

## **A GAMS/MPSGE implementation of the PET model**

Matthias Weitzel, Edward J. Balistreri, Brian C. O'Neill, Xiaolin Ren

**Working Paper 19-WP 593**  
September 2019

**Center for Agricultural and Rural Development**  
**Iowa State University**  
**Ames, Iowa 50011-1070**  
**[www.card.iastate.edu](http://www.card.iastate.edu)**

*Matthias Weitzel is Project Officer, European Commission, Joint Research Centre, Seville, Spain 41092.  
E-mail: [Matthias.weitzel@ec.europa.eu](mailto:Matthias.weitzel@ec.europa.eu).*

*Edward J. Balistreri is Associate Professor, Department of Economics, Iowa State University, Ames, IA 50011. E-mail: [ebalistr@iastate.edu](mailto:ebalistr@iastate.edu).*

*Brian C. O'Neill is Professor, Josef Korbel School of International Studies, University of Denver, Denver, CO 80208. E-mail: [Brian.ONeill@du.edu](mailto:Brian.ONeill@du.edu).*

This publication is available online on the CARD website: [www.card.iastate.edu](http://www.card.iastate.edu). Permission is granted to reproduce this information with appropriate attribution to the author and the Center for Agricultural and Rural Development, Iowa State University, Ames, Iowa 50011-1070.

For questions or comments about the contents of this paper, please contact Edward J. Balistreri, [ebalistr@iastate.edu](mailto:ebalistr@iastate.edu)

Iowa State University does not discriminate on the basis of race, color, age, ethnicity, religion, national origin, pregnancy, sexual orientation, gender identity, genetic information, sex, marital status, disability, or status as a U.S. veteran. Inquiries can be directed to the Interim Assistant Director of Equal Opportunity and Compliance, 3280 Beardshear Hall, (515) 294-7612.

# A GAMS/MPSGE implementation of the PET model

Matthias Weitzel, Edward J. Balistreri, Brian C. O'Neill, Xiaolin Ren

## Abstract:

This paper describes a version of the Population-Economy-Technology (PET) model implemented in the GAMS/MPSGE programming language. The implementation of the model is comparable with the original fortran version of iPETS over a number of test cases. While a number of possible configurations exist for both model types, we demonstrate that there is a configuration that makes both implementations behave very similarly and the remaining difference can be traced to different implementations of dynamic agent behavior. This paper also serves as a methodological blueprint for model translation across different programming languages.

JEL classification: C68, D58, F19, Q54

---

Correspondance: Matthias Weitzel, European Commission, Joint Research Centre (JRC), Seville, Spain ([matthias.weitzel@ec.europa.eu](mailto:matthias.weitzel@ec.europa.eu)); Edward J. Balistreri, Iowa State University, Ames ([ebalistr@iastate.edu](mailto:ebalistr@iastate.edu)); Brian C. O'Neill, Josef Korbel School of International Studies at the University of Denver ([Brian.ONeill@du.edu](mailto:Brian.ONeill@du.edu)).

Part of the work has been carried out at the National Center for Atmospheric Research in Boulder, CO and has benefited from support by the National Science Foundation under Grant Number AGS-1243095. Balistreri acknowledges support from the USDA National Institute of Food and Agriculture (NIFA) Hatch Project 1010309. The authors thank Bas van Ruijven for helpful comments in the comparison of the GAMS and fortran versions of the model. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors. They do not necessarily reflect those of the funders and may not in any circumstances be regarded as stating an official position of the European Commission.

# 1 Introduction

The iPETS model used in NCAR's Integrated Assessment Modeling (IAM) group includes an energy-economic model of the global economy linked to a demographic model and a simple climate model. The energy-economic model is the PET (Population-Economy-Technology) model, a multi-region, multi-sector computable general equilibrium (CGE) model (see [ipetsmodel.com](http://ipetsmodel.com) for complete documentation). The PET model as used in several applications so far is written in fortran (e.g., O'Neill et al., 2010; O'Neill, Ren, Jiang, & Dalton, 2012; Ren et al., 2016; 2018). This paper describes the model's alternative implementation in the General Algebraic Modeling System (GAMS), a higher level programming language. We make use of an extension of GAMS called the mathematical programming system for general equilibrium analysis (MPSGE), which allows solving Arrow-Debreu equilibrium models as a mixed complementary problem (Rutherford, 1999). GAMS/MPSGE has a number of advantages for economic modelling and is the preferred programming environment for many economists.

GAMS has become a standard tool for computational economics and brings along a number of advantages as a modeling and data management language, and it is linked to the most advanced solution algorithms. In particular, GAMS separates model development and model solution into logically separate activities, which means a model is formulated in the code, but then passed on to advanced solvers for the respective problem class. This allows fast model formulation independent of data, the solution algorithm, and the computing platform, and thus allows the model developer to spend more time on improving the science of the model rather than the software engineering to solve the model.

Using the GAMS subsystem MPSGE, CGE models can be set up relatively easily based on flexible production and demand blocks (plus additional constraints). MPSGE automatically generates cost functions to derive zero profit conditions, demand functions to derive market clearance conditions and income balances for regional households and governments (including tax payments). The numeric problem is then passed to GAMS and solved with the PATH solver. The researcher can focus on the economic structure and only has limited direct interaction with the solution algorithm.

There is increased flexibility in adding or changing sectors and in region aggregations in the GAMS version, and the nesting structure of production functions can be easily adjusted in MPSGE. MPSGE also allows for making the model flexible by easily allowing the use of flags to turn on or off specific features. The structure of the standard model can thus be transformed into a model of different flavor relatively easily. Finally, the model code of the new model allows for scripting, e.g. to run several solves with different input parameters to perform an automatic calibration to an exogenous baseline. Key features of the GAMS version of the model are described in Section 2 of this paper.

The goal of the development of the GAMS/MPSGE version of the model was to produce a version of the PET model that is easier to use particularly for model development. For this purpose, we wanted to start with a GAMS version that is as close as possible to the fortran version in terms of the underlying economic structure (expressed as e.g. household's preferences or industries' technologies). Some of this document is therefore devoted to demonstrating that the two versions of the model are functionally equivalent; that is, when running the same experiment, with the same inputs, they produce identical or at least very similar outputs.

At the same time, we also wanted to take advantage of the opportunity of coding a new model from scratch to introduce some differences in the GAMS version in order to improve on some features of the fortran version and its calibration procedure. For the most important of these differences, in order to make sure the models were still equivalent, we added options to one or the other model so that they could be run with comparable features (typically by allowing a particular feature to be turned on or off). Another part of this document is therefore devoted to describing differences between the models, and tests that demonstrate that they are still fundamentally compatible, isolating the effects of particular features on outcomes.

Key features of the GAMS version that differ from the fortran version of the model are described in Section 2. Section 3 shows how model equality was established despite these differences. For several cases, this was done by activating (or deactivating) specific options in the GAMS or fortran model, respectively. Section 4 discusses differences between the models in the calibration to a baseline and section 5 concludes.

## 2 Overview of the PET model and GAMS/fortran version differences

The structure of the GAMS version of the PET model was intended to be the same as (or as similar as possible to) the current implementation in fortran. The PET model structure consists, in each region, of a household sector that consumes goods and makes investments, industries that produce goods, and a government sector that collects taxes and makes transfers to households. Between regions, trade in intermediate goods occurs and is modeled following the Armington (1969) assumption. Households earn income from labor and capital, as well as from government transfers, and make decisions about how much of their income to save versus how much of it to spend on consumption goods. The model assumes forward-looking behavior, i.e. households have perfect foresight about future prices, and make optimal decisions according to a household utility function, taking these prices as given. Industries make decisions about how much of their good to produce, and how to produce it (that is, through which mix of labor, capital, and intermediate inputs such as energy or materials). They make decisions in order to maximize profits, given a set of prices for inputs and outputs. For typical regional and sectoral aggregations see the appendix, and for the explicit nesting of production and consumption functions, see the iPETS documentation wiki<sup>1</sup>.

Differences between the PET model implementations in GAMS and fortran can be categorized as *structural differences* (the economic equations that model economic behavior by households or industries differ, section 2.1) and *model solution differences* (the algorithm to solve for economic behavior differs, section 2.2). With regard to structural differences, the intertemporal optimization of agents differs. While both the fortran and GAMS versions implement forward looking behavior, the agent's option to achieve an intertemporal optimum is restricted in the fortran version to the consumption-investment choice. The GAMS formulation is more general such that forward looking behavior can extend to intertemporal optimization of trade, as well (section 2.1.1). The more general

---

<sup>1</sup> [http://ipetsmodel.com/index.php?title=PET\\_model\\_structure](http://ipetsmodel.com/index.php?title=PET_model_structure)

implementation of intertemporal optimization in the GAMS version can be restricted to resemble the fortran version (section 3.6).

With regard to differences in sectoral aggregation, sections 2.1.2 and 2.1.3 describe structural differences in the modeling of energy and land-using sectors, respectively. It is obvious that these structural differences will directly affect model behavior as the underlying economic optimization problem is different. For comparison of different sectoral aggregation, we adjust the input data, e.g. by not breaking out the land-using sectors to demonstrate equivalence.

To solve the model, the GAMS implementation formulates the economic problem as a mixed complementarity problem (section 2.2.1). This has modeling advantages (e.g. modeling of inactive technologies or linking variables with equations). In practical terms, this mainly affects the implementation of capital dynamics (section 2.2.2) which also has to be formulated within the mixed complementarity framework. As mentioned above, the fortran version is limited to one dynamic state variable (the capital stock, governed by consumption-investment choices). The intratemporal and intertemporal problems can therefore be separated. The more general implementation in GAMS allows for more than one state variable, however, making separation into intratemporal and intertemporal problems infeasible. Hence the capital dynamics in the GAMS model are solved using a different method. Note, however, that despite the different formulation the results are very similar or even identical in the absence of international trade (section 3).

As the GAMS model is formulated in MPSGE, the representation of all production and consumption functions has to follow its special form to represent CES functions. For features of the PET model that are not standard in many typical CGE models, this requires a translation of the equations into GAMS/MPSGE code. In particular, this applies for the dynamic accounting of the capital stock and the intertemporal optimization (section 2.2.2) and the utility function that features (intertemporal) maximization of per capita utility as well as changing preferences over time (section 2.2.3).

Further modeling differences exist related to the terminal condition, but because this affects only the final years we did not observe any meaningful difference in the time horizon of interest (section 2.2.4). Finally, the GAMS model allows for running the model at a reduced temporal resolution (section 2.2.5), which greatly affects the solution time (section 3.8), which is particularly useful at the stage of model development and enhancement.

Differences in the calibration to a baseline are described in section 4. Differences between the models exist in the calibration to energy consumption (section 4.2). Further, the GAMS model allows for additional options to calibrate energy prices (section 4.3), and investment to GDP ratios (sections 4.4 and 4.5).

## 2.1 Structural differences between model versions

### 2.1.1 Intertemporal optimization and trade

A key structural difference between the GAMS and fortran versions is the balance of payments treatment, which influences intertemporal behavior through adjustments in international trade. The fortran version has a period-by-period balance-of-payments constraint. The GAMS version has an intertemporal balance-of-payments constraint.

In the fortran version, each time period  $t$  is solved as a separate sub-problem, and a budget over final demand equation holds in each period. In this balance, for each region  $r$  (index omitted) expenditures for consumption ( $p_{jt}^C C_{jt}$  representing the expenditures for individual good  $j$ ) and investment ( $p_t^I I_t$ ) are equal to net of tax income from labor ( $(1 - \tau_t^L) p_t^L L_t$ ) and capital ( $(1 - \tau_t^K) r_t^K K_t$ ), government transfers  $GT$  and the net trade surplus ( $BOP$ ) expressed in US dollars ( $p_{USA,t}^{\bar{c}} BOP_t$ ):

$$\sum_j p_{jt}^C C_{jt} + p_t^I I_t = (1 - \tau_t^L) p_t^L L_t + (1 - \tau_t^K) r_t^K K_t + GT_t - p_{USA,t}^{\bar{c}} BOP_t.$$

The net trade positions are transfers between the different regions in the model; a net trade surplus means that the region exports more than it imports and that it obtains a claim on future net imports. The exogenous BOP term implies that the balance of trade in each region is fixed at an exogenous value in each year. Convenient choices for this balance of trade value start with the observed base year value, but in principle this is an exogenous choice of the modeler.<sup>2</sup> The formulation with a fixed BOP in each time period allows the economic problem of the agent to be divided into a dynamic problem (with one state variable, i.e. optimizing the intertemporal investment decision affecting the capital stock, which links periods) and static problems for each period. This leads to computational advantages, e.g. being able to parallelize the fortran model (see section 3.8 below).

In the GAMS version, however, the budget over final demand for region  $r$  (index omitted) is formulated over the entire model horizon

$$\sum_{t=1}^T \left( \sum_j p_{jt}^C C_{jt} + p_t^I I_t \right) = \sum_{t=1}^T \left( (1 - \tau_t^L) p_t^L L_t + (1 - \tau_t^K) r_t^K K_t + GT_t - p_{USA,t}^{\bar{c}} BOP_t \right).$$

Because there is no budget constraint for each period, the fortran budget equation (i.e. balancing the budget in each period) does not necessarily hold in the GAMS model. Instead, the net balance of trade between a region and the rest of the world is specified as a sum over the entire model horizon. In particular, regions can adjust net trade intertemporally, thus allowing changes in intertemporal lending/borrowing across countries, which is a real feature of the global economy.<sup>3</sup>

<sup>2</sup> Note that for the initial dynamic steady state (section 2.2.2), this value has to grow at the same rate as all other quantities in the model. This can then be adjusted later during calibration.

