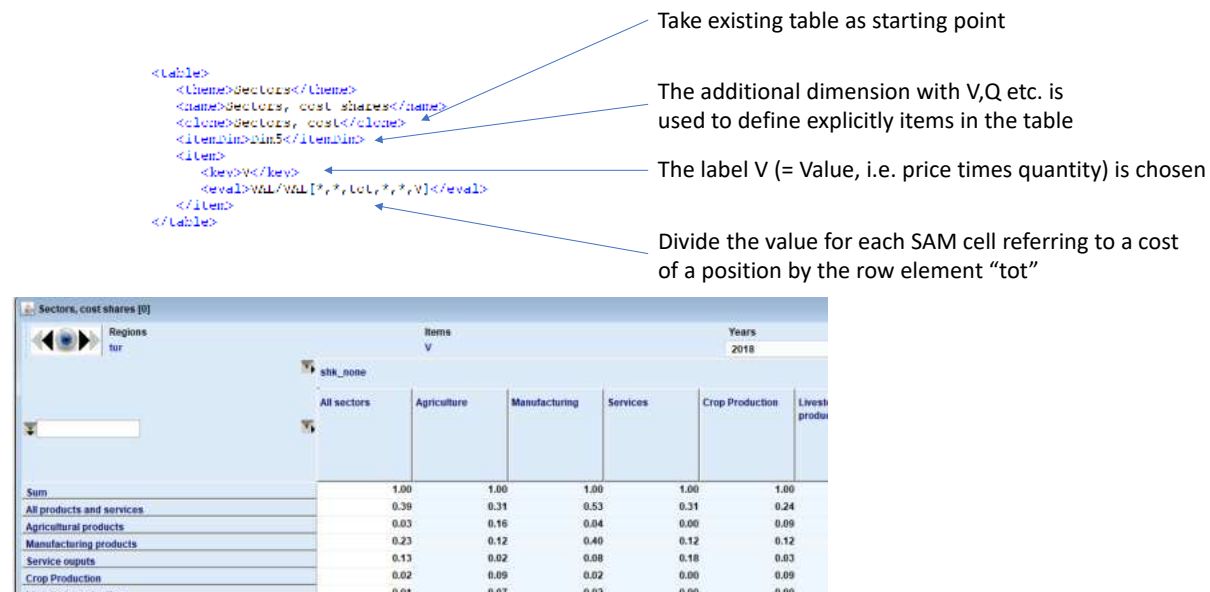
¹⁷ See Britz et al., (2015).

Figure 4.6: Example of a report definition in XML and how it is shown in the reporting back-end



Existing reports can be “cloned” to define variants such as in the following example. It also shows how the expression evaluator built in the reporting tools can be used to define additional items in a table. Note that the evaluator requires to define a list of items for the table manually (not available with `<itemDim>none</itemDim>`).

Figure 4.7: Example of a report definition in XML which clones another report



Using the reporting back-end and available reports

The back end is opened by pressing “Exploit results” from the button bar:



This opens a new content panel which allows you to select the data set to use and shows the GDX containers with results available for this data set. To remove a scenario, leave the box empty. It is recommended to have the BAU scenario last.

Figure 4.8: Scenario selection panel of the reporting back-end

In order to return to the model steering controls, press return. When the “Show results” button is pressed, the GUI will load the GDX containers and show the views (if possible) opened during the last session. On a first-time load, the first report titled “Economy wide totals” will be shown:

Figure 4.9: First report in reporting back-end

View Handling Windows Options

Economy wide totals [0]

Regions

TUR

Items

Value

NoPSB

2018

2019

2020

2021

2022

2023

2024

2025

2026

GDP at market prices

3911.24

4321.43

4655.88

5598.58

6119.64

6560.69

7015.07

7490.71

808

GDP at factor cost

3360.11

3830.78

4142.73

4980.48

5438.29

5828.53

6232.31

6657.92

719

Household consumption

2235.26

2530.82

2699.33

3122.95

3370.94

3578.04

3787.80

4010.66

429

Government consumption

455.57

409.40

409.94

382.18

382.13

382.61

381.82

384.46

38

Investments

904.38

1179.14

1320.82

1778.78

2005.92

2204.78

2414.36

2629.07

289

Total exports

1286.67

1302.62

1341.97

1676.65

1864.09

2030.18

2203.41

2382.90

253

Total imports

1140.68

1305.28

1343.84

1665.17

1844.34

2005.25

2174.40

2350.29

249

B.O.T.

145.99

-2.66

-1.87

11.48

19.75

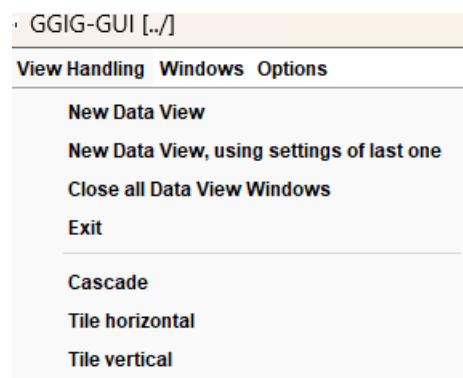
24.94

29.01

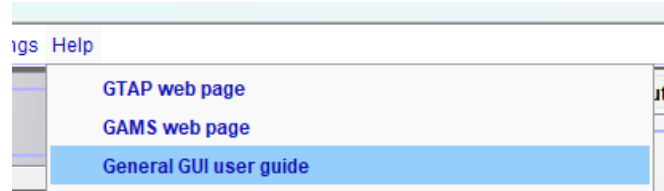
32.62

3

In order to return to the control settings, use "View Handling", "Exit":



Note that multiple reports can be kept open. Further detail on using the reporting tool can be found in the user guide for the reporting back-end available under the help menu. It details the various options to work with tables and graphs such as changing the pivot, filtering, sorting, hiding empty rows and columns, defining percentage differences etc.



The reports currently derived from the model solutions are:

- National accounts:
 - Macro totals (GDP, BOT etc.)
 - Accounts for government, household, savings, enterprise (income and expenditure flows), some key household indicators
 - Household indicators (population, CPI, real income and expenditures, money metric)

- Tax income composition
- Capital stock and depreciation
- Savings composition
- Debt
- Demography
- Sectors: Costs (intermediates, factors, factor taxes), cost shares, tech nests, tech nests as cost shares, make matrix
- Households: detailed expenditures, product demands, budget shares, detailed income composition
- Government: consumption, income flows
- Taxes: product, factor, sector related
- Markets: Factor markets, output/exports/domestic, sales/imports/domestic market appearances, demand at agent prices, Armington decomposition w/wo margins
- Air emission, emission inventories, GHG emissions
- Model diagnostics

It can be expected that with increasing use, more tables and variants of tables will be added.

5. Social Accounting Matrix: Estimation and Update

5.1 Constructing SAMs for MANAGE-WB from the GTAP data base

Motivation and data processing

The GTAP center produces the de facto standard data set for global multi-regional CGE modeling. The latest version 11 being publicly released in spring 2023 covers 160 single countries with a benchmark year of 2017. While in some cases, the IO-Tables underlying single countries are rather old and have not much detail, the database is still a good starting point for a quick start for single country work.

Figure 5.1: Options to construct a SAM from GTAP data base

The program **gtapSAMV6.gms**, developed conducts the following steps:

1. Generates sets from the GTAP data base sets (activities, products, factors etc.).
2. It maps the different matrices of the global GTAP to a single country SAM in MANAGE-WB format.
3. On-demand, it adds data on land use and land cover at the level of Agri-Ecological Zones and integrates them into the SAM.

4. On-demand, it uses the gender differentiated labor database matching GTAP Version 10/11 to define SAM accounts relating to labor differentiated by skill and gender. Countries newly available in Version 11 use the shares of the regional aggregate they were comprised in Version 10 if no new data are available. The more detailed skill categories in the Version 11 database are linked to the skill levels in the gender differentiated labor base via mappings.
5. On-demand, it splits the land account into irrigation water rents and land rents, based on FAO data on area and output by crop and production system (rainfed or irrigated).
6. On-demand, it introduces two new sectors: cattle raising (derived from own use of ruminant meat and raw milk in the respective activities) and electrified other transport. For the latter, the production value is split according to the case share of electric use in the activity; its energy demand is set to the electric energy demand of the other transport activities and all other cost shares scaled to exhaust the constructed output. The remaining (fossil based) other transport activity is treated analogously.
7. Can filter out small shares. A setting of 1.E-10 will skip this step.
8. Balances this SAM in order to remove any remaining tiny imbalances.
9. Produces a simple macro-SAM from this micro-SAM.
10. Generates a new directory under **reg** for the region if not yet existing.
11. Copies a default template for the excel based input for SAMEST described above into the region directories and adds the generated proto-SAM and macro-SAM to the EXCEL sheet.
12. Copies a default template of the bridge file into the region directory and populates it with the SAM.
13. Reads in data on external debts and stores them into the **defpar** sheet.
14. Copies the CDE demand parameters, aggregated according to the mapping information in the **mapk** sheet, into the **hohpar** sheets and adds regression coefficients to updated them based on per capita income changes.
15. Reads in MFMod GDP projections and stores them into dynamics, along with time series which show the VA and GDP composition at the benchmark and some other indicators.