<sup>3</sup> For example, net exports can be seen as a form of lending to the rest of the world as it constitutes a claim on future net imports. In a case without trade, this intertemporal borrowing or lending will not occur because individual regions would be isolated. Within each (isolated) region, the value of production has to equal the sum of

This feature essentially gives the GAMS version of the model an additional dimension of intertemporal behavior (beyond forward-looking household savings decisions). This implies more flexibility compared to the fortran version as intertemporal borrowing/lending can be used to shift consumption and investment between time periods. All markets still have to be balanced at each time period, so actors in the model cannot consume goods that were never produced, which is ensured by the market clearance equations in the model. This also implies that (net) trade has to be balanced globally in any given model period, because each imported good was produced as an export good in another region.

To establish model equivalence with regard to budget balance and intertemporal trade behavior, an additional constraint can be added to the GAMS version to resemble the (implicit) constraint of the fortran version. Specifically, an additional balance of trade constraint can be added to the GAMS version in each period so that its time path is the same between models. This forces the balance of trade in each period to an exogenous value which is the same as the one implemented in the fortran version. The constraint consists of endogenous tariffs on imports and exports of all but one region.<sup>4</sup> If in a region exports are too high or imports too low to achieve the exogenous balance of trade constraint, exports will be taxed and imports subsidized until the constraint is achieved. The tax and subsidy instruments are counteracting such that no or little net government revenue is realized. Alternatively, the instruments could therefore be interpreted as import/export productivity adjustments.

Adding this constraint to the GAMS version is important to reproduce the outcomes of the fortran model. The effect of a shock can be quite different between model versions when using the default version of the GAMS model. For a set of simple shocks, it can be demonstrated that a sudden shock is smoothed out in the GAMS version, having prolonged consequences due to intertemporal adjustments in the balance of trade.

When the constraint is added to the GAMS version, the resulting (endogenous) tariffs and subsidies need to be taken into account to properly compare relative prices between regions, as these are the interregional prices that agents are faced with in the model. For example, to compare a price ratio between regions  $A$  and  $B$  between the models, the following adjustment is made in the GAMS version:

$$\frac{P_A^{fortran}}{P_B^{fortran}} = \frac{P_A^{GAMS}(1+\tau_B)}{P_B^{GAMS}(1+\tau_A)},$$

where  $P_A$  and  $P_B$  are two prices in regions  $A$  and  $B$  and  $\tau_A$  and  $\tau_B$  are the endogenous tariff/subsidy rates. This also has implications for comparing GDP or other derived values. Instead of using e.g.  $P_B$  as a numeraire, the numeraire has to include the adjustment term, i.e. be defined as  $P_B(1+\tau_B)/(1+\tau_A)$ .

---

all expenditures and all factor payments. Hence the budget equation holds for each year and there is no net borrowing or lending.

<sup>4</sup> If all but one region achieves the balance of trade constraint, the remaining region will also achieve the constraint by Walras' Law.

### 2.1.2 Energy sectors

The energy sectors explicitly modeled in GAMS and fortran are the same with the exception of renewable energy, which is not explicitly treated in the fortran version. In the fortran version, input from the electricity sector into the electricity sector is interpreted as renewable or carbon-free (including nuclear) electricity generation. It can thus be explicitly reported as a separate generation type. While it is advantageous to have the ability to report carbon-free electricity, the fortran version approach also has some shortcomings, such as unrealistic shares of carbon-free electricity in a climate policy run, driven by an economic response that shifts the input structure towards capital input rather than electricity under a carbon price. This can lead to unrealistic conversion efficiencies in policy scenarios given the modeling scheme implemented in the fortran version.

Given these potential shortcomings and that it might be more promising to completely update the modeling of energy conversion sectors in the future, we decided to not explicitly model carbon-free electricity in the GAMS version. That means that the electricity sector is modeled just as in the fortran version, however, no special meaning is attached to the input of electricity to the electricity sector.

Due to these differences, no comparison runs were made for renewable energy targets or reported renewable energy.

### 2.1.3 Land using sectors

The fortran version includes three land-using sectors (crops, animal products, forestry), while these sectors are currently not included in the GAMS version (see the documentation wiki for implementation details). Versions of the fortran implementation without land use were used for this comparison. An experimental GAMS version with land use has been developed, but has not yet been explicitly compared with the fortran version. This is mainly due to the fact that additional sectors and factors lead to increased complexity. These additional sectors, which often are relatively small relative to the rest of the economy, can easily induce numeric issues and the additional equations and variables lead to further increases in computation time to solve the model. The experimental GAMS version with land has only been run at a lower temporal resolution (see section 2.2.5 below) which is not directly comparable with the fortran version due to differences in time steps. For the calibration of land using sectors to a baseline scenario, the experimental GAMS version uses the same approach as in the energy calibration (and is therefore different from the fortran version in this respect, as well).

## 2.2 Model solution differences

### 2.2.1 Solving the model as a mixed complementarity problem

The GAMS model is formulated and solved as a mixed complementarity problem (MCP). Conventional CGE modelling (including the fortran version of the PET model) relies on a system of equations. In contrast, a MCP formulation links functional equations and variables through weak inequalities. This allows for activities to become "slack" (i.e., inactive). The explicit link between equations and variables has some advantage for model development and debugging and can also convey an economic meaning. For example, industry production becomes "slack" if cost exceeds prices. The complementary weak-inequalities of the MCP characterize three key outcomes of a competitive equilibrium:



- Market clearance conditions indicate no excess supply exists as long as the price of the respective goods or factor market is positive. Mathematically, in equilibrium  $\text{supply} - \text{demand} \geq 0 \perp \text{price} \geq 0$ . The  $\perp$  symbol indicates complementarity, meaning that one of the conditions has to hold with equality in equilibrium, i.e.  $\text{price} * (\text{supply} - \text{demand}) = 0$ . This implies that with excess supply, the price in this market falls to zero.
- Zero profit conditions indicate that no production makes a positive profit; perfect competition is assumed to dissipate all excess profits. However, when cost exceeds revenues, then no production activity will take place in this market. Mathematically,  $\text{cost} - \text{revenue} \geq 0 \perp \text{activity level} \geq 0$ , or equivalently  $\text{activity level} * (\text{cost} - \text{revenue}) = 0$ .
- Household budget balance is achieved by linking income to the budget constraint:  $\text{income} - \text{expenditure} \geq 0 \perp \text{income} \geq 0$ , or equivalently  $\text{income} * (\text{income} - \text{expenditure}) = 0$ . For any active household, the balancing condition holds with equality.

While this modeling technique has some advantages, for the translation of the PET model into GAMS, they are not required and the same solution can be obtained using the solution algorithm of the fortran model. In practical terms, this mainly affects the implementation of capital dynamics (section 2.2.2) which also has to be formulated within the mixed complementarity framework. This implies that the intertemporal capital dynamics have to be formulated as market clearance and zero profit conditions, which might be a less intuitive, but elegant way to re-formulate the problem, as we will show in the next sub-section. This also means that the intertemporal problem is an integral part of the overall MCP problem and that the intertemporal as well as the intra-temporal problems faced by agents are solved simultaneously. In contrast, the fortran formulation of the model separates these two problems and finds solutions by iterating between them until convergence is achieved.<sup>5</sup> The non-linear system of model equations is solved for in a single shot, using a call to the PATH (Newton type algorithm) solver (Ferris & Munson, 2000). In the resulting solution, all activities and relative present-value prices are simultaneously consistent with the formulated equilibrium.

Solving the model for the entire time horizon at once enables the modeler to keep track of more than one state variable. In the fortran model there is one state variable (capital) that is governed by an intertemporal decision. In the standard GAMS model (without additional constraints to induce behavior as in the fortran model as discussed in section 3.6), the balance of trade is also a state variable that is governed by an intertemporal decision, allowing for a more flexible treatment of trade dynamics. Rather than having a budget equation for each time step (as in the fortran version), the agents in the GAMS version are faced with a single budget equation over the entire model horizon, and the balance of trade therefore evolves endogenously over the time horizon. Another potential intertemporal state variable that could be tackled with this modeling approach would be emissions budgets, i.e. achieving a cumulative emissions budget with intertemporally optimal abatement choices.

While it is convenient to only have a single solve and no other solution algorithm required, the formulation of the model in present value (i.e. discounted) prices may cause numerical scaling problems, since these prices tend to be very small at the end of the model horizon while quantities tend to be very

---

<sup>5</sup> See also [http://ipetsmodel.com/index.php?title=Solution\\_approach\\_and\\_computational\\_procedure](http://ipetsmodel.com/index.php?title=Solution_approach_and_computational_procedure)

large. In addition, solving for all prices and quantities in all time periods increases the computational burden, although the number of variables can be reduced by not solving the model at annual time steps but rather by making the length of a time step a choice of the modeler (see also 2.2.5).

In the fortran model, as each time step is solved individually, there is a numeraire in each time. In the GAMS model which formulates the entire problem as a single problem, only one numeraire is present in the model.

## 2.2.2 Capital dynamics

While computationally challenging, the treatment of capital dynamics in the GAMS version can be greatly simplified in comparison with the fortran version, both in terms of the algebra required as well as the lines of code that handle capital dynamics.

Capital dynamics and intertemporally consistent forward looking behavior are implemented in GAMS/MPSGE as discussed in Paltsev (2004).<sup>6</sup> Because we use the GAMS/MPSGE framework, we have to bring the capital dynamics into the form of production activities which can then be represented easily in GAMS/MPSGE to generate zero profit and market clearance conditions (see section 2.2.1).

Specifically, we define an investment good activity and a capital preserving activity in GAMS/MPSGE. The respective market clearance and zero profit conditions then characterize optimal consumption/investment decisions by agents. Hence, no specific algorithm is needed to inform consumption-investment trade-offs in any given period, as all future returns to investment are already integrated into the problem.

The investment good activity governs the production of investment, or new capital to be added to the capital stock. This activity's zero profit condition ensures that the production cost of one unit of the investment good is equal to the price of capital that it generates. The capital preserving (accounting) activity keeps track of the capital stock by taking current available capital as an input and producing capital that can be rented by producers in this period and future capital (subject to depreciation). The capital preserving (accounting) activity's zero profit condition ensures that the price of capital reflects all future rental returns (subject to depreciation). For practical implementation, two (related) prices for capital are used in the model. The price of a unit of capital  $PK$  reflects the cost to produce this unit of capital. The rental rate  $RK$  is the price that a producer pays to the owner of the capital in order to use it for one period.

In addition, there are market clearing conditions that determine the rental price of capital  $RK(r, t)$ , the unit price of capital  $PK(r, t)$ , and the price of investment  $PI(r, t)$ . The market clearing condition of the investment good activity ensures that the investment good increases the capital stock (as long as  $PI(r, t)$  is positive, otherwise there will be no investment activity). The market clearing condition of the capital preserving activity ensures that the entire capital stock is rented out to industry (as long as  $RK(r, t)$  is positive), and that no capital is "lost" after accounting for depreciation when going from one period to the next (as long as  $PK(r, t)$  is positive).

---

<sup>6</sup> The fundamental dynamic structure as an intertemporal MCP is developed in Lau, Pahlke, & Rutherford, 2002.

### 2.2.3 Utility function

The PET model is designed to reflect the effects of changes in the projected composition of the population over time according to various demographic characteristics. The utility function in the PET model takes into account *per capita* utility as well as preference changes over time. As it is non-standard for CGE models (the most common CES utility function used in CGE models aggregates overall *total* consumption), we will first describe how this function was implemented within the GAMS/MPSGE framework. As the experiments in sections 3.2 and 3.3 will show, GAMS and fortran model behave identically.

The PET model uses a form in which *per capita* consumption enters the intertemporal utility function and total population  $n_t$  is included as follows

$$U = \sum_t \beta^t n_t \frac{\bar{c}_t^{\rho^U}}{\rho^U},$$

with  $\beta^t$  representing the discount factor at time  $t$ ,  $\bar{c}_t$  per capita consumption of an aggregate consumption good, and  $\rho^U$  the intertemporal substitution parameter.

If population is constant over time, this formulation is equal to the total consumption form as in this case  $n_t = n_0$ , which can be taken outside of the summation. For the GAMS/MPSGE version, translating this utility function is not straightforward because it has to fit into the structure of a “standard” CES function and total consumption has to be used as an input in the consumption function. This is achieved by modifying the equation above and expressing population relative to base year population  $n_0$

$$U = \sum_t \beta^t \frac{n_t n_0 \bar{c}_t^{\rho^U}}{n_0 \rho^U}.$$

This normalizes population to  $n_0$ , as  $n_0/\rho^U$  can be taken out of the utility maximization without changing the results. In MPSGE this is achieved by adding the term  $\text{pop}(r,t)/\text{pop}(r,"2000")$  with a base year 2000<sup>7</sup>, into the function description for the aggregation of intertemporal consumption:

```
$prod:W(r) s:tsig
    o:PW(r) q:(sum(t,vc(r)*qref(t)*pref(t)))
    i:P(r,t) q:(vc(r)*qref(t)*pop(r,t)/sum(tf,pop(r,tf)) p:(pref(t))
```

The formulation can thus be interpreted as adjustment parameters that shift the weights of the individual years in the intertemporal utility function towards years with higher population.