Standard aggregation

The template assumes a 1:1 aggregation from the initial SAM to the proto-SAM with the exemption of electricity which is aggregated over the different power generation products.

Missing bits and pieces

1. Currently, the user has to define the domestic debt stock relative to government consumption on the GUI and provides related interest and re-payment rates. Equally, the new debt (= government budget deficit) has to be defined relative to government consumption.
2. The program assumes that 50% of capital returns plus depreciation constitutes the enterprise income. They are used to finance depreciation, the remainder flows as a transfer to households.
3. If the gender differentiated labor force is not used, no data on physical employment is available. In this case, **inc/base.inc** will use the SAM entries as distribution keys to generate a labor force equal to 2/3 of the population of the 15-64 cohort, at a wage rate of 10.000.

Once such a database is constructed, it can be fed in the process detailed next, for instance for further splitting, or can be directly used to construct a baseline.

5.2. Updating SAMs to macro-totals, balancing and splitting

Updating an existing SAM based on Macro totals or Macro SAM and SAM balancing are an integral part of the MANAGE-WB system. Competing approaches to do so are discussed in the literature (e.g., Britz, 2021). Both from a computational and methodological viewpoint, so-called Highest Posterior Density (HPD) estimators are inviting. They can be understood as a Bayesian framework where the given data act as a-priori information and the balancing constraints as the data information. If normal distributed errors are assumed, the log likelihood of the maximization of the posterior density leads to minimization of squared (relative) errors which is computationally efficient.

The updated SAMEST program presented in this chapter was developed out of code used so far by the MTI-CGE group drawing on a Cross-Entropy approach with requires for each estimate a set of support points with attached a-priori probabilities. The support points capture typically moments of a distribution to approximate (such as mean and variance). The entropy criterion then minimizes the differences between the a-priori and posteriori distributions based on its specific loss function. While (cross-)entropy approaches have come in vogue a while ago, often accompanied by not-well funded criticism about alternative methods, cross-entropy requires more assumptions compared to a HPD framework and results are often harder to understand as the support space implicitly also restricts the estimates. The previous code of SAMEST comprised a range of quite evolved options with regard to Cross-Entropy such as different numbers of supports, cell specific choices for assumed standard errors, and with regard to data handling more generally such as automatic scaling etc. Very little of this flexibility was actual in use, leading to a large code base not easy to understand. The refactored version of SAMEST discussed here removes most of this flexibility to arrive at a streamlined version, offers core options as GUI controls and is based on a HPD estimator.

5.2.1 GUI choices

The user is supposed to put the EXCEL file which comprises the proto and the macro-SAM or macro-totals and related sets into the regional specific directory where later the bridge-file resides which is updated by SAMEST. Available GTAP data sets are searched for in the data directory by the GUI and shown in the GTAP data selection control. They can be used for splitting.

Figure 5.2: Core GUI controls for SAMEST

The screenshot displays the 'General settings' window of the SAMEST software. The window is divided into several sections:

- General settings:** Includes tabs for 'General settings', 'Options for damages', and 'GAMS'.
- Input:** Contains dropdown menus for 'Region input dir' (set to 'BEN'), 'SAM input (xlsx)' (set to 'SAM_BEN_for-update_v1b'), and 'GTAP data' (set to 'gtapData11p'). It also has checkboxes for 'Write new SAM to bridge file' (checked) and 'Choose the Bridge file to write' (set to 'BaseBridge_BEN').
- Algorithm control:** Includes a dropdown for 'Macro control' (set to 'Macro SAM'), a checkbox for 'Column/row sum targets', a text input for 'Weights for coefficients' (set to '0.5'), a text input for 'Scaling factor for SAM' (set to '1'), and a text input for 'Cutoff value for small SAM cells 1.E' (set to '-3').
- Split control for activities:** Contains four rows of controls for 'Lower bound for make matrix entries', 'Lower bound for factor demand', 'Lower bound for intermediate demand', and 'Lower bound for indirect firm taxes'. Each row has two text inputs, both set to '0.30' and '100.00'.
- Split control for products:** Contains three rows of controls for 'Lower bound for final demand', 'Lower bound for trade', and 'Lower bound for indirect product taxes'. Each row has two text inputs, both set to '0.30' and '100.00'.

Macro information can be inputted in the updating steps either based on a Macro SAM or Macro totals. The latter is an option currently being explored, in order to use the MFMod database in case a Macro SAM is not available.



5.2.3 Main processing steps

The main steps in the application of SAMEST are:

1. Reading in the proto-SAM and macro-SAM and related sets and maps from the user supplied EXCEL file
2. Potentially aggregating the proto-SAM
3. Updating the (aggregated) proto-SAM based on the macro-SAM or macro-totals
4. Balancing of the updated SAM in a HPD framework, a macro-SAM is treated as fixed and given and must be balanced.
5. Defining split shares and application of a second HPD framework to balance the split accounts (on-demand)
6. Output of the resulting SAM into the bridge file

Data input

As before, the code reads the data, related sets and maps from the EXCEL sheet as selected on the GUI.

```

$$CALL "GDXIN i-reg\%regUtapiName%\%samXL8%.xlsx o-reg\%regUtapiName%\%samXL8%.gdx index-Layout!A4 MaxTypeErrors = 100"
$$GDXIN "reg\%regUtapiName%\%samXL8%.gdx"
  Macro sam Sets
  $$LOADdc ac acini<samini.diml aciniMac
  $$LOADdc c mrgrn a f l k h g ginv e i w emrgn etax
  $$LOADdc gt gtind gtdir
  $$LOADdc acm
  $$LOADdc macro macro2<macroMat.diml
  $$LOADdc macroam2
  $$LOADdc macromat
  $$LOADdc macrocale
  $$loaddc macmap3 samini map acini ac map acini acini
  $$GDXIN

```

If macro totals are not used, the table can be left empty. Similarly, empty sets and an empty table can be used if no macro-SAM is used. GTAP data are only read and processed if related split sets are not empty (not shown here).

Initial aggregation

The code allows to first aggregate the original proto-SAM based on the *map_acini_acini* set.

```

sam{ac:Tni,ac:Tnip,"read"} = samTni{ac:ini,ac:inip};

sam{ac:Tni,ac:Tnip,"read"}
= sum{(map_acTni_acTni{ac:Tnip,ac:Tni},ac:Tnip)} $ map_acTni_acTni{ac:Tnip,ac:Tni},samTni{ac:Tnip,ac:Tnip)};

```

The set must provide a full mapping (including the 1:1 cases), as otherwise, certain SAM accounts will be empty.

Macro SAM updating

In case of a macro-SAM being used, some checks are run first to ensure that each proto-SAM account is mapped exactly to one macro-SAM account.

```

*
* --- check for correct mapping of macro accounts
*
Parameter macMapCount(ac) "Count of ac entries in macmap3";
macMapCount(acnt) = sum(macmap3(acm,acnt), 1);

abort $ sum(acnt $ (macMapCount(acnt) ne 1),1)
      "All macMapCount entries should be 1", macMapCount;

abort $ (sum(acnt, macMapCount(acnt)) ne card(acnt))
      "Not all macro accounts are mapped to a account in the proto SAM",macMapCount;

macMapIni(acm,acIni) $ sum(map_acIni_ac(acIni,ac),macmap3(acm,ac)) = YES;

```

In order to update the proto-SAM based on a given macro-SAM, first a macro-SAM from the given proto-SAM is defined, based on the mapping information provided in the EXCEL sheet. This allows to define shares from the proto-SAM into its own macro-SAM.

```

* --- define macro SAM shares from SAM as read in

parameter macTotIni(acm,acmp)
      macShares
;
macTotIni(acm,acmp) = sum((macmapIni(acm,acInipp),acInipp) $ macmapIni(acmp,acInipp),
      sam(acInipp,acInipp,"read"));
macShares(acInipp,acInipp)
      = sum((macmapIni(acm,acInipp),acmp) $ (macmapIni(acmp,acInipp) $ macTotIni(acm,acmp)),
      sam(acInipp,acInipp,"read") / macTotIni(acm,acmp));

```

The special set *mapMapIni* can be used for 1:1 assignments from the given macro-SAM to the proto-SAM update. Related shares for these cases are hence unity.

```

alias (acIniMac,acIniMacp);
macShares(acIniMac,acIniMacp) $ sum((macMapIni(acm,acIniMac),acmp) $ macMapIni(acmp,acIniMacp),1) = 1;

```

Given the shares, either derived from the original proto-SAM or as 1:1 mappings to the given macro-SAM, we update the original proto-SAM.

```

* --- create the initial sam by applying shares derived from read-in SAM to Macro SAM

sam(acTni,acTnipp,"ini") = sum((macMapTni(acm,acTni),acmp) $ macMapTni(acmp,acTnipp), macSam(acm,acmp,"ini") * macShares(acTni,acTnipp));

```

This implies that blocks of SAM cells are updated with the same multiplicative factor. This will usually imply that the update results in an unbalanced SAM which motivates the follow up SAM balancing step. It is also useful if the original proto-SAM is not updated, but not yet fully balanced.