The implementation of changes in preference parameters for consumption within a time period is more difficult in the GAMS version because the utility function in MPSGE is generated from the calibrated share form, which has different parameters than the “standard” formulation of the CES function. The preference parameters used in the fortran model, therefore cannot directly be entered into the GAMS

---

<sup>7</sup> To be flexible in the code with respect to the base year, we use  $\text{sum}(tf,\text{pop}(r,tf))$ , where  $tf$  is a set including only the first year of the model horizon. The summation over this one element set thus returns the value of the base year; it is required to pass the GAMS syntax checks.

version. Instead, we use changes in preferences from in expenditure shares relative to the base year. Under a Cobb Douglas utility function, the exogenous expenditure shares will always be the same as in the model solution. For a CES utility function as used in the fortran model to implement preference changes this is generally not the case.

In MPSGE, the adjustment parameter  $\text{mucpr}(cg, r, t)$  in the CES consumption function is added as follows:

```

$Prod:C(r,t) s:csig
o:P(r,t) q:(vc(r))
i:PJ(cg,r,t) q:(vfd(cg,r)*mucpr(cg,r,t)) p:(pref(t))

```

Initially,  $\text{mucpr}$  is set to unity to have constant preferences for the for steady state growth. Starting from there, the parameter is adjusted by

$$\text{mucpr}(cg, r, t) = \text{tgtexpsh}(r, t, cg) / \text{expsh}(r, t, cg);$$

where  $\text{expsh}$  is the share of expenditure within the consumption goods observed in the base year data (and maintained in the steady state growth path) and  $\text{tgtexpsh}$  is the target expenditure share that would follow differences in preferences under unchanged prices. Note that endogenous price changes in response to the shock (or any other shock) might lead to expenditure shares in the model different from those target shares depending on the elasticity of substitution  $\text{csig}$  between consumption goods.

## 2.2.4 Terminal condition

The GAMS version uses an implementation of terminal conditions to approximate an infinite horizon which does not require the specification of an ex ante growth rate in the final period  $T$  (Lau, Pahlke, & Rutherford, 2002). The condition imposed in the model requires that investment grows at the same rate as consumption growth:

$$\frac{I_{r,T}}{I_{r,T-1}} = \frac{C_{r,T}}{C_{r,T-1}}$$

The terminal condition in the fortran model varies from this condition as it explicitly specifies the required demand for the terminal capital stock. However, this demand is adjusted iteratively for each shoot during the solution iteration and hence constitutes a reasonable approximation.<sup>8</sup> Both conditions yield the same result in "steady state runs" where the growth rate of consumption and investment are equal for the entire time horizon by definition.

From the comparison exercises in section 3, we can conclude that the differences in terminal conditions do not play a role for comparing results in a relevant time frame. Specific to our analysis, this did not affect any of the comparisons between versions for model outcomes prior to 2100. In the years to about 2120 (with the exact year slightly varying for different model versions), differences in some cases become significant but are not relevant to the targeted time horizon for analyses.

---

<sup>8</sup> See for [http://ipetsmodel.com/index.php?title=Solution\\_approach\\_and\\_computational\\_procedure](http://ipetsmodel.com/index.php?title=Solution_approach_and_computational_procedure) more details on the implementation in the fortran model.

### 2.2.5 Variable temporal resolution

The fortran version of the PET model runs with 1 year time steps. When the fortran version is solved in parallel (Melnikov et al., 2016; Gruzdev et al., submitted), the number of years to solve are of lesser importance, because the intertemporal problem can be separated from the static equilibrium problem for each model year. In the GAMS formulation, this separation is not possible. Therefore, the computational burden in the GAMS version can be drastically reduced by reducing the temporal dimension of the model, i.e. not explicitly simulating every year, but only every 5 or 10 years. However, the reduced temporal dimension is only an approximation of the investment behavior in which capital becomes available the year after it is created. In this approximation for 5- or 10-year time steps, part of the investment matures immediately and directly adds to the capital stock. In section 3.7, we explore how this approximation affects the results of model runs. In section 3.8, we compare the computational differences between the GAMS and fortran versions, taking into account the option to parallelize the fortran version and to reduce the time dimension of the GAMS model.

## 3 Establishing comparability with the fortran version

In this section, we discuss comparability between the GAMS and fortran versions for key features of the PET model. As mentioned above, we strive for models with an economic structure as similar as possible while maintaining the option to implement additional features in either version. As there are structural differences between the model version (section 2.1), we would expect the default model to differ in response to an economic shock. We therefore need to find adjustments to differences in the treatment of intertemporal trade optimization as well as differences in energy and land using sectors.

Since the GAMS version does not include land use, while Ren et al (2016) does, we use an earlier set of input data that does not disaggregate agricultural (land-using) sectors, from either O'Neill et al. (2010)<sup>9</sup> or O'Neill et al. (2012), and adjust the fortran model accordingly. Since zero carbon electricity is not subject to specific treatment in the GAMS version, we exclude this from the comparison. With regard to differences in the intertemporal treatment of trade, the comparison is done in one of two ways.<sup>10</sup> For most comparisons, where trade is not important (e.g. changes in preferences or population), an input data set without trade can be used, in which case trade is excluded from both models. To implement this, we modified the O'Neill et al. (2010) input data to exclude trade between the regions in the model. This avoids confounding effects and as a side effect achieves faster solution times by solving a smaller equilibrium problem in each region rather than one integrated equilibrium problem across regions. Since the regions and thus their prices are completely separate, it requires setting a numeraire in each region. To compare responses in trade to shocks in the model versions a more sophisticated strategy is required. Specifically, we add an additional constraint on the balance of trade through endogenous

---

<sup>9</sup> This version of data allowed for easier removal of international trade and was hence used in model runs without international trade. Note that this model also has a different energy structure by combining oil and gas sectors and an earlier start year (2000).

<sup>10</sup> In principle, the GAMS version could have been built to solve the model in a structure that looped over a recursive model with adjustments to the consumption-savings trade off just as in the fortran version. However, we felt that the current state of implementing forward-looking models requires solving the model at a single shot. The models thus differ in this central aspect which affects model behavior. The GAMS version was therefore adjusted by adding additional constraints to the model for this comparison.

tariffs, which is (almost) equivalent to implementing a budget balance as in the fortran version. The details are described in section 3.6.

For this comparison, we use version 2.0 of the fortran model code (as used in Ren et al., 2016, however with fixes of minor errors). However, we use a different method of initial steady state calibration from Ren et al. (2016), in particular a different method to reconcile base year data on rental payments to capital and investment with (exogenous) parameters of time preference and depreciation rates. Here, we use a tax on rental capital to achieve this reconciliation, as described in Balistreri (1997). The GAMS model is used with annual time steps in this exercise and usually run from 2004 (or 2000) to the year 2124 (or 2120, respectively).

The comparison consisted of several steps and roughly follows the order of a typical calibration of the GAMS version. First, we implement a steady state growth path (with or without a steady state growth larger than zero) to confirm that the steady state growth path is identical in the two models (section 3.1).<sup>11</sup> Then, the various components of comparison were carried out by shocking the model with changes in population size (section 3.2), preferences (section 3.3), and labor productivity (section 3.4). We also tested shocks to the initial capital stocks in both models (section 3.5), as this is one of the options to reconcile initial investment/GDP ratios (section 4.5). The different sections below list the experiments carried out and the results. While results often were not identical due to the numerics, they proved to be close enough to conclude that the models behave the same (i.e. differences of <0.1%).

In order to increase harmonization between the models and locate initial differences the following technique proved to be helpful. Starting from the same steady state pathway in both models, a shock was applied to both models. The price and quantity variables from the fortran model were then exported to a file and imported into GAMS. Re-running the GAMS model with the `iterlim=0` option (i.e. only perform an evaluation of all model equations with the variable values of the fortran model without an actual optimization) then points directly to equations that include differences. In other words, this method plugs in the values of a fortran solution into the GAMS equations to see which ones are violated. This method therefore allows a stepwise check for quantities (of specific or all sectors), prices and other variables in the model (e.g. transfers to the government).<sup>12</sup>

---

<sup>11</sup> This step was also crucial to fix a mistake in the trade code in the fortran model, as old versions had small deviations already in the steady state growth path.

<sup>12</sup> This required some modifications to the GAMS version to make some variables comparable to the fortran version of the model. In particular, the scaling of the rental price for capital and the associated quantity had to be changed in GAMS, since the variable is defined as the time preference plus the depreciation rate in fortran, whereas it is normalized to unity in the GAMS version. In addition, the investment price is a variable in fortran, but not in the standard GAMS version. In GAMS, the investment price cancels out as two conditions are merged into one single production block: (1) zero profit in the production of the investment good (the sum of cost of inputs to produce one unit of the investment good has to equal the price of investment) and (2) the zero profit condition in the investment allocation to the capital stock (the price of investment has to equal all future rental returns to this investment). For the comparison, these two parts of investment production and allocation were split into separate production blocks.

### 3.1 Steady State growth path

For an initial set-up, we construct a balanced steady state growth path. A tax on rental capital is implemented to reconcile observations of rental payments to capital with investment levels, given parameter choices of time preferences and the steady state growth rate (Balistreri, 1997). In the Fortran version, the capital taxes are levied on consumers, rather than producers as in the GAMS version. This difference results in different capital tax *rates* but the same market outcome (payment volumes). The capital tax rate calculation is added in the fortran model<sup>13</sup> for this comparison exercise and will be used as default feature for the model in the future.

For the steady state runs without supplementary shocks, identical absolute prices in both models were obtained. In the fortran model, each time step is solved individually, hence there is a numeraire in each time step and hence absolute price levels differ between the models (as prices are discounted in GAMS but not in fortran). In practice, we followed an approach that left price levels in the fortran model unchanged inside the model (with a numeraire for each time step) and post-processed the fortran prices by multiplying with the discount factor.<sup>14</sup>

In all further experiments below this convenient property does not hold. Once a shock is added to the model, it is difficult to convert prices obtained from the GAMS model solution into prices equivalent to those from the fortran version.<sup>15</sup> A shock can change the relative contribution of different time periods to the overall price index used in the GAMS model, and hence absolute prices in each time period that are equivalent to those in the fortran version can no longer be easily calculated by multiplying with a discount factor. Therefore, relative prices *within* each time step were compared instead, as these are consistent in both model versions.

Using relative rather than absolute prices, the model versions produce identical outcomes to a number of shocks starting from steady state baselines with various growth rates (tested with 0% or 2%).

### 3.2 Changes in population size

The PET model is designed to reflect the effects of changes in the projected size and composition of the population over time according to various demographic characteristics. In this section, we analyze changes in population size. In the next section, experiments are run that deal with preference changes which can be induced by changes in the demographic composition. Population size enters the utility function which is represented in per capita terms (see section 2.2.3) as well as (effective) labor supply. For this experiment, we isolated the effect that population size has on the economy through the

---

<sup>13</sup> Specifically, this is added to the CalPet part of the fortran model code, see [http://ipetsmodel.com/index.php?title=Obtaining\\_and\\_running\\_the\\_model](http://ipetsmodel.com/index.php?title=Obtaining_and_running_the_model).

<sup>14</sup> All prices in the model depend only on the time preference  $\rho$  and the time step  $t$ ,  $p_t = 1/(1+\rho)^t$ .

<sup>15</sup> In the version without trade, the labor price of each region is set to unity in each time step in the fortran model. In GAMS, the labor price in the first simulation year is set to unity and decays over time. As in the GAMS model all model periods are explicitly linked via a single income balance constraint, relative prices exist not only within one simulation year but also between simulation years. For simple cases (e.g. single region or single good models), the price conversion factors can be explicitly derived, however, it was not possible to derive a formula algebraically for a multi-sector, multi-region case.

representation of utility in per capita terms; changes in labor supply were analyzed separately (see section 3.4).

In our population experiments, we shocked both the fortran and the GAMS model versions that were calibrated to a steady state growth with several population trajectories. We performed the following experiments related to the population size:

- Constant base year population size in all regions
- A 10% change in a single region (China) for a single year (2020)
- Calibration of a single region (China) to the population trajectory in SSP2
- Calibration in all regions to the population trajectory in SSP2

In all cases, we could confirm that the model versions behaved identically.

### 3.3 Preference changes

Changes in the composition of population as well as in per capita income lead to changes in preferences for different consumption goods. Section 2.2.3 described how the consumption function in GAMS/MPSGE was implemented to account for these changes in preferences. To check whether changes in the preference parameters had identical effects on model behavior, we ran several experiments:

- Constant preferences (i.e. preferences as in the base year)
- Preference changes in line with SSP2 due to changes in population composition and income (as in Ren et al., 2018) in a single region (China)
- Preference changes in line with SSP2 (as above) in all region

In all cases, we observed identical model outcomes. As both population and preference changes enter the utility function, we ensured that also joint shocks to population and preferences were reproduced in both model versions. In particular, we combined the most complex shocks (i.e. the last item in the respective lists) from sections 3.2 and 3.3. In addition, we also ensured that the results hold under additional shock from labor productivity (section 3.4)

### 3.4 Labor productivity changes

It is important to establish equal model behavior in response to changes in labor productivity because this is the adjustment made to calibrate to an exogenous GDP pathway in the PET model. In addition, changes in labor supply are also a result of demographic changes. Once potential differences in trade were accounted for (see below), changes in labor productivity induced the same change in GDP in both model versions. This was established by running various scenarios with and without trade, starting from steady state pathways and adding labor productivity shocks:

- An 10% increase of labor productivity in a single region (China) a single year (2020)
- Introducing additional labor productivity growth of 1% in a single region (China) or all regions
- Replacing the labor supply from growing at a steady state growth rate with labor supply resulting from an SSP2 scenario (as in Ren et al., 2018) in a single region (China) or all regions



In all cases, we could confirm that the model versions behaved identically.