Balancing framework for the non-split proto-SAM

In contrast to the previous version of SAMEST, the preparation of split shares and balancing of the split cells is now handled in a second separate processing step as discussed below. This increases transparency and makes it easier to organize the workflow.

During balancing of the updated proto-SAM, all activities related costs as well as final demands are targeted as coefficients on the related column totals (total cost or expenditures). These coefficients are by far the largest part of the SAM such that they tend to dominate the outcome of the estimator. To match the coefficients, the estimator can also be tempted to put larger changes on account totals. To avoid this, both column/row sum targets can be used and the weights for relative deviations from the coefficient targets can be reduced. The user can also fix specific SAM cells as discussed below.

Based on the HPD approach, the balancing framework thus minimizes relative quadratic deviations for coefficients for all costs positions of activities and for final demand of households, government and for investment. All other cells are targeted as absolute values. Currently, all errors are defined as additive and sign changes are not allowed. The code, as before, will keep zero cells at zero. Column sum constraints are added on-demand only, and only if the column or related row comprise more than two non-zero entries. Row sum constraints are introduced if the related column total is empty. If both a column and row sum is given, the average of the two is targeted.

The first two constraints in the estimation framework target in absolute terms non-zero SAM cells *irow,icol*. The selection of these pairs is based on the *iValue* matrix. The errors can be defined as additive or multiplicative (currently not in use).

```
e_samFlow1(irow,icol) $ (iValue(irow,icol) and aCell(irow,icol)) ..
v_tSam(irow,icol)    =E= sam(irow,icol,"target") + v_errCell(irow,icol);

e_samFlow2(irow,icol) $ (iValue(irow,icol) and lCell(irow,icol)) ..
v_tSam(irow,icol)    =E= sam(irow,icol,"target")*EXP(v_errCell(irow,icol));
```

Similarly, the matrix *iCoeff* defines the cells where coefficients on related column totals are targeted, motivating the following two equations:

```
e_samCoeffA(irow,icol) $ (iCoeff(irow,icol) and aCell(irow,icol)) ..
v_coeff(irow,icol)    =E= (coeff0(irow,icol) + v_errCell(irow,icol));

e_samCoeffL(irow,icol) $ (iCoeff(irow,icol) and lCell(irow,icol)) ..
v_coeff(irow,icol)    =E= coeff0(irow,icol)*EXP(v_errCell(irow,icol));
```

Column and row sum errors can be defined on-demand, triggered by their presence on the sets *colTarget* or *rowTarget*.

```
.
* --- Column and row sum constraints
.
e_colSum(icol2) $ colTarget(iCol2) ..
v_colSum(icol2)  =E= colTarget(icol2) + v_errCRSum(icol2,"col1");
.
e_rowSum(irow2) $ rowTarget(iRow2) ..
v_rowSum(irow2)  =E= rowtarget(irow2) + v_errCRSum(irow2,"row1");
.
```

Column and row constraints together with additional constraints which equilibrate column and row sums ensure that the SAM is balanced.

```
* --- Define column and row sums and coefficients

e_colSumDef(icol) ..
v_colSum(icol)    =E= sum(irow, v_tSam(irow,icol));

e_rowSumDef(irow) ..
v_rowSum(irow)    =E= sum(icol, v_tSam(irow,icol));

* --- Row and column sum equality constraint
e_sam(acbal) ..
v_colSum(acbal)   =E= v_rowSum(acbal);
```

The targeted coefficients are defined from the column sum and the cell estimates.

```
.
e_coeffDef(irow,icol) $ ((not izero(irow,icol)) and icolnz(icol) and iCoeff(irow,icol)) ..
v_coeff(irow,icol)*v_colSum(icol) =E= v_tSam(irow,icol);
.
```

Two sets of constraints (from which only one is active) match the cell estimates against the macro-SAM or macro totals.

```
* -- Macro-sam constraints

e_macroSam(acmnt,acmntp) $ macmNz(acmnt,acmntp) ..

macSam(acmnt,acmntp,"used") =E=
  sum((acnt,acntp) $ (macmap3(acmnt,acnt) and macmap3(acmntp,acntp)),
    v_tSam(acnt,acntp));

* -- Macro constraints (different from v_macSam)

e_macro(macro2) $ macTotal(macro2,"scaled") ..
  sum((irow,icol), macAgg(irow,icol,macro2)*v_tSam(irow,icol)) =E= macTotal(macro2,"scaled");
```

In contrast to previous versions, no separate “aggregated SAM” is present. Equally, errors on the macro-SAM or the macro totals are no longer supported.

The objective function minimizes squared relative errors. If all weights are unity, this implicitly assumes that all errors have the same coefficient of variation.

```
e_hpd ..
v_hpd {
  sum((icol2,rwl) $ (colTarget(icol2) $ sameas(rwl,"col")),1)
  sum((ikow2,rwl) $ (rowTarget(ikow2) $ sameas(rwl,"row")),1)
  sum((ikow,icol) $ ((sam(ikow,icol,"target") $ ivalue(irow,icol)) or (coeffTarget(ikow,icol) $ iCoeff(irow,icol)), p Wgt(ikow,icol))
  /100
  -E- sum((icol2,rwl) $ (colTarget(icol2) $ sameas(rwl,"col")),
    oqr(v errCKSum(icol2,rwl)/colTarget(icol2)))
  sum((ikow2,rwl) $ (rowTarget(ikow2) $ sameas(rwl,"row")),
    oqr(v errCKSum(ikow2,rwl)/rowTarget(ikow2)))
  sum((ikow,icol) $ ((sam(ikow,icol,"target") $ ivalue(irow,icol)) or (coeffTarget(ikow,icol) $ iCoeff(irow,icol))),
    oqr(v errCell(ikow,icol)/((sam(ikow,icol,"target") $ ivalue(irow,icol))
      coeffTarget(ikow,icol) $ iCoeff(irow,icol)) ^ p Wgt(ikow,icol));
```

Fixing of estimates

A regular headache for an analyst is the fact that such a balancing framework will never exactly recover a target for a single cell which is not equal to a macro-SAM cell or a macro total. The old approach allowed to provide standard errors for each individual cell. Setting the standard errors to zero for specific estimates implicitly removed the related error from the estimation framework. The estimate was then forced to the target. This option is now supported by the sheet “fixes” in the EXCEL workbook.

	A	B	C	D	E	F
1	debt-0	g-govt-0	9,077,280			
2						
3						
4						
5						
6						
7						
8						

Navigation bar: dictionary | Layout | Sets | **Macro-SAM** | Proto-SAM | Fixes

The example above fixes new government debt. This option must be used with caution as too many fixed entries are likely to (1) provoke infeasibilities and (2) shift the balancing errors into other cells. Fixes can be applied in the step where the proto-SAM is balanced or in a follow-up split step.

Splitting

The next step consists of splitting certain accounts (on-demand). This is regularly done to introduce multiple households in MANAGE-WB. Multiple households are typically not part of the proto-SAM. In other cases, also specific activities or commodities are split, such as for power generation.