### 3.5 Capital dynamics

To confirm that capital dynamics in both model versions are the same and that both models behave identically to an exogenous shock to (initial) capital endowments, we increased or decreased the capital endowment that was prescribed for a dynamic steady state run. This is important, because the adjustment of initial capital can serve as an option to reconcile observed investment to GDP ratios after calibration (see section 4.5). As the capital stock of an economy is observed only with a large uncertainty, this is a reasonable way to bring another model outcome towards observations (investment to GDP ratios are observed with less uncertainty).

The experiment to confirm confirmed identical responses to adjustments in initial capital endowment reduced or increased capital endowment in a single region (China) or all regions by 10%. We could observe that all price and quantity variables responded identically in the GAMS and fortran versions, including an identical adjustment of the investment rate in the initial year.

### 3.6 Trade

Starting from a steady state growth of 2% (with international trade between all model regions), additional labor productivity increases of 1% per year occur in China for three decades.<sup>16</sup> Figure 1 shows simple example runs to demonstrate the role of different trade treatments. The figure analyzes relative prices within regions (panel (a)), quantities (panel (b)), and relative prices between regions (panel (c)).

In panel (a), relative prices within regions, the price levels of composite consumption (which also serves as an indicator of the overall price level) in China and the USA are expressed as a ratio relative to the labor price level of the respective region, i.e.  $P/PL$ . It can be seen that the unconstrained GAMS model (labeled “nofix”) deviates quite substantially from the fortran model’s outcome and that the constrained GAMS model (labeled “bopfix”) resembles the fortran version much better. In panel (b), quantity changes in the aggregate consumption bundle are shown. Again, once the GAMS model is constrained (“bopfix”), it matches the fortran version well. In panel (c), relative prices between regions are shown. The left sub-panel shows the ratio between consumption prices (or overall price levels) between the US and China, and the right sub-panel shows relative labor prices after the shock. Here, the GAMS prices in “bopfix” do not correspond to the ones in the fortran version, but this difference is removed when the correction factor described in section 2.1.1 is applied (“bopfix-corr”), to account for the price differences between regions that arise from the tariff/subsidy resulting from the balance of trade constraint.

From Figure 1, it can be seen that without the additional “bopfix” constraint, an immediate adjustment reaction takes place in the year 2000 in the US (i.e. the region not directly affected by the shock) and that the real effect in China is dampened over time (see panel (b)). This is driven by consumption smoothing in China, i.e. a reduction in the intertemporal changes of consumption. As this leads to increased demand in early years from additional consumption, other regions such as the US are willing to react to this by reducing their consumption and increasing exports to China, although the change is

---

<sup>16</sup> The productivity increases are faded out between 2020 and 2040.

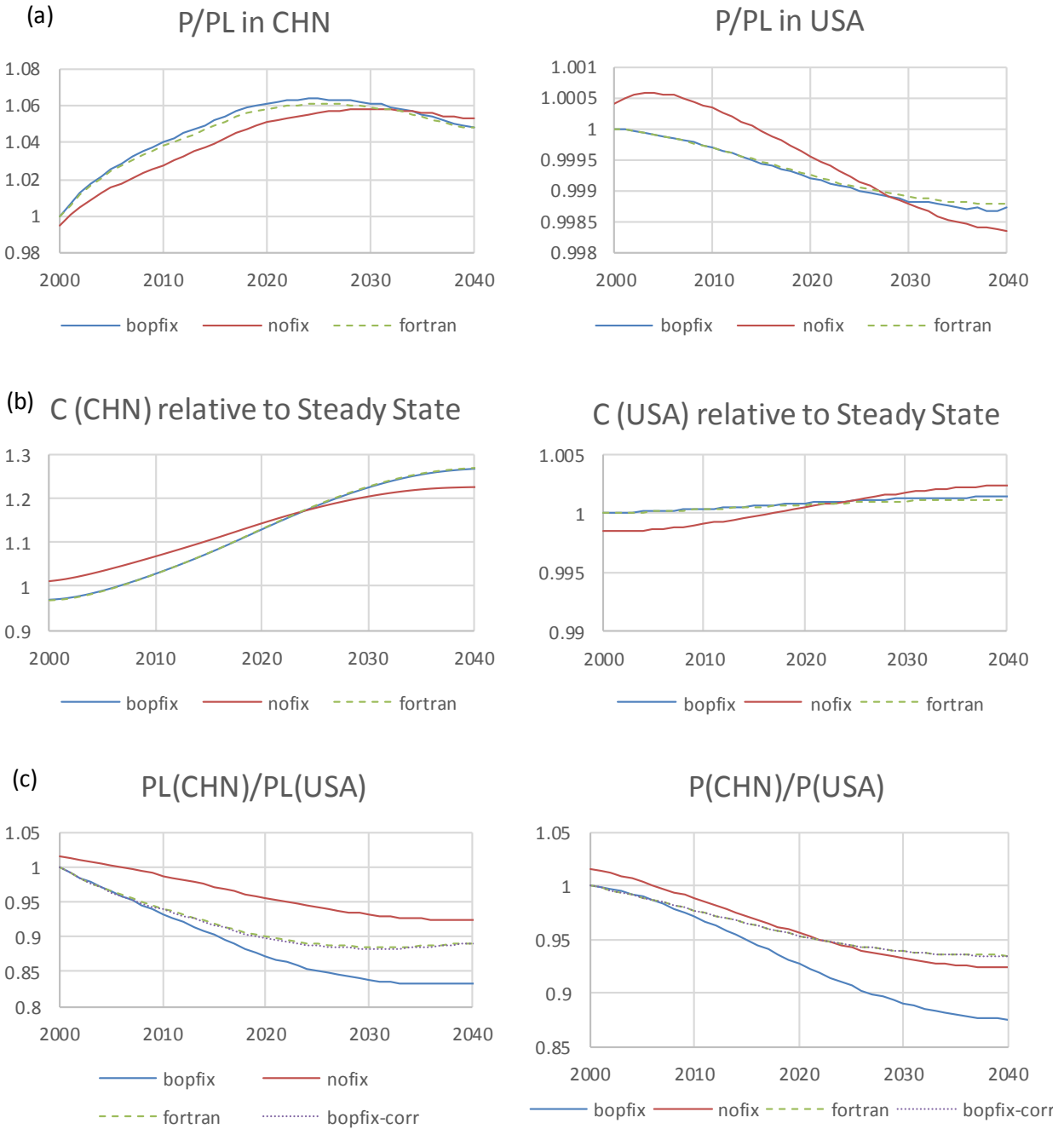


Figure 1. Responses under different trade settings. (a) Relative prices within China and USA, (b) changes in aggregate consumption relative to the steady state, (c) relative between China and USA.

much smaller than the one observed in China. It can also be seen that in this simulation small differences remain between the bopfix GAMS version and the fortran version (in this case, best visible in the left sub-panel of panel (a)). These differences are due to the distorting effect of the endogenous

tariffs, which lead to changes in the terms of trade and thus secondary effects in the countries involved.<sup>17</sup>

### 3.7 Temporal resolution

As described in section 2.2.5, the GAMS model can be solved in different temporal resolutions. A reduced temporal resolution lowers the number of variables and hence the resolution time. While a choice of annual time steps is equivalent to the fortran version of the model, a reduced temporal resolution will lead to differences. This follows from when investment matures and contributes to the capital stock (see section 2.2.5). As a consequence, the tax rate on capital required to reconcile observed investment and returns to capital with assumptions about the growth and discount rates (see section 3.1) depends on the choice of the time step. With data from O’Neill et al. (2010), capital tax rates in the dynamic steady state are 20% for annual time steps, 26.7% for 5-year time steps, and 33.4% for 10-year time steps.

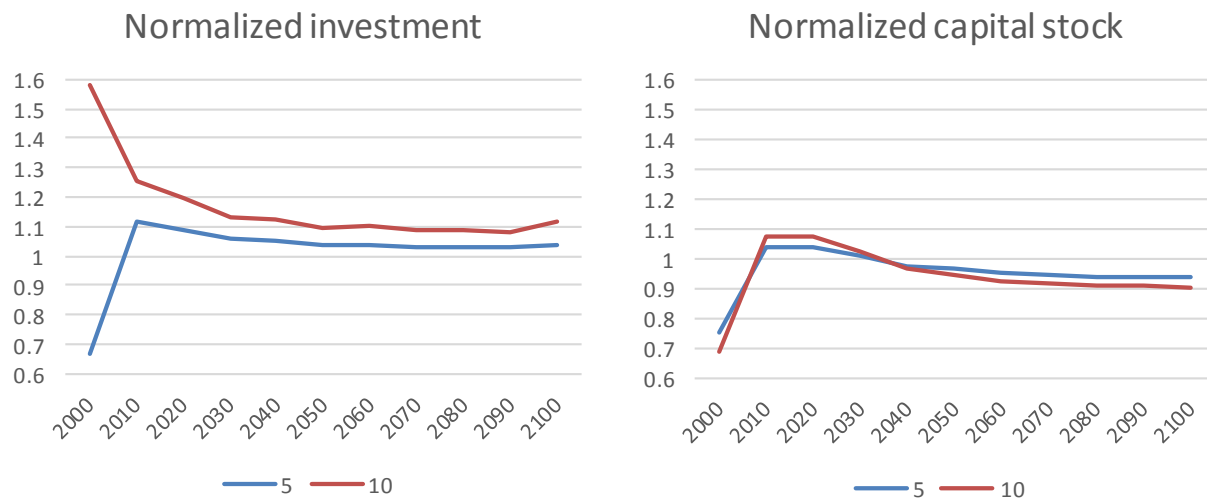


Figure 2. Investment (left panel) and Capital stock (right panel) in China in the 5 and 10 year runs relative to levels in model run with annual time steps.

To test the effect on model outcomes of the choice of time resolution, we run the following experiment with different time steps (1 year, 5 years, 10 years). We calibrate preference changes, adjust GDP to reach the SSP2 pathway, and free up the capital stock (we omit the energy calibration in this experiment because it has fewer dynamic implications for capital stock, the main variable affected by the choice of time steps). While the GDP outcome from the calibration is the same across versions with different time steps, some differences exist for the level of investment activity and the capital stock. Figure 2 shows the relative difference in capital stock and investment for the 5 year and 10 year time step runs compared to a model run with annual resolution. A value of 1 implies that the value exactly corresponds to the value observed in the run with annual time steps. There are several observations to make. First, there is a strong initial difference in the first simulation year (2000), but differences decline over time.

<sup>17</sup> These effects are more prominent when a region has a net trade surplus or deficit over the model horizon, as in this case the endogenous tariffs are inducing rents.

Part of the reason for this initial difference is that the capital stock in the base year is fixed in the annual run but adjusts in the runs with lower temporal resolution, because part of the investment in the first year directly affects the capital stock. Also, note that the 5 year time step run generally is closer to the annual time step run than the run with 10 year time steps. The sign of the deviation differs between the 5- and 10-year resolution runs only for the initial investment. The changes caused by capital dynamics are also reflected in patterns of consumption and production (not shown). As GDP is identical across all runs, consumption is a mirror image of investment. Production in different sectors differs depending on how intensively it is used as input to consumption or investment, respectively.

Next, we add a shock to the baseline, as in most analyses we are primarily interested in the differences between a baseline and a counterfactual scenario. In this case, the shock implemented is rather large, a 25% tax of all non-factor (i.e., material and energy) inputs in China. Figure 3 shows the difference between the scenario with the shock and the baseline. Again, initial differences are strong, but diminish over time. As the capital stock in the base year is fixed in the annual time steps run, no change can occur here. Changes in consumption are also reflected in intertemporal welfare changes, shown in Figure 4. In general, the results are quite comparable between different runs with different temporal resolution. For all regions, the welfare impact is larger under a higher temporal resolution, probably due to the fact that the initial capital stock cannot be adjusted as much as in the approximation with lower temporal resolution.

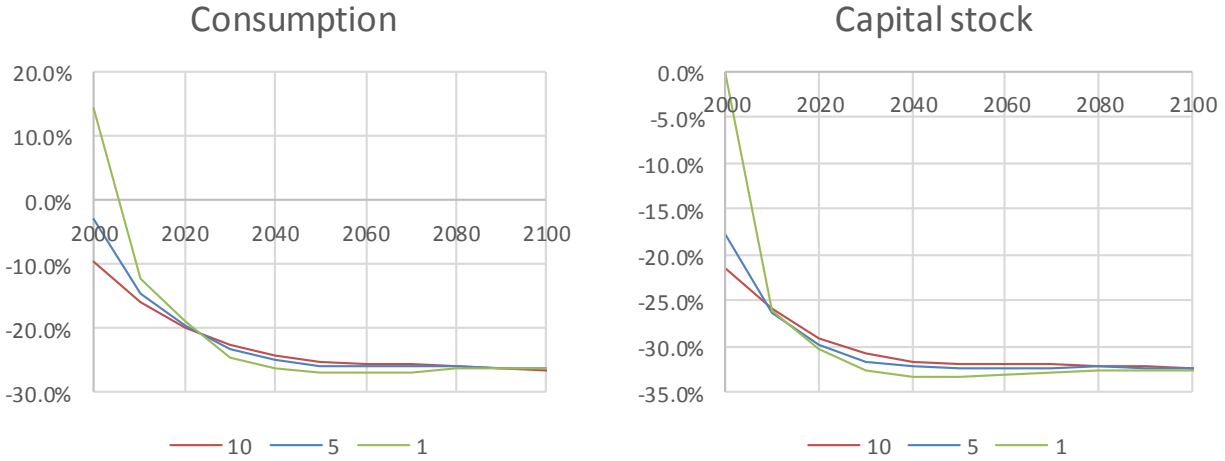


Figure 3. Consumption (left panel) and capital stock (right panel) relative to baseline after a tax shock.