In the first step, split factors as provided by the user in the EXCEL sheet are applied. Split factors are scaled to unity such that, for instance, missing split factors for households will imply that each household gets the same share on the total assigned as the targeted value. Such cases should be avoided, if possible.

```

--- missing user provided split shares are set to unity, 1:1 relation to SAM updated from Macro SAM
sumSplit(acTni,acTnP) = sum((map_acTni_ac(acTni,ac),acp) & (map_acTni_ac(acTni,acp) & split(ac,acp)) , splitShr(acTni,ac,acTnP,acp)) z
splitShr(acTni,ac,acTnP,acp) & ((map_acTni_ac(acTni,ac) and map_acTni_ac(acTni,acp))
& (not sumSplit(acTni,acTnP) & split(ac,acp)) = 1 z

sumSplit(acTni,acTnP) & ((not sumSplit(acTni,acTnP)))
= sum((map_acTni_ac(acTni,ac),acp) & map_acTni_ac(acTni,acp),splitShr(acTni,ac,acTnP,acp)) z
splitShr(acTni,ac,acTnP,acp) & splitShr(acTni,ac,acTnP,acp) = splitShr(acTni,ac,acTnP,acp)/sumSplit(acTni,acTnP) z

```

As a second step, the user can provide mapping relations to GTAP factors, activities and commodities via the EXCEL sheet, and indicate in two sets which activities and commodities are to split based on the GTAP data. If one of two sets is non-empty, the GTAP database is processed, and split factors are derived.

```

set asplit_gtap(a),csplit_gtap(c);
$$GDXIN "reg\%regGtapName%\%samXLS%.gdx"
$$LOAD asplit_gtap,csplit_gtap
$$GDXIN

$$setglobal useGtap off
$$ife card(asplit_gtap)>0 $$setglobal useGtap on
$$ife card(csplit_gtap)>0 $$setglobal useGtap on

$$ifthen1.useGtap "%useGtap%"=="on"

```

We skip here the code which defines different vectors from the GTAP database used for splitting. Based on this information, split shares specific to GTAP *p_splitShr2* are derived and overwrite split shares in *p_splitShr* as loaded from EXCEL. Note here that split share loaded from the EXCEL file (indicated by the indicator parameter *splitShrOri*) are multiplied with the GTAP derived factors. This can have hard to control consequences and it is generally not recommended to mix the two approaches.

```

ac(ac) & sum(acp, quantShr(ac,acp)) = YTSz
--- calculate proteasam with disaggregations based on GTAP
splitShr(acTni,ac,acTnP,acp) & ((map_acTni_ac(acTni,ac) and map_acTni_ac(acTni,acp) & (not splitShr(acTni,ac,acTnP,acp))) = 1 z
splitShr2(acTni,ac,acTnP,acp) & (map_acTni_ac(acTni,ac) and map_acTni_ac(acTni,acp) & splitShrOri(acTni,ac,acTnP,acp))
= splitShr(acTni,ac,acTnP,acp)*quantShr(ac,acTni,acp,acTnP) z
splitShr2(acTni,ac,acTnP,acp) & (map_acTni_ac(acTni,ac) and map_acTni_ac(acTni,acp) & (not splitShrOri(acTni,ac,acTnP,acp)))
= quantShr(ac,acTni,acp,acTnP) z
splitShr2(acTni,ac,acTnP,as) & ((map_acTni_ac(acTni,ac,as) and map_acTni_ac(acTni,as)) & (not splitShr2(acTni,ac,acTnP,as))) = cpxz
splitShr(acTni,ac,acTnP,acp) & splitShr2(acTni,ac,acTnP,acp) = splitShr2(acTni,ac,acTnP,acp) z

```

The combined split factors define the dis-aggregated proto-SAM which after scaling and cleansing of small values enters the balancing framework.

```
sum(ac,acp,"dis") = sum{(acTnI,acTnIp) $ (map_acTnI_ac(acTnI,ac) and map_acTnI_ac(acTnIp,acp)),sum(acTnI,acTnIp,"ini")*splitShr(acTnI,ac,acTnIp,acp)};
sum(ac,acp,"dis") $ (sum(ac,acp,"dis") - eps) = 0;
```

Note that in contrast to factor taxes, use taxes from GTAP are not processed as the structure of the taxes in MANAGE SAM can deviate substantially from the way taxes are defined in GTAP. In order to use the information from GTAP to split tax accounts, a file "sam_corr.inc" can be generated and placed in the folder for the region.

Note here the following:

1. If split shares are derived from summation (such as for total use of a commodity as in the case for commodity taxes, see first two lines), the weights are not the summed-up split-shares, but the related targets.
2. Weights should be scaled before being used as split shares.
3. Empty split shares from GTAP (= real missing values) are flagged with eps. A division by eps provokes a math trap error. Use dollar operated to avoid this.

Bounds on split cells

The controls shown above under "Split Control for activities" and "Split control for products" allow to restrict for groups of split cells the introduction of bounds relative to their targets. This is useful to control large differences to target values and to easily find problematic split shares. The settings are used as follows in the code where an upper limit of 100 is treated as an infinite bound.

```
*
* --- reflect general bounds on split cells from interface
*
v_splitCellLo(ac,α) $ (sum{α,α,"dis"} > 0) $ (sum{α,α,"dis"} > 0) $ (splitBoundMakeMatrix ne 100) - sum{α,α,"dis"} * $lowBoundMakeMatrix$;
v_splitCellUp(ac,α) $ (sum{α,α,"dis"} > 0) $ (sum{α,α,"dis"} > 0) $ (splitBoundMakeMatrix ne 100) - sum{α,α,"dis"} * $splitBoundMakeMatrix$;

v_splitCellLo(I,α) $ (sum{I,α,"dis"} > 0) $ (sum{I,α,"dis"} > 0) $ (splitBoundFacDemand$ ne 100) - sum{I,α,"dis"} * $lowBoundFacDemand$;
v_splitCellUp(I,α) $ (sum{I,α,"dis"} > 0) $ (sum{I,α,"dis"} > 0) $ (splitBoundFacDemand$ ne 100) - sum{I,α,"dis"} * $splitBoundFacDemand$;

v_splitCellLo(α,α) $ (sum{α,α,"dis"} > 0) $ (sum{α,α,"dis"} > 0) $ (splitBoundTnlDemand$ ne 100) - sum{α,α,"dis"} * $lowBoundTnlDemand$;
v_splitCellUp(α,α) $ (sum{α,α,"dis"} > 0) $ (sum{α,α,"dis"} > 0) $ (splitBoundTnlDemand$ ne 100) - sum{α,α,"dis"} * $splitBoundTnlDemand$;

v_splitCellLo(qI,α) $ (sum{qI,α,"dis"} > 0) $ (sum{qI,α,"dis"} > 0) $ (splitBoundTndFirmTaxes$ ne 100) - sum{qI,α,"dis"} * $lowBoundTndFirmTaxes$;
v_splitCellUp(qI,α) $ (sum{qI,α,"dis"} > 0) $ (sum{qI,α,"dis"} > 0) $ (splitBoundTndFirmTaxes$ ne 100) - sum{qI,α,"dis"} * $splitBoundTndFirmTaxes$;

v_splitCellLo(qI,β) $ (sum{qI,β,"dis"} < 0) $ (sum{qI,β,"dis"} < 0) $ (splitBoundTndFirmTaxes$ ne 100) - sum{qI,β,"dis"} * $lowBoundTndFirmTaxes$;
v_splitCellUp(qI,β) $ (sum{qI,β,"dis"} < 0) $ (sum{qI,β,"dis"} < 0) $ (splitBoundTndFirmTaxes$ ne 100) - sum{qI,β,"dis"} * $splitBoundTndFirmTaxes$;

v_splitCellLo(α,IIn) $ (sum{α,IIn,"dis"} > 0) $ (sum{α,IIn,"dis"} > 0) $ (splitBoundFinDemand$ ne 100) - sum{α,IIn,"dis"} * $lowBoundFinDemand$;
v_splitCellUp(α,IIn) $ (sum{α,IIn,"dis"} > 0) $ (sum{α,IIn,"dis"} > 0) $ (splitBoundFinDemand$ ne 100) - sum{α,IIn,"dis"} * $splitBoundFinDemand$;

v_splitCellLo(α,w) $ (sum{α,w,"dis"} > 0) $ (sum{α,w,"dis"} > 0) $ (splitBoundTrade$ ne 100) - sum{α,w,"dis"} * $lowBoundTrade$;
v_splitCellUp(α,w) $ (sum{α,w,"dis"} > 0) $ (sum{α,w,"dis"} > 0) $ (splitBoundTrade$ ne 100) - sum{α,w,"dis"} * $splitBoundTrade$;

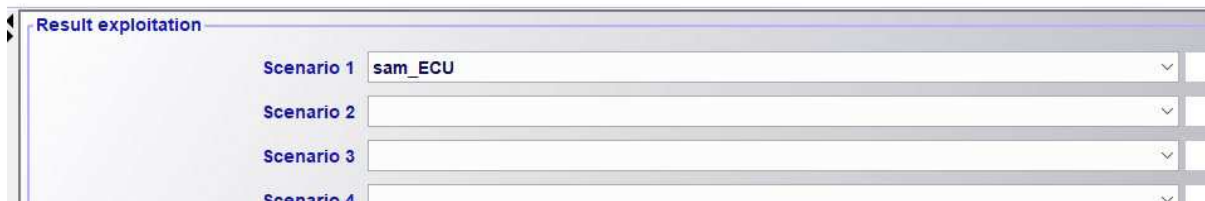
v_splitCellLo(w,α) $ (sum{w,α,"dis"} > 0) $ (sum{w,α,"dis"} > 0) $ (splitBoundTrade$ ne 100) - sum{w,α,"dis"} * $lowBoundTrade$;
v_splitCellUp(w,α) $ (sum{w,α,"dis"} > 0) $ (sum{w,α,"dis"} > 0) $ (splitBoundTrade$ ne 100) - sum{w,α,"dis"} * $splitBoundTrade$;

v_splitCellLo(qI,α) $ (sum{qI,α,"dis"} > 0) $ (sum{qI,α,"dis"} > 0) $ (splitBoundTndProdTaxes$ ne 100) - sum{qI,α,"dis"} * $lowBoundTndProdTaxes$;
v_splitCellUp(qI,α) $ (sum{qI,α,"dis"} > 0) $ (sum{qI,α,"dis"} > 0) $ (splitBoundTndProdTaxes$ ne 100) - sum{qI,α,"dis"} * $splitBoundTndProdTaxes$;

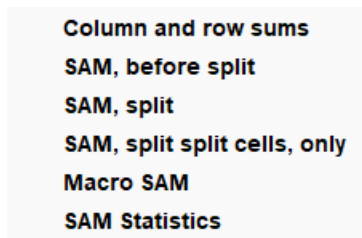
v_splitCellLo(qI,β) $ (sum{qI,β,"dis"} < 0) $ (sum{qI,β,"dis"} < 0) $ (splitBoundTndProdTaxes$ ne 100) - sum{qI,β,"dis"} * $lowBoundTndProdTaxes$;
v_splitCellUp(qI,β) $ (sum{qI,β,"dis"} < 0) $ (sum{qI,β,"dis"} < 0) $ (splitBoundTndProdTaxes$ ne 100) - sum{qI,β,"dis"} * $splitBoundTndProdTaxes$;
```

5.3.2 Exploring the results

The parameter p_SAM which tracks the SAM over the different steps can be loaded into the explorer.



The code offers the following views:



They also allow to compare different versions as shown below:

	target	fixes	solution	% diff	Abs diff	
a-cbcc-0 c-cbcc-0		384.72		369.63	-4.91	18.89
a-cbcc-0 c-wtra-0		4.31		4.31	0.00	0.00
a-cbcc-0 total0		389.63		370.75	-4.85	18.89
a-cere-0 c-cere-0		75.68		70.91	-6.30	4.77
a-cere-0 c-wtra-0		0.43		0.43	-0.04	0.00
a-cere-0 total0		76.11		71.34	-6.27	4.77

By sorting based on the "% diff" or "Abs diff" columns, potentially outliers can be tracked easily.

By, for instance, moving the SAM columns into a box and adding statistics, the maximal and minimal deviations for all non-zero rows for a column become immediately visible.

	fixes	solution	% diff	Abs diff	
Mean	3.40	0.22	3.28	3.40	0.22
Median	0.32	0.02	0.30	0.32	0.02
min	-4.17	0.00	-3.18	-4.17	0.00
max	66.25	2.10	64.15	66.25	2.10
c-ptfs-0	0.11		0.10	-3.21	0.00
c-feed-0	6.76		6.45	-4.63	0.31
c-text-0	0.18		0.17	-3.27	0.01

A similar report is available for the macro-SAM:

	Solution	% diff	Abs diff
DEBT ENT	1684780.00	1684780.00	
DEBT GOV	9077280.00	9077280.00	
DEBT WORLD	8594530.00	8594530.00	
act com	178421392.00	178421392.00	
com act	77549824.00	77549824.00	
com com	15498018.00	15498018.00	
com HH	65291712.00	65291712.00	
com GOV	15583493.00	15583493.00	
com INV	20306584.00	22106694.00	8.14
com INV PUB	7547122.00	7917122.00	-0.38
com STKC	-247240.00	-2017348.88	87.74
com WORLD	24210996.00	24210994.00	-0.00
lab act	40328004.00	40328004.00	
cap act	89491160.00	89491160.00	
HH lab	40328004.00	40328004.00	
HH cap	35737100.00	35737100.00	
HH ENT	1190154.25	1190155.25	0.00
HH GOV	242376.89	242376.89	

Equally, some key statistics can be inspected:

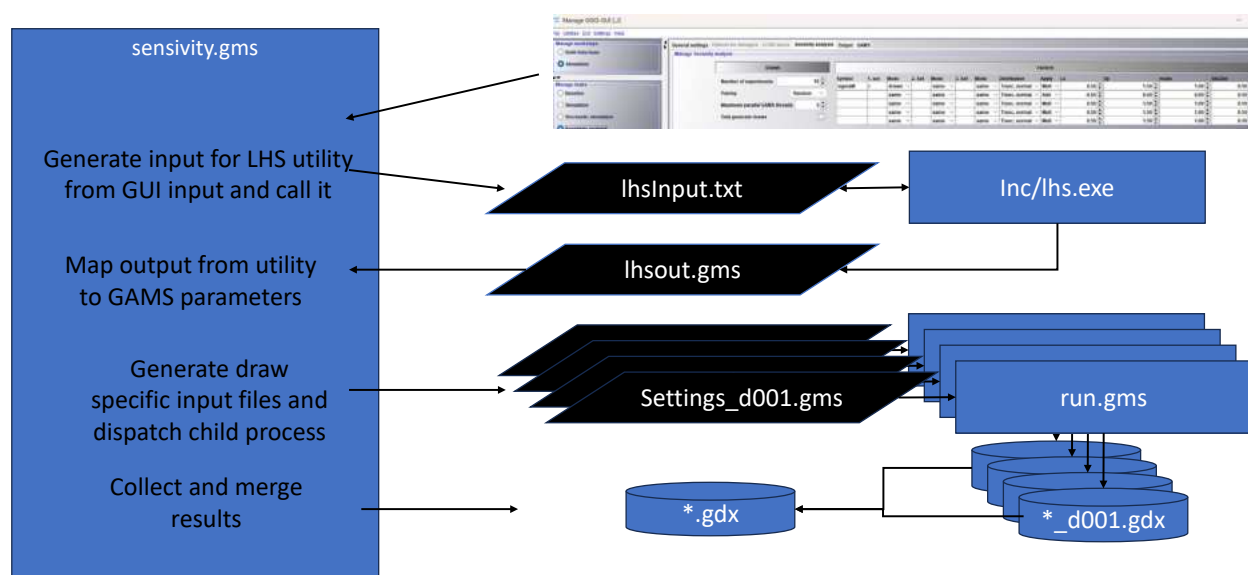
	target	solution
nonZcnt	3853.00	3848.00
Mean	18.45	18.49
Max	1962.08	2058.84
min	-201.73	-201.73
StdDev	93.86	95.20
1DevLoCnt	4.00	4.00
1DevHiCnt	126.00	125.00
2DevLoCnt	1.00	1.00
2DevHiCnt	69.00	66.00

6. Sensitivity analysis

The sensitivity analysis builds on the simulation task of MANGE-WB by adding a Latin-Hypercube Sampling (LHS) based approach to conduct stratified random sampling of user defined model parameters or exogenous shock elements. The use of LHS gives a good coverage of probability distributions with limited number of draws, important, as each draw requires a separate model run.

An overview of the process is given below. The user defines the parameters subject to sensitivity analysis and their distribution on an interface control detailed below. This input serves enters the LHS utility which will define the draws for each individual experiment based on this information. The output from the utility is transformed by the GAMS code of the mother process **sensitivity.gms** into GAMS statements which are inputted into a run-specific include file read by **run.gms**. The mother process will start individual simulation runs; on demand in parallel. Each run will store its results into a separate GDX container of which the name comprises a field referring to the experiment number, such as *d001* in the example below. Once all simulations are run, the mother process will collect and merge the results from all runs into one single GDX container which the user can then explore with the reporting backend.

Figure 6.1: Overview on sensitivity analysis



The additional GUI control for the sensitivity analysis, see Figure 6.2, comprises on the left-hand side a panel with options which refer to the overall set-up of the analysis and not the individual parameter. The *number of experiments* defines the number of simulations runs to perform and is equal to number of quantiles covered by LHS. LHS will guarantee that there is exactly one draw from the distribution of each random variable in each quantile. The start number allows to run the sensitivity analysis in blocks such that repeated runs of the utility don't overwrite previous results. Using this option requires however an additional program if results from multiple runs of the utility need to be combined.

The option *"Restricted"* for pairing will let the LHS utility target a zero-co-variance matrix between each pair of random vars by reshuffling the drawn random numbers. The success of this will depend on the number of random variables and experiments, with a higher number of random variables and a lower number of experiments increasing the chance to larger non-zero co-variances. *"Random"* pairing will not control for the co-variances, resulting in a faster execution of the LHS utility.

The *"Maximum parallel GAMS threads"* controls how many MANAGE-WB simulations are run in parallel. Once this maximum is reached, the mother process waits until one of the runs is finished before starting another one. It is recommended to use less than the available physical cores available on the CPU as the maximum. The parallel execution can be switched off by setting the control to 1. Equally, as solvers such as CONOPT4 or PATH can use parallel cores, the GAMS run parameter threads should be set to 1 if multiple simulation runs are performed in parallel.