In general, consumption and welfare results point to similar results, especially in the longer run that is of interest to long term scenarios (for shorter term simulations, higher time steps might be advisable to capture a better initial investment/consumption response). The comparison of welfare results also showed that the ordering is consistent and the differences in the magnitude lie within other uncertainties of the model. For most applications focusing on the long term, a temporal resolution of 5 or even 10 years can therefore be justified.

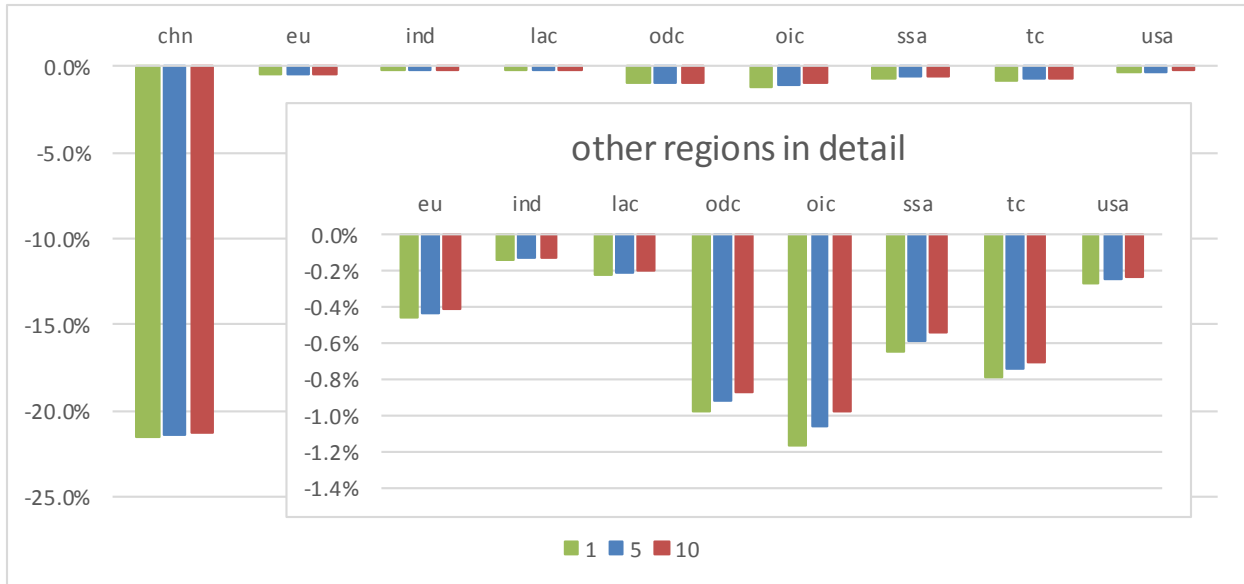


Figure 4. Welfare change in different regions due to a tax shock.

### 3.8 Computational aspects

The solution speed of the fortran version has been improved recently by several factors. This includes using the Nitsol solver (Pernice & Walker, 1998) which improves the solution speed compared to the previously used, simpler Newton solver by up to 90%. The fortran model also has been parallelized which is feasible due to the design of the solution algorithm, which splits the problem into an intertemporal and static problems (Melnikov et al., 2016; Gruzdev et al., submitted).<sup>18</sup> Further, the model initialization was improved by loading variable values from a previous model solve which increases the ability to find a solution as well as time to reach the solution, especially if the difference to a previous solve is only a relatively minor perturbation. This last feature is similar to the GAMS model, where subsequent model solves are initialized with the variable values from a previous solve by default.

The closest comparison of the runtime needed by the GAMS and fortran versions to solve the model consists of using a single core model run. Starting with a solution from a previous run, two cases are assessed. The first case starts with a (relatively small) preference shock on a prior labor productivity shock on the dynamic steady state (Figure 5, top panel). The GAMS version is run as a default version, as well as a version that imposes a specific balance of trade in each period (bopfix=yes) and a version that does not include trade (and therefore solves nine individual regions)<sup>19</sup>. Results show that adding the additional trade constraint increases the complexity of the model and therefore leads to a longer solve time. The second case reverses the order of the shocks, i.e. puts a larger shock on top of a smaller shock (bottom panel). This increases the overall solution time for all runs, but does not change the general picture. In both cases, the fortran model (using the Nitsol solver) took about 6 times as long to find a solution as the default GAMS model. In this comparison, options to help the solver were passed to the

<sup>18</sup> See also [http://ipetsmodel.com/index.php?title=Solution\\_approach\\_and\\_computational\\_procedure](http://ipetsmodel.com/index.php?title=Solution_approach_and_computational_procedure) for the solution approach and details on the parallelization.

<sup>19</sup> The fortran version was also run without trade, but runtime was very similar to the version with trade or even exceeded it in some cases.

PATH solver. In calibrating the model, these options are often helpful to find a solution and are relatively consistent when applying similar shocks (e.g. changes in labor productivity during the GDP calibration) and can be obtained from the log output of successful or failed solves.

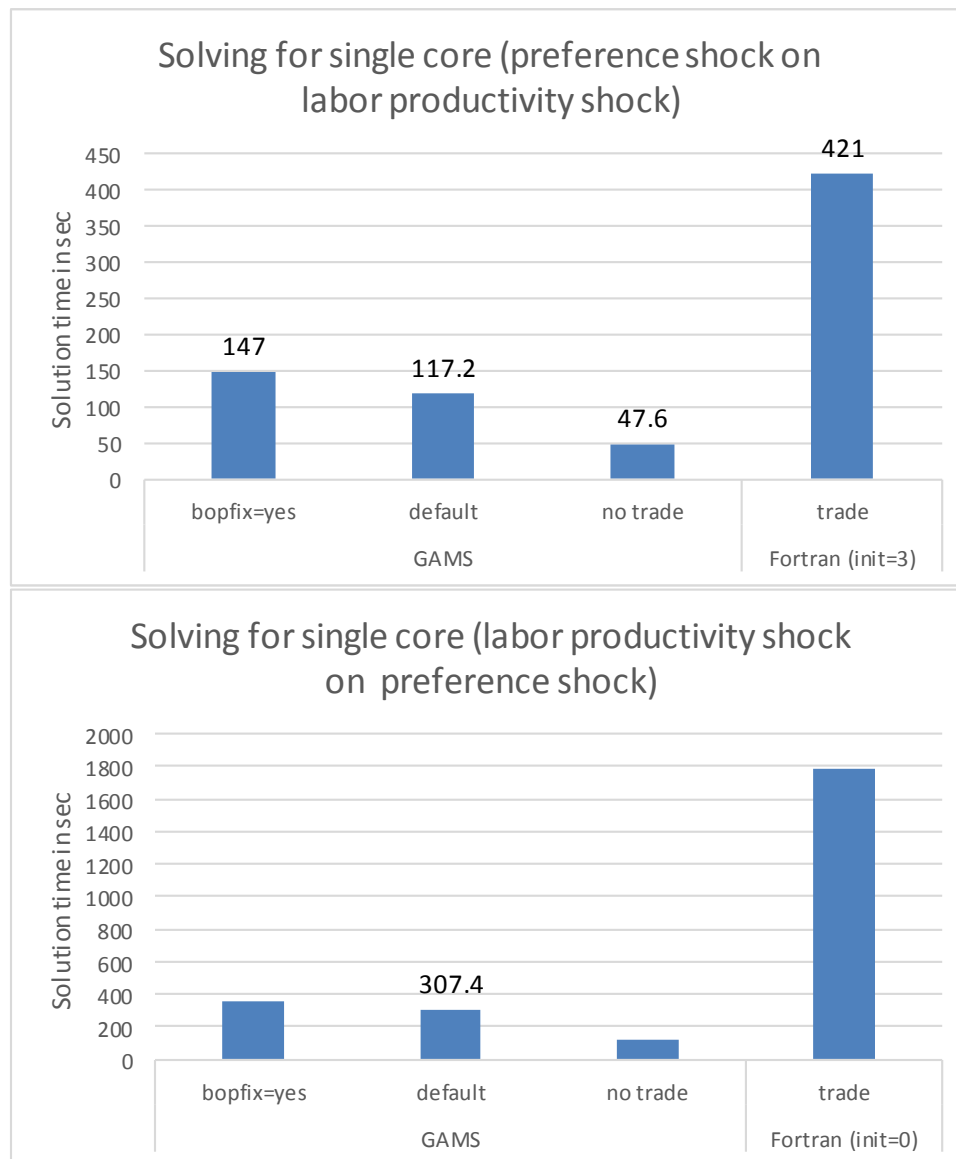


Figure 5. Comparison of solution time in seconds on NCAR’s CGD server “burnt” (Intel(R) Xeon(R) CPU E5-4640 v4 processors with 2.10GHz). The top panel shows the solution time after imposing a preference shock on the previous solution which was based on a labor productivity shock on the dynamic steady state. The bottom panel reverses the order of the shocks. Note that in the bottom panel the fortran model performed better when not starting from the previous solution, but with an initialization of the capital dynamics based on changes in the labor productivity growth (initialization setting  $init=0$ ).

As noted above, the fortran version can be solved with parallel processors and time to find a solution can be reduced substantially. The model scales quite well, i.e. the solution time can be decreased efficiently by adding additional cores (Melnikov, Gruzdev, Dalton, & O'Neill, 2016). In order to achieve the same solution time as the GAMS model, about 6-8 cores would be required.

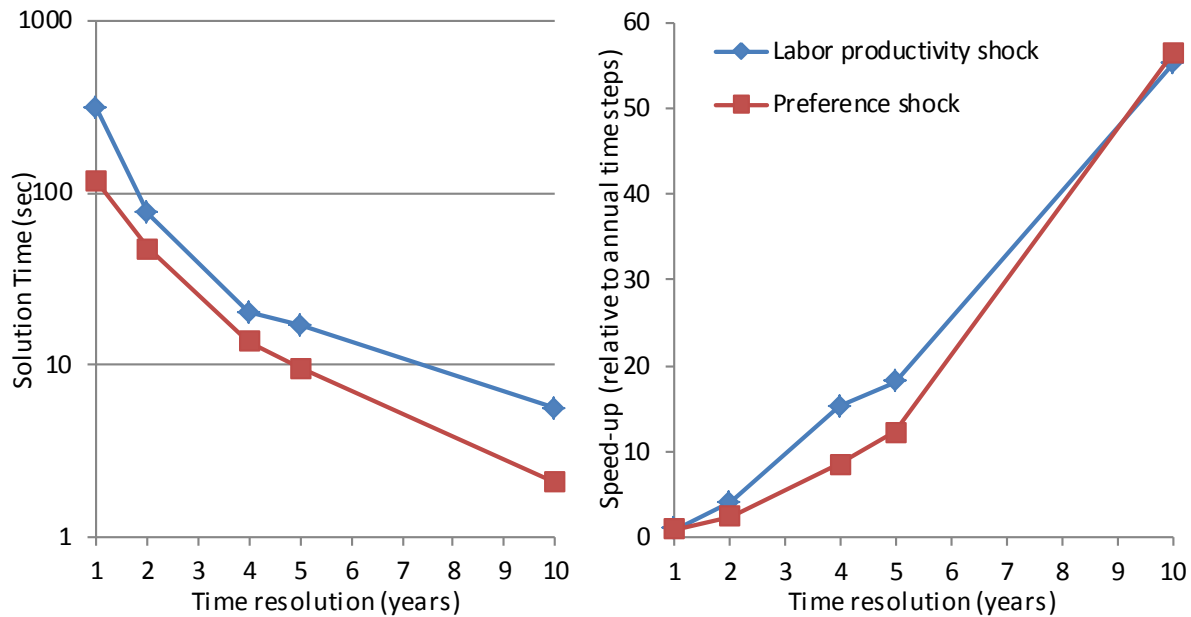


Figure 6. Solving time and speed-up by reducing the number of time steps explicitly modeled (speed-up is relative to the solution time with annual time steps).

As discussed above, the GAMS version formulates the dynamic and static problems as single problem to solve and hence cannot be parallelized in the same way as the fortran model. It therefore forgoes the ability to split the problem into smaller, parallelizable problems.<sup>20</sup> However, the current GAMS formulation has the advantage that it allows for treating more than one state variable.

In the GAMS version, it is possible to speed up the solution time by limiting the number of time steps to explicitly solve for, as described above. Figure 6 shows the solution time and the speed-up that can be gained from reducing the number of time steps. For development and debugging purposes, the 10 year time steps solve the model in less than 5 seconds and offer more than a 50 fold increase in speed, and 5 year time steps which are usually sufficient for work on long term scenarios solve the model in less than half a minute and offer a speed-up by a factor of 10-20 compared to annual time steps at only small loss of precision.

<sup>20</sup> It would in principle be possible to use MPSGE problem for the static problems only and solve the intertemporal problem in the same way as the fortran model. This would allow for parallelization the same way as done in the fortran model.

## 4 Differences in calibration to a baseline

Independent of similarities in model structure described above is the strategy to calibrate to a baseline scenario that serves as a starting point for economic analysis. The baseline is a plausible representation of the future and hence differs from the stylized steady state path that was the starting point for the experiments performed to demonstrate equivalence between both model versions. This is done by adjusting parameters, productivities or preferences.