The option *"Only generate draws"* will report the drawn values for the random vars in the listing file, but not start the actual simulation runs. This is useful to control the set-up of the sensitivity analysis before the potential time-consuming actual analysis is started.

Figure 6.2: GUI control for overall set-up of sensitivity analysis.

The right-hand side of the tab comprises a user-edited table, see Figure 6.3. The field *"Symbol"* defines the parameter or variable to disturb (non-existing ones in the GAMS code will be defined as an additional parameter). The follow up elements refer to the names of the sets found on up to three dimensions of the parameter (parameters with more dimensions are not supported), followed by mode field which can be set to *"draws"* or *"same"*. Using *"draws"* implies that each element on the set will be treated like a separate random variable where *"same"* will use the same realization for all set elements. For instance, in the example below, the one-dimensional parameter *p_ghgRedRate* will have different

realizations for the different time point as depicted by the set *grctt* in each run. If “same” is used instead, for a single run, the same realization would be found for all time points. The other three random variables are scalars and the name for all sets is left empty.

The follow up control for each parameter defines the type of distribution to use (truncated normal, uniform, triangular), if the random realization should be added to or multiplied with the given parameter value from the base line or found in the shock file, and the lower and upper limit for the distribution, its mode and the standard deviation, the latter only for the truncated normal distribution.

Figure 6.3: GUI control to define parameters as random vars and related distributions

Factors													
Symbol	1. set	Mode	2. Set	Mode	3. Set	Mode	Distribution	Apply	Lo	Up	mode	Std.Dev	
p_salesTaxRecl		same		same		same	Uniform	Mult	0.10	1.90	1.00	0.50	
p_factTaxRecl		same		same		same	Uniform	Mult	0.10	1.90	1.00	0.50	
p_dirTaxRecl		same		same		same	Uniform	Mult	0.10	1.90	1.00	0.50	
p_ghgRedRate	grctt	draws		same		same	Trunc. normal	Mult	0.50	5.00	1.01	1.00	
		same		same		same	Trunc. normal	Mult	0.50	1.50	1.00	0.10	

Results shown from a sensitivity analysis will show simultaneously the outcomes from all runs as seen below:

Overview [0]

Regions

tza

Items

Quantity

Draws

base

tza_ccdr

2019

2020

2021

2022

2023

2024

2025

2026

2027

2028

2029

2030

2031

Real GDP at market prices

53.19

54.11

56.56

59.51

62.84

66.55

71.29

76.22

81.34

86.53

91.96

97.64

103.56

Real GDP, p.a. change

1.73

4.52

5.22

5.59

5.91

7.12

6.91

6.72

6.38

6.27

6.18

6.06

By using a pivot which shows the different runs in the rows of a table, their outcomes can be directly compared:

Overview [0]

Regions

tza

Items

Quantity

Products

Real GDP at market prices

tza_ccdr

	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031	2032	2033	2034	2035	2036
base	53.19	54.11	56.56	59.51	62.84	66.55	71.29	76.22	81.34	86.53	91.96	97.64	103.56	111.13	119.22	127.56	136.15	
d1	53.19	54.11	56.56	59.51	62.84	66.55	71.29	76.21	81.33	86.52	91.94	97.61	103.50	111.07	119.15	127.48	136.07	
d2	53.19	54.11	56.56	59.51	62.84	66.55	71.29	76.21	81.34	86.53	91.96	97.64	103.56	111.18	119.26	127.58	136.16	
d3	53.19	54.11	56.56	59.51	62.84	66.55	71.29	76.22	81.34	86.53	91.95	97.62	103.52	111.12	119.19	127.52	136.10	
d4	53.19	54.11	56.56	59.51	62.84	66.55	71.28	76.20	81.31	86.48	91.88	97.52	103.37	110.89	118.96	127.28	135.84	
d5	53.19	54.11	56.56	59.51	62.84	66.55	71.29	76.21	81.33	86.52	91.95	97.63	103.55	111.14	119.23	127.57	136.15	

Note here that the run “base” is one with unchanged parameters from the benchmark. By adding relative or absolute changes against the outcome of run “base”, the changes provoked by the different random realizations of the parameters can be directly seen:

Overview [0]																											
Regions				Items		Products										Percentage diff. to											
tza				Quantity		Real GDP at market prices										Draws											
																Base											
	2027	2028	2029	2030	2031	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044	2045	2046	2047	2048	2049	2050			
Mean	81.34	86.53	91.96	97.64	103.56	111.12	119.21	127.55	136.14	144.84	153.88	163.27	172.81	182.58	192.52	202.61	212.75	222.89	233.07	244.70	256.12	267.87	279.88	292.06			
%	-0.00%	-0.00%	-0.00%	-0.00%	-0.00%	-0.01%	-0.01%	-0.01%	-0.00%	-0.00%	-0.00%	-0.00%	-0.00%	-0.00%	-0.00%	-0.00%	-0.00%	-0.00%	-0.00%	-0.00%	-0.00%	-0.01%	-0.01%	-0.01%			
min	81.31	86.48	91.88	97.52	103.37	110.89	119.06	127.28	135.84	144.61	153.62	162.89	172.39	182.13	192.04	202.10	212.19	222.38	232.99	243.95	255.28	266.92	278.81	290.86			
%	-0.04%	-0.02%	-0.01%	-0.01%	-0.02%	-0.22%	-0.22%	-0.22%	-0.22%	-0.24%	-0.24%	-0.24%	-0.25%	-0.25%	-0.25%	-0.25%	-0.25%	-0.25%	-0.25%	-0.25%	-0.25%	-0.25%	-0.25%	-0.25%			
max	81.38	86.60	92.06	97.78	103.78	111.37	119.49	127.86	136.49	145.32	154.40	163.74	173.32	183.14	193.12	203.27	213.47	223.78	234.54	245.66	257.18	269.03	281.16	293.47			
%	0.00%	0.02%	0.11%	0.10%	0.21%	0.21%	0.23%	0.24%	0.25%	0.28%	0.28%	0.28%	0.29%	0.29%	0.31%	0.32%	0.32%	0.32%	0.32%	0.32%	0.37%	0.38%	0.41%	0.43%	0.43%		

It is also possible to add statistics such as “mean”, “max” and “min” to a table, as depicted above.

The sensitivity analysis is hence a tool to assess result robustness or to improve the scenario definition by testing alternative parameterization of shock files. The set-up can be in most cases directly defined via the GUI without further coding efforts, and the reporting tool allows the user to easily derive statistics on reported model outcomes.

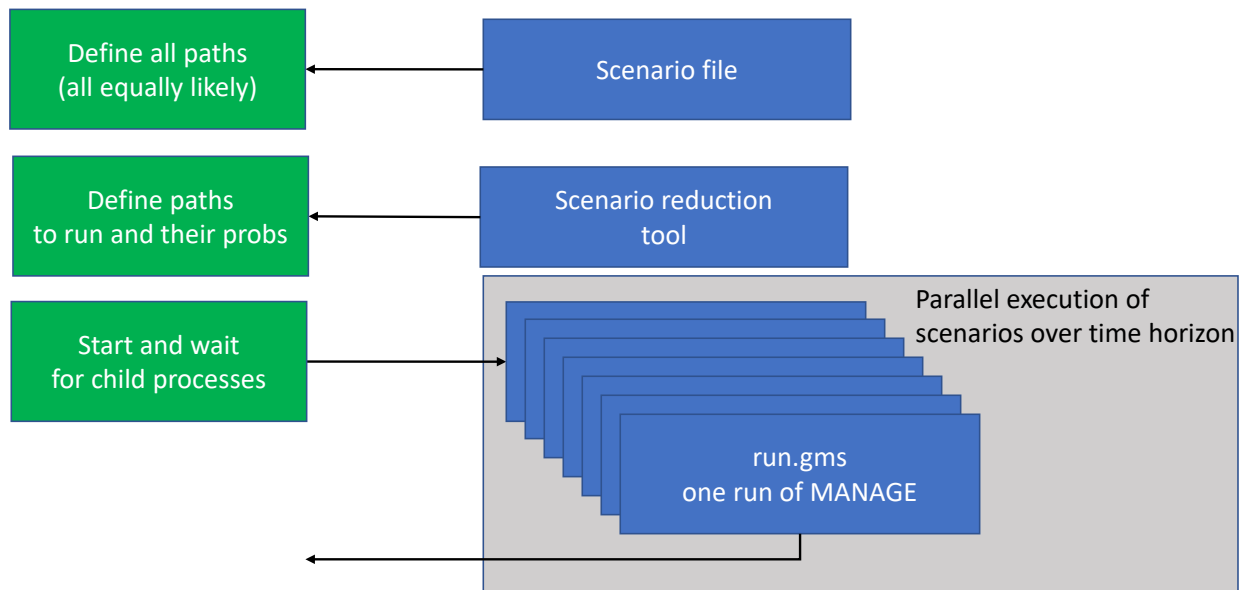
7. Stochastic simulation runs

The stochastic simulation mode is similar to the sensitivity analysis as it also performs a pre-defined number of simulations run. However, whereas sensitivity analysis uses LHS which results in a set-up where each single simulation is equally likely, the stochastic simulation runs are suitable for cases where also small probability shocks with high impacts are analyzed. Typical examples are catastrophic weather-related events such as tropical storms or floods which combine low probabilities with high damages. Their so-called Expected Annual Damages (EAD) are often rather low as high damage events have often return periods as high as 100 or 1000 years. In order to capture an event with a 1000-year return period (= 0.001% probability to occur in any year), a LHS approach would require already 1.000 quantiles. If the probability for a storm or flood to occur is low, most of these 1000 draws would run a no-change experiment compared to a baseline where no storms or floods are considered if the sensitivity analysis would be used. The stochastic simulation run utility therefore uses a specific algorithm, a so-called "scenario reduction" algorithm¹⁸, which combines similar outcomes from a large number of Monte-Carlo draws into one experiment. If there are many similar ones Monte-Carlo draws (such as no or quite limited storms or floods occurring), these are summarized by one run with a higher probability, only. Compared to LHS, this leaves more room for runs referring to less likely outcomes.

The following 7.1 gives an overview on the stochastic extension of MANAGE-WB. The user will define in its shock file the Monte-Carlo draws. Each time series with random variables drawn, typically damages, defines one so-called path from the base to last simulation year. The scenario reduction tool will analyze all these paths and average similar one into actual runs to perform. The GAMS mother will run these as individual MANAGE-WB simulations in parallel, as in sensitivity analysis, and collect and merge the results into one GDX container.

¹⁸ H. Heitsch and W. Römisch. Scenario tree reduction for multistage stochastic programs. Computational Management Science, 6:117-133, 2009

Figure 7.1: Overview on sensitivity analysis



In opposite to the sensitivity analysis, the use of the stochastic extension requires a specific set-up of the shock file, in the example used below, disasters which destroy a part of the capital stocks are modelled by random shocks to the depreciation rate. The shock files need to be organized as follows:

1. Prepare for the scenario tree reduction in your shock file

```
$$include '%incf%/iniStoch.gms'
```

2. Draw randomly realizations of the shock and put into pre-defined parameter *p_randvar*

```
p_randVar("depr",nodes) = sum(t_nodes(tCur,nodes),p_depr(tCur) * max(1,normal(1,0.2)));
p_randVar("depr","n1") = p_depr("%baseYear%");
```

3. Let the tool summarize to the desired # of scenarios to run

```
$$include '%incf%/runStoch.gms'
```

4. And pick the realizations for a specific draw later *run.gms*

```
-----
depr.fx(t) = sum(t_nodes(t,nCur) $ draws_nodes(draws,nCur), p_randVar("depr",nCur));
```

These parts are in the following shown as they would appear in a shock file:

```
-----
depr.fx(t) = sum(t_nodes(t,nCur) $ draws_nodes(draws,nCur), p_randVar("depr",nCur));
```

The following example shows a shock file drawing on the example above:

```

$iftheni.decl "%1"--"decl"
    set randVars / "depr" /;
    $$include '%inof%/iniStoch.gms'
$elseif.decl "%1"--"stoch"
    parameter p depr(t); p depr(t) = depr.1(t);
    p randVar("depr",nodes) = sum(t nodes(tCur,nodes),p depr(tCur) * max(1,normal(1,0.2)));
    p randVar("depr","n1") = p depr("%baseyear%");
    $$include '%inof%/runStoch.gms'
$else.decl
    depr.fx(t) = sum(t nodes(t,nCur) $ draws nodes("d%iScen%",nCur), p randVar("depr",nCur));
$endif.decl

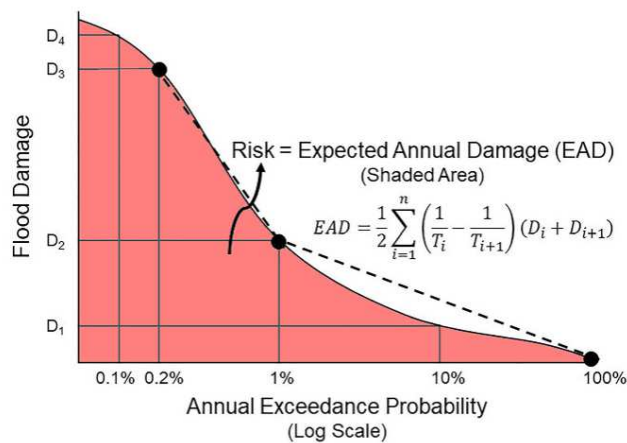
```

Needed for both child and mother:
necessary set definitions

Only used by mother process:
Define random vars
and apply scenario
reduction

Used in child process ("run.gms") to pick specific realizations
for current scenario %iScen%

Damages reported by size for a limited number of selected return period provide a typical application field. The expected annual damage is regularly calculated by linear interpolation between given points, assuming that the damage size is inversely proportional on these segments:



From: Bilskie, M. V., Angel, D. D., Yoskowitz, D., & Hagen, S. C. (2022). Future flood risk exacerbated by the dynamic impacts of sea level rise along the northern gulf of Mexico. *Earth's Future*, 10(4), e2021EF002414.

The following code uses this approach to define random draws where the GAMS random variable generator for the uniform distribution is used to draw a probability between 0 and 1, and the GAMS code interpolate the damage between the segment encompassing the drawn probabilities.

The input consists of the damage estimates, here an increase in the depreciation rates stored in the table *p_drm*, for different return periods. A T5 (= five-year return period) event has a probability of 20% and leads in the example to a very modest increase in the depreciation rate of 0.001, when compared to a 100-year event (1% probability) which increases the deprecation rate by 0.04:

```

sel. rel.Per "Return periods" / T1,T5,T10,T50,T100,Tend //

Table      p_drm "Increase in depreciation for disasters with a certain return period"
      dmq      prob
T1      1.7e-10      1
T5      0.001      0.2
T10     0.01      0.1
T50     0.02      0.02
T100    0.04      0.01
Tend    0.04      0.00

/

--- First draw probability from uniform
p_randVar{"depr",nodes} = uniform(0,1);

--- map to damage based on trapezoidal rule (i.e. the damage is proportional to the
    probability)
p_randVar{"depr",nodes} = p_depr("Stochastic")

+ sum{rel.Per & { (p_randVar{"depr",nodes} <= p_drm{rel.Per-1,"prob"}) & {p_randVar{"depr",nodes} <= p_drm{rel.Per,"prob"}}},
    p_drm{rel.Per-1,"dmq"} + (p_drm{rel.Per,"dmq"} - p_drm{rel.Per-1,"dmq"}) * (p_drm{rel.Per-1,"prob"} - p_drm{rel.Per,"prob"})) ;

```

Results from the stochastic extension can be analyzed similar to those from a sensitivity analysis, however, the user now needs to reflect that the probability of the different runs will differ. The in-built statistics calculated and shown by reporting tool will consider this automatically. The following screenshots show how to add some statistics:

The screenshot shows the 'Set statistics' dialog box in a software interface. The dialog has the following settings:

- ☒ Treat zeros as missing values for statistics
- ☐ Only show outliers
- Set maximum percentage of outliers: 2
- Set outlier detection method: No outlier detection
- Select statistics:
 - Sum
 - Nobs
 - Mean
 - Median
 - StdDev
 - q1
 - q3
 - min
 - max
 - minOutlier
 - maxOutlier
 - freeEval
- Free evaluation field: (empty)
- Buttons: ok, Cancel, Update

The background shows a table with columns for years 2018-2023 and rows for different regions (d1-d21). The table contains numerical values for each cell.