While both model versions generally use the same parameters to reach a given calibration target, the actual implementation is different between versions. This is due to certain approaches to calibration being easier in either the fortran or GAMS versions. For example, both models adjust labor productivity to target exogenous GDP values. However, the algorithm that formalizes the process of updating labor productivity differs (section 4.1). As shocks to labor productivity lead to identical model outcomes (as we established in section 3.4), including GDP, it does not matter what specific rule is chosen to update labor productivity.

For other calibration targets, most notably energy quantities (section 4.2), productivity parameters are adjusted in both model versions to achieve the same energy targets. In this case, however, the default approach for the GAMS version differs from the fortran model, as different productivity parameters are adjusted. In principle, the method applied in the fortran model could be used, but we found the default implementation in the GAMS model to be quite robust. In addition, implementing the fortran model approach in GAMS might not be straightforward, as the model parameters are not exactly the same in both models but are related through algebraic manipulation of the underlying CES production functions. This is due to the GAMS model being written in MPSGE (which itself is a simplification of the CES functions written in calibrated share form; see Böhringer et al. (2003) for details), while fortran adheres to the more “conventional” functional form of the CES functions. Using different parameters to reach a target leads to different states of the economy despite both reaching the same targets. Hence model response in a counterfactual scenario will be different as well. There exists a model version of the fortran implementation that uses the calibration used in the GAMS version of the model, under which the two baselines will reach the same state to meet the quantity targets.

There are also additional calibration options in the GAMS version that go beyond the calibration options implemented in the fortran version of the model. As these are optional, to achieve identical calibration behavior, they could be switched off. As these options improve the characteristics of the calibrated baseline, it might be beneficial to incorporate these into the fortran model, as well. This already has happened for the adjustment of investment to GDP ratios.

There are some considerations to be made when calibrating to a baseline with the GAMS version of the model. The PATH solver is powerful when the solution is not too far away from initial conditions. We therefore need to specify variable values that are close to a final solution. The calibration to any baseline scenario in the GAMS model therefore starts with a known solution of the multiple-region, multiple-sector model and then adds desired features like region-specific development in terms of GDP and energy growth as well as energy prices. Specifically, the model starts with a static equilibrium based on the (balanced) input data and then adds a steady state growth path. This dynamic equilibrium starts off



with an equal growth rate for all regions. That is, the static model version is projected to initially be unchanged in relative terms; future prices are however discounted as the model is formulated in present value terms. Since the initial balanced growth path of a dynamic steady state can be calculated, it provides a solution that can be passed on to the solver (see section 3.1). This initial model solve also serves the purpose of checking whether the model is formulated correctly, i.e. whether the initial inputs are a solution to the model equations ("replication check").

We can then move the model away from this steady state pathway towards a baseline scenario of interest, which is typically characterized by regionally differentiated growth rates, changes in energy use and consumption patterns over time. We split the adjustments required to move the model solution from its balanced growth path to the baseline scenario into a series of changes small enough to allow the PATH solver to converge quickly to a solution based on a previous solution. Typically, the following modifications are made to the steady state path to obtain a calibrated baseline scenario:

1. Adjust preferences to a utility function maximizing per capita consumption and adjust preferences for changes over time (see section 2.2.3).
2. Calibrate energy prices (see section 4.3).
3. Calibrate GDP (see section 4.1).
4. Calibrate investment to GDP ratios (see sections 4.4 and 4.5).
5. Adjust the terminal condition, i.e. freeing up the fixed capital stock that is used when setting up the steady state run (see section 2.2.4).
6. Calibrate energy use (see section 4.2).
7. Add emissions to the model as additional variables, based on calibrated energy flows and exogenous emission factors.

Other orders to these steps are possible, but this sequence has proven relatively robust. Some calibration steps in this list are optional and can be skipped. The GAMS files that contain the respective calibration steps can be run with a batch or shell script, using the restart option in GAMS.<sup>21</sup>

#### 4.1 GDP growth

While the response to an exogenous input of labor productivity change is identical in the fortran and GAMS versions, the method of calibrating to an exogenous GDP target is not.<sup>22</sup> In the fortran version, a Newton algorithm automatically searches for labor productivity increases that are consistent with GDP values in key years (usually 5-7 time points along the entire model horizon) and interpolates labor productivity in the years in between. In the GAMS version, labor productivity of each year is adjusted directly based on the deviation between the current calculated GDP in the model and the targeted GDP. For example, if the current calculated GDP value is half of the GDP target, then labor productivity in this given year is doubled. In order to avoid overly large changes in the parameters and thus difficulties in solving the model, the size of the shock is capped. The updating formula (time and region indices

---

<sup>21</sup> See [https://www.gams.com/24.8/docs/userguides/userguide/ u g \\_ save \\_ restart.html](https://www.gams.com/24.8/docs/userguides/userguide/ u g _ save _ restart.html)

<sup>22</sup> For details on the calibration in the fortran version, see [http://ipetsmodel.com/index.php?title=Model\\_calibration\\_to\\_a\\_baseline](http://ipetsmodel.com/index.php?title=Model_calibration_to_a_baseline)

omitted) applied to labor productivity  $ladj$  (an adjustment multiplier on the initial guess for labor productivity) to derive an updated guess of  $ladj'$  is as follows

$$ladj' = ladj \left( 1 + \varphi \left( \frac{GDP_{targ}}{GDP_{calc}} - 1 \right) \right).$$

In this equation,  $GDP_{targ}$  is the target to achieve in the calibration,  $GDP_{calc}$  is the GDP calculated from the last model run and  $\varphi$  is a scaling parameter ( $0 < \varphi < 1$ ) to avoid large changes.

## 4.2 Energy consumption

Several methods exist to calibrate to energy targets. In the fortran version, productivity parameters in the CES production function are adjusted by a Newton algorithm until energy targets are met.<sup>23</sup> This approach is depicted in Figure 7 which shows a stylized isocost line (indicating all input combinations with the same cost, with the slope expressing the relative prices of inputs) and the isoquant curves (indicating all production combinations to produce the same quantity) of the original equilibrium (blue star) with energy inputs  $E$  and other inputs  $Oth$ .<sup>24</sup> Now assume that energy use in this equilibrium exceeds the exogenous target. The calibration procedure will seek to reduce the energy input  $E$  in all sectors by making the use of  $E$  more efficient (i.e. to produce one unit of output, fewer  $E$  will be needed); note that input of Other is unchanged. The result (red star) is a shift of the equilibrium solution to a lower isocost curve; i.e. the same output at a lower cost. This approach has repercussions in the rest of the economy, because lower unit costs of energy using translate to increases in competitiveness; *ceteris paribus*, demand for goods produced by energy using sectors will increase. This effect will offset part of the reduction in energy demand induced by the efficiency increase, because increased output will require more inputs (rebound effect). In extreme cases, energy use in the economy could even increase as a response to increased efficiency.<sup>25</sup> Furthermore, the Newton method's automatic search for parameters to achieve all energy targets might lead to questionable results in which a primary sector becomes very efficient (inefficient) and an energy conversion sector very inefficient (efficient) to offset the change that a final energy consumer experiences.

The GAMS model has a different method of calibrating to exogenous energy targets, depicted in Figure 8. As above, energy input  $E$  is reduced. However, in addition, inputs of other goods are increased to offset the reduction in costs due to lower energy input. This induces a shift along the same isocost line. As the good produced in this sector does not change its cost, there will be no change in demand for this good, and repercussions for the rest of the economy are greatly reduced compared to the fortran version's method. There can still be general equilibrium effects caused by reduced demand for energy (and increased demand for other goods), which will change the relative prices of these goods and therefore have further effects on consumption. These effects might lead to the energy target(s) being

---

<sup>23</sup> In particular, total and partial factor productivity parameters of energy goods are adjusted. For details, see [http://ipetsmodel.com/index.php?title=Model\\_calibration\\_to\\_a\\_baseline](http://ipetsmodel.com/index.php?title=Model_calibration_to_a_baseline).

<sup>24</sup> In the model, other inputs refer to all other intermediate and energy inputs as well as value added inputs. To simplify the representation in figures and in the text, we abstract to the use of only two inputs.

<sup>25</sup> Usually this is not the case for the global economy, but is more likely to happen when several energy goods are targeted in many regions.

missed in the first pass, therefore the procedure of adjusting inputs (while holding costs constant) is repeated until the targets are reached with a pre-defined level of precision. Note that the initial isocost line is held constant throughout this process, even though the general equilibrium effects move relative prices and hence the slope of the isocost line. If the isocost line were adjusted, the choice of the step size could then influence the exact outcome.

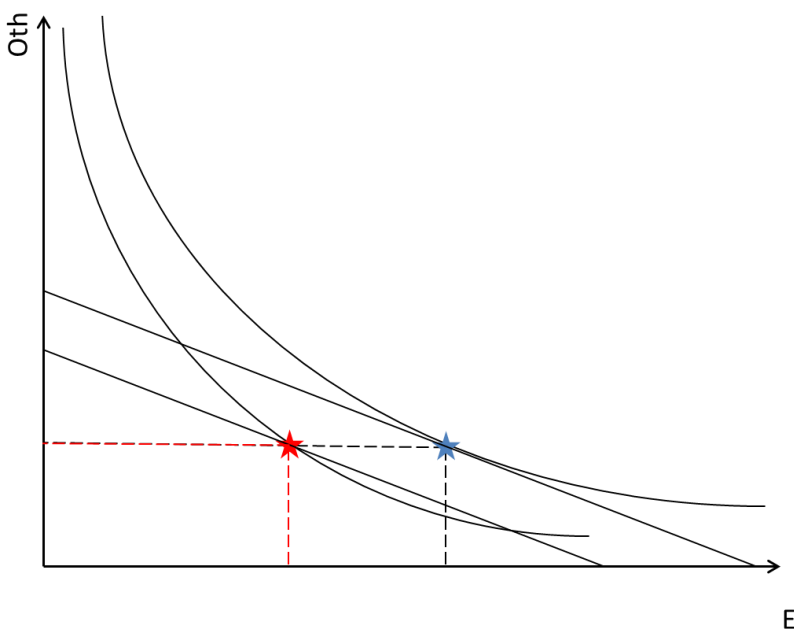


Figure 7. Adjusting factor productivities to reach energy targets in the fortran version of the model. The isocost lines are the linear lines of the equal costs from input combinations of E and Oth. The isoquant curves describe all combinations of inputs that yield the same level of output. In the initial situation, the blue star is the production option that represents cost minimal production. Adjusting energy productivity only leads to the situation depicted with the red star, i.e. a lower isocost curve and a different isoquant.

Using the different calibration approaches will lead to different baselines, despite reaching the same energy targets. This implies that the two versions of the model would then react differently to a policy applied to the two baselines, because they are starting in different states. It depends on the underlying storyline of the baseline scenario to decide which approach is more suitable for calibration. To compare the models and establish that we can reproduce results, we modified the fortran version rather than the GAMS version in this case. Specifically, we added the option to update input shares in a cost-preserving manner in the fortran version. The changes are implemented in the SimPet component of the code.<sup>26</sup> SimPet uses the share parameters as direct inputs. The share parameters are then converted into input shares and the input shares are modified so the production costs are preserved. The modified input shares are then converted back to share parameters to solve the system.

<sup>26</sup> See [http://ipetsmodel.com/index.php?title=Obtaining\\_and\\_running\\_the\\_model](http://ipetsmodel.com/index.php?title=Obtaining_and_running_the_model)

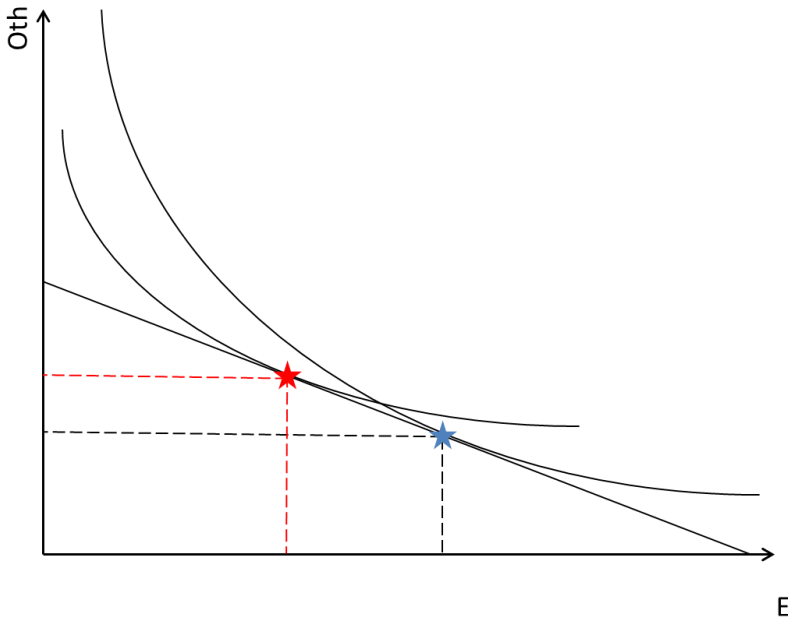


Figure 8. Shifting input shares to reach energy targets in the GAMS version of the model. The isocost lines are the linear lines of the equal costs from input combinations of  $E$  and  $Oth$ . The isoquant curves describe all combinations of inputs that yield the same level of output. In the initial situation, the blue star is the production option that represents cost minimal production. Adjusting productivities leads to a shift along the initial isocost line. The new production point (i.e. a point along the new isoquant curve) is represented by the red star.

An experiment was carried out to test whether the energy calibrations in the two models produced the same results, after this fortran version modification. The experiment was designed to adjust the shares to achieve a specific target in a given energy type, we implemented this for the coal use target in China taken from an SSP2 scenario to demonstrate equivalence between the GAMS and the modified fortran versions. Identical share adjustments were used in both the GAMS and the fortran versions and the same model state characterized by all model quantities and prices was obtained from both models. In future applications of the fortran version of the model, choices will have to be made regarding which method of calibrating to energy targets should be used and how the final consumption sectors should be treated (they were excluded from the comparison experiment).

The implementation in GAMS/MPSGE is as follows. In the production block for the stylized example from above,  $\mu$  parameters are introduced to shift inputs in the production function, which are initialized at unity. The MPSGE code below contains the nested production function  $Y$  to produce good  $g$  in region  $r$  at time  $t$ . To produce output  $PG$ , intermediate inputs  $PA$  and factor inputs  $PL$  and  $RK$  are required. Note that in MPSGE, these variables ( $PG$ ,  $PA$ ,  $PL$ ,  $RK$ ) represent price values. The input quantities indicated in the  $q$ : fields are calibrated quantities that result from an initial calibration to a GDP pathway, adjustments to energy prices, etc.

```

$Prod:Y(g,r,t)s:klemsig t:0 e(ss):eesig
+ mat(ss):0 oil(e):0 gas(e):0 col(e):0 elc(e):0 ref(e):0 l(ss):0 k(ss):0
o:PG(g,r,t) q:vy(g,r) a:gov(r,t) t:rto(g,r)
i:PA(i,r,t) q:(mu(i,g,r,t)*y1_dpa(i,g,r,t)) p:(pa1(i,r,t)*(1+rfi(g,r))) a:gov(r,t) t:rfi(g,r) i.tl:
i:PL(r,t) q:(mu("va",g,r,t)*y1_dpl(g,r,t)) p:pl1(r,t) l:
i:RK(r,t) q:(mu("va",g,r,t)*y1_drk(g,r,t)) p:rk1(r,t) a:gov(r,t) t:trk(r,t) a:gov(r,t)

```

After each model solve, first the mu parameters for energy goods are adjusted according to the relative difference between model results and targets (*prperreng*) for energy goods *eng*

$$\mu(\text{eng},g,r,t) \cdot y1\_dpa(\text{eng},g,r,t) = prperreng(r,\text{eng},t) \cdot \mu(\text{eng},g,r,t);$$

If energy use in a particular model run is half of what it should be according to the exogenous target, then the mu parameter will be set to 2 in this equation.<sup>27</sup> For the non-energy intermediate inputs, *mu* parameters are calculated as follows:

$$\begin{aligned} \mu(i,g,r,t) \cdot (\text{not eng}(i)) \\ = & (PG1(g,r,t) \cdot y1(g,r,t) \cdot (1-rto(g,r)) \\ & - \text{sum}(\text{eng}, PA1(\text{eng},r,t) \cdot (1+rfi(g,r)) \cdot \mu(\text{eng},g,r,t) \cdot y1\_dpa(\text{eng},g,r,t))) \\ & / (\text{sum}(ii \cdot (\text{not eng}(ii)), PA1(ii,r,t) \cdot (1+rfi(g,r)) \cdot y1\_dpa(ii,g,r,t)) \\ & + PL1(r,t) \cdot y1\_dpl(g,r,t) \\ & + RK1(r,t) \cdot y1\_drk(g,r,t)); \end{aligned}$$

The intuition behind this calculation is to keep total cost ( $PG1(g,r,t) \cdot y1(g,r,t) \cdot (1-rto(g,r))$ ) constant. Taking cost for exogenous energy inputs as given, the input quantity of other inputs is then adjusted accordingly. Multipliers for factor inputs (i.e.  $\mu("va",g,r,t)$  above) are adjusted equivalently. As mentioned above, in further iterations towards energy targets, prices from prior to the energy calibration (*PA1*, *RK1*, *PL1*) are used.

Note that this adjustment method does not respect the CES nesting of the PET model in this step, i.e. the nesting structure is collapsed into a single nest to update the mu parameters. When making adjustments affecting the choice of consumption goods (e.g. energy goods vs. other goods), parameter adjustments are made in a similar fashion. This procedure has the option to "override" the initial preference parameters because they might be incompatible with other model targets.

### 4.3 Energy supply response and energy price calibration

The GAMS model version is set up to not only calibrate energy demand to exogenous values, but also to match exogenous energy prices for coal, gas and (crude) oil. The realization of endogenous energy prices is achieved by including a sector-specific factor and endogenously rationing the available quantity of that factor. The sector-specific factor is a common tool in CGE modeling of upward sloping demand curves; if placed in the top nest of a CES production function, the elasticity of substitution between this

---

<sup>27</sup> Note that in the actual implementation, *prperreng* is not strictly the relative deviation as large changes to the mu parameters can potentially cause numerical issues in finding a solution.

factor and other inputs can be chosen such that it resembles the supply elasticity for this sector (Boeters & Bollen, 2012). Since the payment to this factor is not initially present in the data, we split capital input into the respective sectors into input of the sector-specific factor and (remaining) capital input<sup>28</sup>, similarly to the treatment of land supply in the fortran version of the model (see documentation wiki). The sector specific factor can be interpreted as rent from resource extraction from scarce primary energy goods.

This setup can then be used to target primary energy prices at the calibration stage. For this purpose, the available quantity of the specific factor is rationed endogenously to match exogenous fossil fuel prices. This will lead to some endogenous price response, e.g. reduced demand for oil under a scenario with rising oil prices or increased demand for natural gas in a scenario with low gas prices. Any further calibration to exogenous energy targets is then done after this initial endogenous response, using the method described in section 4.2 above. After the full calibration to a baseline, the endogenous adjustment of the available sector specific factor is frozen and the price made variable. This allows for a price response in any counterfactual scenario.

Depending on the choice of the fossil fuel elasticities, the response e.g. to the introduction of a carbon tax will differ. In general, a lower supply elasticity will induce higher welfare costs for energy exporting countries as they are not able to capitalize on their fossil fuel resources, and therefore climate policies restricting the use of fossil fuels will be more consequential. Without activating this calibration option (or in the fortran version, where this option is not available) there is no sector specific capital. As without this calibration option, the capital employed in primary energy sectors in the baseline can be freely shifted to other sectors in a policy scenario, the cost of emission reduction is generally lower in energy producing countries or regions. With the option activated, the sector specific capital cannot be used in other sectors. Instead, the value of resources drops and the income of resource owners is reduced.

Figure 9 illustrates this by showing welfare impacts from a carbon policy. Two versions of the GAMS model version (differing only in the modeling of fossil fuel resources, i.e. with and without resource rents represented) were calibrated to the data and B2 baseline from O'Neill et al. (2010). To calibrate the resource endowments, fossil fuel prices are assumed to be increasing at an annual rate of ca. 1%.<sup>29</sup> Then, emissions in each region were reduced by similar percentages below the baseline to roughly follow the global emissions trajectory of RCP 4.5. When resource rents are explicitly modeled, costs for large exporters of fossil fuel (Transition Countries, tc; and to some extent Other Developing Countries, odc) suffered more from climate policy because of reduced fossil fuel prices. Note that this also leads to higher global welfare costs because fossil fuel resources are a resource that has less value – in the standard model version, the capital stock could shift to other sectors relatively easily. The difference in

---

<sup>28</sup> While the resources are separate from return to capital in the GTAP database, the current data processing routines merge these two elements before generating the input files used by the iPETS model. For simplicity, it is assumed that 65% of the payments to capital are resource rents.

<sup>29</sup> This leads to much higher energy prices compared to the standard fortran version. Whereas in the standard model, energy prices in the calibrated baseline change at a rate similar to the regional price level, they get more expensive with resources explicitly modelled.

modeling assumptions is also reflected in changes in the energy price. For example, in this simulation the output price for oil and gas (ong) in the Transition Countries is reduced by less than 1% in the standard model, but by up to 40% in the model with resource rents. This price drop can be beneficial for resource importers. For example, China (chn) and India (ind) even have some small welfare gains under the model with explicit fossil resources which is not the case in the standard model where all regions suffer at least some small welfare losses (see also Peterson & Weitzel, 2016 for a more detailed analysis on the role of international energy price repercussions on regional welfare).

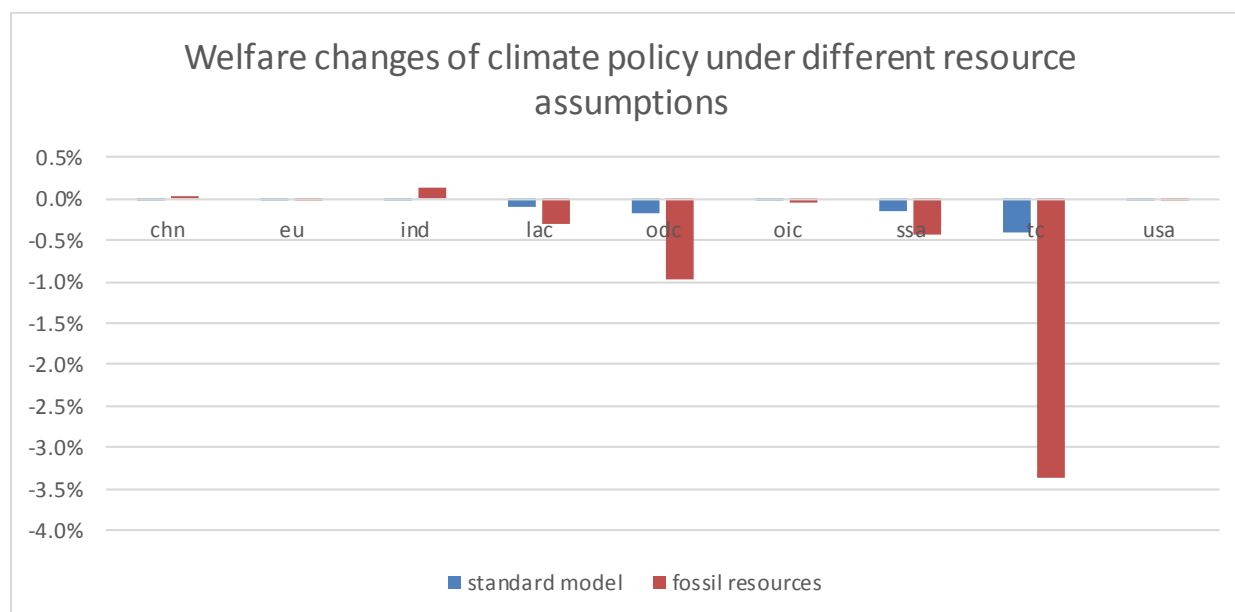


Figure 9. Welfare changes under different modeling assumption for fossil fuels. “Fossil resources” models sector a sector specific factor for primary fossil fuels and calibrates to exogenous fuel prices. In all other aspects, calibration to the baseline and policy scenarios are the same.

#### 4.4 Ability to calibrate for long term investment/GDP ratios

Initial calibration to a typical GDP pathway often yields unreasonable investment to GDP ratios for some regions, due to persistence of base year data in the dynamic calibration and the fact that GDP calibration is only achieved through changes in labor productivity. While the base year data is a good representation of the current situation, it is very likely for e.g. China to experience a lower savings rate over the coming decades and converge towards saving rates of other developed economies in the long term. To mitigate the persistence to base year data, an optional feature was introduced into the GAMS model to allow targeting reasonable long-term investment rates. Two adjustments are made to remedy unreasonable investment-GDP ratios that result from calibration of GDP to an SSP2 scenario (Figure 10). First, the tax on the rental rate of capital introduced in the steady state set-up (see section 3.1) is adjusted to converge to a rate of 20% by the end of the century.<sup>30</sup> In addition, the investment productivity is adjusted to bring investment to GDP ratios to reasonable ranges. A common choice

<sup>30</sup> The 20% rate is chosen as a reasonable (marginal) tax rate on capital observed in developed countries. Also note that the initial rate from which conversion starts can be an instrument to adjust *initial* investment to GDP ratios in the base year (see below).

would be adjusting productivity until investment ratios converge at a level close to current observed levels in developed countries (see Figure 11).

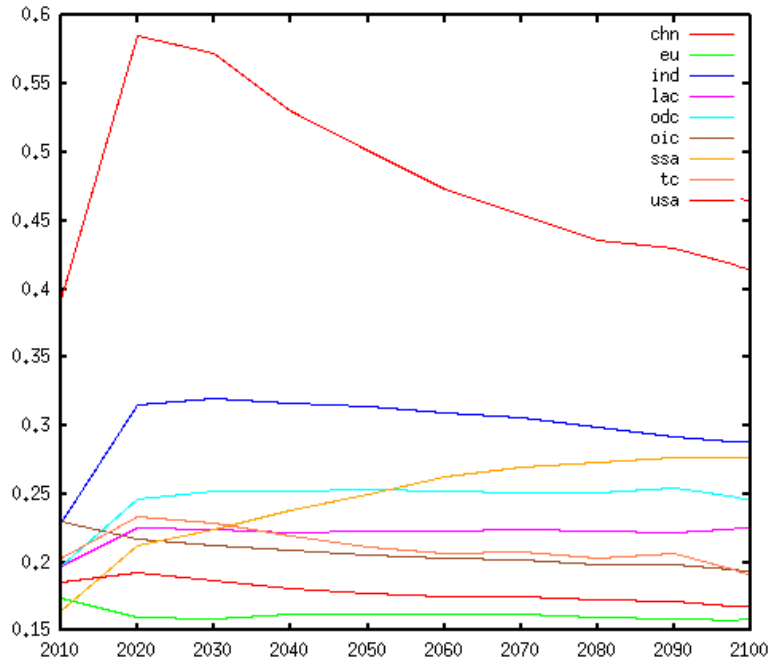


Figure 10. Investment to GDP ratios after GDP calibration to SSP2. Note the 40% savings rate for China in 2100. Results past 2100 are only in the model to approach a steady state path in the terminal period and are not interpreted in a typical model application.