Which gives:

Capital stock and depreciation [0]

Regions: TUR
Products: Depreciation rate
TUR_shk_drm

	2018	2019	2020	2021	2022	2023	2024	2025	2026
Mean		0.051974	0.051476	0.051959	0.051732	0.051886	0.051728	0.051664	
StdDev		0.005210	0.005158	0.005210	0.005185	0.005200	0.005184	0.005177	
q1		0.050000	0.050000	0.050000	0.050000	0.050000	0.050000	0.050000	
q3		0.052556	0.050000	0.052005	0.052266	0.052189	0.051604	0.051238	
min		0.050000	0.050000	0.050000	0.050000	0.050000	0.050000	0.050000	
max		0.066921	0.068120	0.068789	0.067204	0.068732	0.067254	0.063054	
d1		0.050000	0.050000	0.054656	0.055163	0.052057	0.050000	0.050000	
d2		0.050000	0.053956	0.050000	0.050000	0.054645	0.050000	0.050000	
d3		0.050000	0.050000	0.050000	0.053202	0.050645	0.060437	0.050000	
d4		0.050000	0.050000	0.050000	0.054800	0.058772	0.050000	0.060776	
d5		0.053120	0.050000	0.055613	0.050000	0.050000	0.050000	0.054095	
d6		0.058063	0.050000	0.051071	0.050000	0.055193	0.053813	0.056691	

By solely selecting the statistics, an overview on the distribution can be gained:

Overview [1]

Regions

TUR

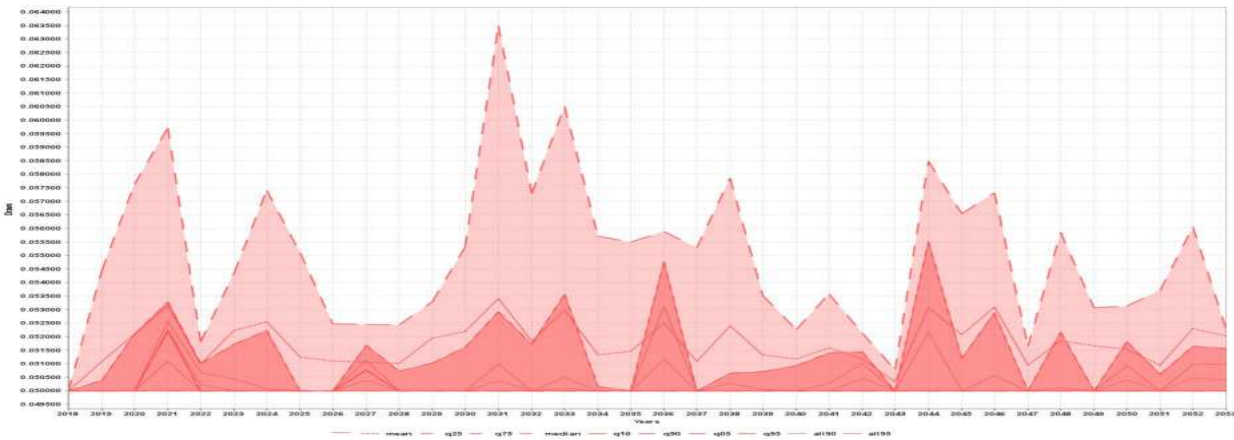
Items

Quantity

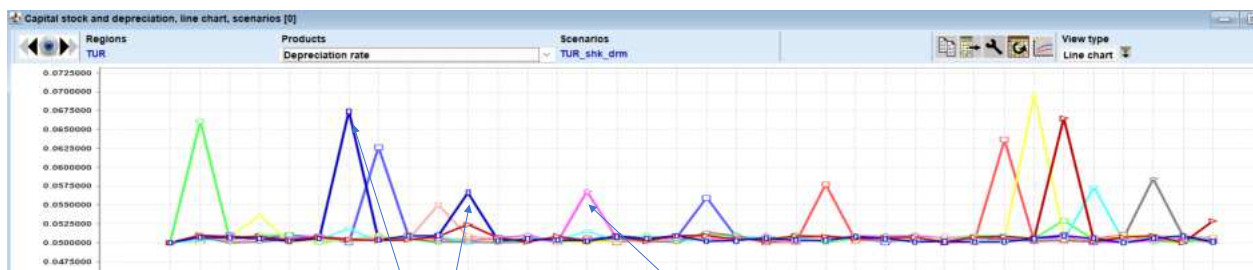
TUR_shk_drm

		2018	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028
Real GDP at market prices	Mean	3724.39	3752.33	3820.66	4251.87	4460.08	4681.33	4935.40	5202.95	5492.30	5808.65	6127.99
	Median	3724.39	3754.18	3822.06	4254.54	4457.52	4678.26	4935.68	5202.15	5492.93	5809.31	6129.00
	min	3724.39	3740.85	3801.73	4232.40	4437.83	4660.37	4907.42	5147.04	5435.26	5756.19	6076.68
	max	3724.39	3754.18	3825.51	4261.62	4474.70	4698.44	4956.85	5229.48	5522.33	5831.93	6152.68
Real GDP, p.a. change	Mean		0.75	1.82	11.29	4.90	4.96	5.43	5.42	5.56	5.76	
	Median		0.80	1.90	11.38	4.96	5.00	5.50	5.50	5.60	5.80	
	min		0.44	1.27	11.00	4.46	4.67	4.59	4.88	5.21	5.54	
	max		0.80	1.90	11.40	5.03	5.04	5.56	5.55	5.67	5.90	
Real GDP at market prices, per capita	Mean	44975.47	44947.88	45410.86	50154.51	52261.77	54550.63	57215.15	60013.41	63033.98	66332.23	69699.00
	Median	44975.48	44970.14	45427.45	50185.98	52231.72	54514.84	57218.33	60004.18	63041.25	66339.79	69700.00
	min	44975.48	44810.46	45185.84	49924.91	52001.02	54306.41	56890.78	59368.56	62379.42	65733.24	69000.00
	max	44975.48	44970.14	45468.51	50269.55	52433.06	54750.00	57463.80	60319.38	63378.64	66598.05	70000.00
Real Household consumption, per capita	Mean	25456.29	25842.26	27215.93	30048.45	31061.07	32116.13	33458.23	34849.20	36716.66	38761.05	40819.00
	Median	25456.29	25838.88	27225.63	30070.02	31078.01	32090.24	33424.31	34864.59	36710.59	38772.35	40820.00
	min	25456.29	25838.88	27112.60	29847.35	30917.28	31923.86	33275.66	34583.34	36249.77	38319.55	40000.00
	max	25456.29	25865.20	27286.61	30123.59	31210.63	32235.06	33723.14	35016.26	36953.71	39029.92	41000.00
Private consumption, real on real GDP	Mean	56.60	57.49	59.93	59.91	59.43	58.87	58.48	58.07	58.25	58.43	
	Median	56.60	57.46	59.88	59.85	59.38	58.85	58.41	58.02	58.23	58.40	
	min	56.60	57.46	59.86	59.78	59.32	58.78	58.35	57.94	58.11	58.30	
	max	56.60	57.72	60.39	60.14	59.83	59.13	59.28	58.37	58.50	58.64	
Government consumption, real, share on real GDP	Mean	15.52	15.52	15.52	15.52	15.52	15.52	15.52	15.52	15.52	15.52	
	Median	15.52	15.52	15.52	15.52	15.52	15.52	15.52	15.52	15.52	15.52	
	min	15.52	15.52	15.52	15.52	15.52	15.52	15.52	15.52	15.52	15.52	
	max	15.52	15.52	15.52	15.52	15.52	15.52	15.52	15.52	15.52	15.52	

A so-called "deviation" graph can also help to analyze outcomes:



The following graph shows the generated draws for 10 scenarios with the damage data shown above:



Close to 2% of capital stock destroyed (50 year event)
followed two years later by ~10 year event

A run where solely one rather probable
disaster occurs

Summarizing, the stochastic extension is useful for cases where low probability events are analyzed such that even LHS would require too many runs to capture the distribution. Damages reported for different higher return periods such as 50, 100, 500 or 1.000 years provide a typical use case. The reporting interface supports the analysis by automatically calculated probability weighted statistics and by certain graphs to visualize outcomes.

References

Britz, W. (2021). Comparing Penalty Functions in Balancing and Dis-aggregating Social Accounting Matrices. *Journal of Global Economic Analysis*, 6(1), 34.-81.

Britz, W. (2014). A New Graphical User Interface Generator for Economic Models and its Comparison to Existing Approaches. *German Journal of Agricultural Economics*, 63(4), 271-285.

Britz, W., Pèrez Dominguez, I., Narayanan, G. B. (2015). Analyzing Results from Agricultural Large-scale Economic Simulation Models: Recent Progress and the Way Ahead. *German Journal of Agricultural Economics*, 64(2), 107-119.

Roson, R. (2019). Sectoral Differences in Labor Productivity Growth: Estimation and Modeling. *Research in Applied Economics*, 11(1), 1-8.

Roson, R., Britz, W. (2021). Simulating long run structural change with a dynamic general equilibrium model. *International Journal of Computational Economics and Econometrics*, 11(4), 368-404.

Roson, R., Sartori, M. (2016). Estimation of Climate Change Damage Functions for 140 Regions in the GTAP 9 Data Base. *Journal of Global Economic Analysis*, 1(2), 78-115.

van der Mensbrugghe, D, (2020). The Mitigation, Adaptation and New Technologies Applied General Equilibrium (MANAGE) Model, Version 2.0g, GTAP Technical Paper, TP/20/xx, Center for Global Trade Analysis, Purdue University, West Lafayette, IN.

Global Warming Potentials (IPCC Second Assessment Report). http://unfccc.int/ghg_data/items/3825.php, United Nations Climate Change.

Revised 1996 IPCC Guidelines for National Greenhouse Gas Inventories. <https://www.ipcc.ch/report/revised-1996-ipcc-guidelines-for-national-greenhouse-gas-inventories/>.