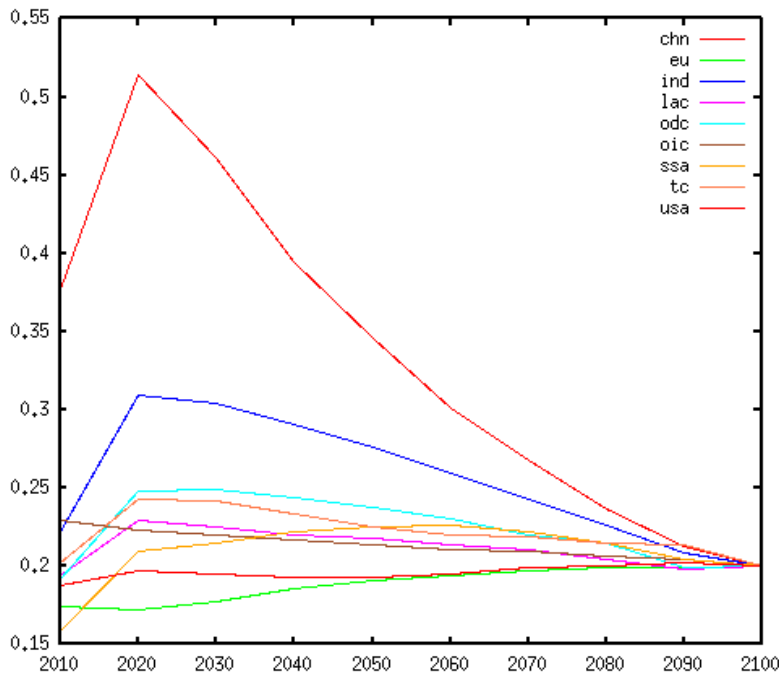


Figure 11. Investment to GDP ratios after long-term adjustments. Note that the adjustments have implications for investment to GDP ratios even early in the simulation, as can be seen in this example for China where the 2020 investment ratio is reduced by several percentage points from Figure 10 above.



#### 4.5 Calibration of initial investment/GDP ratio

Due to forward looking behavior of the PET model, incentives to invest in the base year change during calibration. This is at odds with the data fed into the model and several options exist to reconcile the model with observed investment/GDP ratios. The two methods implemented in the GAMS model are (1) adjustment of the tax rate on rental capital and (2) adjustment of the initial capital stock. There are several options to remedy this feature in the fortran version of the model; in a recent application, an adjustment to regional discount rates was used (Ren et al., 2016).

Adjusting the tax rate on rental capital influences the marginal consumption-investment choice and investment can be increased or decreased to obtain observed investment rates.<sup>31</sup> This introduces an additional equation (i.e., setting the investment/GDP ratio in the base year to the value observed in the input data) to the model; the associated additional variable introduced to the model is an endogenous tax rate on top of the tax rate on rental return to capital. This method adjusts the regionally differentiated tax rates on rental return to capital that were introduced to reconcile initial investment behavior as explained in section 3.1.

An alternative method for adjustment introduces the same additional constraint but adjusts the initial capital stock endowment, hence making capital more or less scarce and triggering an adjustment in investment rates. The additional variable added to the model in this case is a multiplier on the initial capital endowment.

#### 4.6 Government purchases

Further options in the GAMS model include the treatment of taxes and the constraint that determines the size of the government. While sector-specific taxes are reported in the data defining the base year IO table, these are not associated with specific inputs or outputs. Hence, it is possible to allocate the taxes in various ways that all conform to the initial IO table. The fortran version levies a sector's taxes on all intermediate inputs at the same rate, but not on factor inputs. This option is implemented in the GAMS model as well. However, the GAMS version can also be run with taxes levied on the output of goods (which is equivalent to taxing all inputs, including factor inputs, at the same rate).

In the fortran version, the purchase of government goods is exogenously determined by the growth in population and labor productivity. This feature is motivated by the desire to have the size of the government grow approximately at the rate of GDP. For impact or policy analysis, this approach also ensures a constant purchase of government goods, when comparing a baseline scenario with a counterfactual scenario.<sup>32</sup> When the price of labor declines faster than the price of the government good in calibration, this formulation leads to an increasing share of government expenditures in GDP. The GAMS version therefore also has an option available that accounts for price changes, setting government expenditures as a ratio of GDP.

---

<sup>31</sup> A similar feature is also implemented in the fortran version, although not with an automated search.

<sup>32</sup> See [http://ipetsmodel.com/index.php?title=PET\\_model\\_structure#Government](http://ipetsmodel.com/index.php?title=PET_model_structure#Government) for more details.

## 5 Conclusions

This paper describes the GAMS implementation of the PET model, especially in comparison to the fortran version. In terms of the model structure, both model versions are identical for the majority of structural equations. This has been demonstrated by model runs targeted at changes in population, preferences, labor productivity, capital endowment as well as factor productivity changes (as part of calibration). While the implementation design in the GAMS version differs from the fortran version by formulating the problem as a mixed complementarity problem and using MPSGE to describe production structure as well as capital dynamics, identical or very similar outcomes are achieved in both model versions.

A key difference between the model versions is the balance of payments constraint, which in the GAMS version is implemented as intertemporal constraint, while in the fortran model it is specified period-by-period. This has implications in the intertemporal adjustment through international trade. The default GAMS model allows agents to shift trade surpluses or deficits intertemporally and hence results between the model versions differ. However, to achieve very similar behavior, it is possible to add an additional balance of trade constraint in the GAMS model. This constraint then leads to behavior in the GAMS version which is almost identical to that of the fortran version. For runs without international trade and regions being completely independent, completely identical behavior was observed for shocks to labor productivity, changes to the initial capital stock, population or preferences. Further structural differences (i.e. those that result in different model outcomes) in the latest model versions are the missing land use sectors in the GAMS implementation and the absence of a special treatment of zero carbon electricity in the GAMS version.

In terms of implementation differences, the integration of the intertemporal and intratemporal problem into a single economic problem in the GAMS version sets it apart from the fortran version. In the fortran version, intertemporal and intratemporal problems can be separated and parallelization can be effectively used to achieve reasonable model runtime. Due to its modeling approach, the GAMS version can however easily be extended to dynamic variables beyond capital dynamics. This is already done by default with intertemporal optimization of trade. To make up for the inability to run the model in parallel and still reach a reasonable solution time, the GAMS model code is set up to allow for a reduction of the number of time steps that are solved explicitly.

We therefore conclude that this new implementation of the model is largely the same as the fortran model (or can be made the almost same with using specific options). In addition, there are options in both the fortran and GAMS models that can be turned on to move the model away from the specific model configuration in which both models behave very similarly. Additional options in the GAMS version are mainly targeted at a better implementation of a baseline scenario, e.g. to improve the investment to GDP ratio to more reasonable rates or to target energy prices in addition to energy quantities. An additional difference in the default calibration is related to the calibration of energy quantities, which might prove more robust than the standard approach used in the fortran version. This approach is however now also introduced in the fortran model version.

Beyond establishing comparability between the fortran and the GAMS versions of the iPETS model, the work behind this model comparison has allowed for a better understanding of the model's behavior. We were able to find small bugs in both the fortran and the GAMS version of the code that would most likely have gone unnoticed without this comparison. The comparison has also contributed to scrutinizing existing features of both model versions and thoughts about implementing it in the respective other model version. Currently, this is done mainly through optional switches, i.e. it is possible to maintain the default model version or resemble the respective other model version. In general, the close inspection of various parts of the models will be useful for future applications and development work.

## 6 References

- Armington, P. S. (1969). A theory of demand for products distinguished by place of production. *IMF Staff Papers*, 16(1), 159-178.
- Balistreri, E. J. (1997). A Few Simple Examples of Dynamic Calibration in MPSGE. Retrieved September 4, 2019, from <http://www2.econ.iastate.edu/faculty/balistreri/dyncalib/dyncal.html>
- Boeters, S., & Bollen, J. (2012). Fossil fuel supply, leakage and the effectiveness of border measures in climate policy. *Energy Economics*, 34, Supplement 2, S181–S189. <https://doi.org/10.1016/j.eneco.2012.08.017>
- Böhringer, C., Rutherford, T. F., & Wiegard, W. (2003), Computable General Equilibrium Analysis: Opening a Black Box. ZEW Discussion Paper No. 03-56, Mannheim.
- Ferris, M. C., & Munson, T. S. (2000). Complementarity problems in GAMS and the PATH solver. *Journal of Economic Dynamics and Control*, 24(2), 165–188. [https://doi.org/10.1016/S0165-1889\(98\)00092-X](https://doi.org/10.1016/S0165-1889(98)00092-X)
- A.P. Gruzdev, A. P., Melnikov, N. B., Dalton, M. G., Weitzel, M., O'Neill, B. C. (submitted). Parallel Extended Path Method for Solving Perfect Foresight Models
- Lau, M. I., Pahlke, A., & Rutherford, T. F. (2002). Approximating infinite-horizon models in a complementarity format: A primer in dynamic general equilibrium analysis. *Journal of Economic Dynamics and Control*, 26(4), 577–609. [https://doi.org/10.1016/S0165-1889\(00\)00071-3](https://doi.org/10.1016/S0165-1889(00)00071-3)
- Melnikov, N. B., Gruzdev, A. P., Dalton, M. G., & O'Neill, B. C. (2016). Parallel Algorithm for Calculating General Equilibrium in Multiregion Economic Growth Models. *Ural Mathematical Journal*, 2(2), 45–57. <https://doi.org/10.15826/umj.2016.2.005>
- O'Neill, B. C., Dalton, M., Fuchs, R., Jiang, L., Pachauri, S., & Zigova, K. (2010). Global demographic trends and future carbon emissions. *Proceedings of the National Academy of Sciences*, 107(41), 17521–17526. <https://doi.org/10.1073/pnas.1004581107>
- O'Neill, B. C., Ren, X., Jiang, L., & Dalton, M. (2012). The effect of urbanization on energy use in India and China in the iPETS model. *Energy Economics*, 34, S339–S345. <https://doi.org/10.1016/j.eneco.2012.04.004>
- Paltsev, S. (2004). Moving from static to dynamic general equilibrium economic models: Notes for a beginner in MPSGE. Joint Program on the Science and Policy of Global Change Technical Note 4, Massachusetts Institute of Technology, Cambridge, MA. <https://globalchange.mit.edu/publication/14238>

- Pernice, M., & Walker, H. (1998). NITSOL: A Newton Iterative Solver for Nonlinear Systems. *SIAM Journal on Scientific Computing*, 19(1), 302–318. <https://doi.org/10.1137/S1064827596303843>
- Peterson, S., & Weitzel, M. (2016). Reaching a climate agreement: compensating for energy market effects of climate policy. *Climate Policy*, 16(8), 993–1010. <https://doi.org/10.1080/14693062.2015.1064346>
- Ren, X., Weitzel, M., O'Neill, B. C., Lawrence, P., Meiyappan, P., Levis, S., Balistreri, E. J., Dalton, M. (2018). Avoided economic impacts of climate change on agriculture: integrating a land surface model (CLM) with a global economic model (iPETS). *Climatic Change*, 146:(3-4), 517–531. <https://doi.org/10.1007/s10584-016-1791-1>
- Ren, X., Lu, Y., O'Neill, B. C., Weitzel, M. (2018). Economic and biophysical impacts on agriculture under 1.5 °C and 2 °C warming. *Environmental Research Letters* 13(11). <https://doi.org/10.1088/1748-9326/aae6a9>
- Rutherford, T. F. (1999). Applied General Equilibrium Modeling with MPSGE as a GAMS Subsystem: An Overview of the Modeling Framework and Syntax. *Computational Economics*, 14(1/2), 1–46. <https://doi.org/10.1023/A:1008655831209>
- Weitzel, M., Balistreri, E. J., & Ren, X. (2016). *From recursive-dynamic to forward-looking: The importance of allowing for intertemporal investment and net trade adjustments* (Paper presented at the 19th Annual Conference on Global Economic Analysis, Washington DC, USA). Retrieved from [https://www.gtap.agecon.purdue.edu/resources/res\\_display.asp?RecordID=5065](https://www.gtap.agecon.purdue.edu/resources/res_display.asp?RecordID=5065)

# Appendix

## Current sectoral structure of the iPETS model (GAMS version)

Intermediate Goods	Final Goods
<b>Energy</b> <ul style="list-style-type: none"> <li>• Coal</li> <li>• (Crude) Oil</li> <li>• Natural Gas</li> <li>• Electricity</li> <li>• Refined Fuels</li> </ul>	<b>Energy</b> <ul style="list-style-type: none"> <li>• Coal</li> <li>• Electricity</li> <li>• Other Energy</li> </ul>
<b>Materials and other Production</b>	<b>Food</b>
	<b>Transport</b>
	<b>Other Consumption</b>
	<b>Investment Good</b>
	<b>Government Purchases</b>

## Current regional aggregation of the iPETS model (GAMS version)

