Dynamic General Equilibrium Model for Climate Resilient Economic Development

Andrej Drygalla, Katja Heinisch and Christoph Schult* | July 2020
Halle Institute for Economic Research

Deutsche Gesellschaft für Internationale Zusammenarbeit (GIZ) GmbH

IWH Leibniz-Institut für Wirtschaftsforschung Halle

On behalf of:

Federal Ministry for the Environment, Nature Conservation and Nuclear Safety

of the Federal Republic of Germany

* Research assistance by Yoshiki Wiskamp is greatly acknowledged.

Outline

1. Economic Models and Climate Change
2. Model building steps
3. Introduction to Dynare
4. DGE–CRED Model
5. Model Simulation and Calibration

Outline

1.1 Training Goals

- Inclusion of climate change and potential adaptation measures as variables in macroeconomic models to assess the impact of climate change and evaluate adaptation policies
- A sustainable implementation of the developed model
- Relationship between climate and the economy needs to be continuously monitored
- Allow for model flexibility to address specific policy questions when needed

1.2 Overview of Modeling Approaches

- Commonly top-down approaches are used to model the economy
- Three broad categories of macroeconomic models are used for modeling the interaction between climate and the economy
    1. Input-output models
    2. Macroeconometric models
    3. General Equilibrium models

Input-output (I-O) Models

- I-O models are based on national accounts and the respective supply and use tables
- I-O models are useful to study sectoral interdependencies
- I-O tables are not available on a sub-national level
- Long-term adjustment costs are overestimated using static I-O models
- See Miller and Blair 2009 for environmental input-output analysis

Macroeconometric Models

- Combination of I-O models with econometric models to conduct simulation studies
- Estimated parameters for the goods and labour market are used to inform the model about price changes
- The estimated behavioral equations are less valid the greater the simulation horizon
- Hsiang 2016 provides an excellent overview of climate econometrics

General Equilibrium Models (CGE)

- The class of general equilibrium models relies on the principle of optimizing agents
  - The system of equations is derived from optimization problems of households and firms
  - Optimization is based on expectations about the future
- One can distinguish between static and dynamic general equilibrium models
- For a comprehensive introduction to the main principles of general equilibrium models, see Wing 2004

General Equilibrium Models (CGE)

- See Arndt et al. 2015 for how to use CGE models to evaluate the impact of climate change on the economy
- This study uses a computational dynamic general equilibrium model for Vietnam
- Different climate change scenarios are simulated
- Climate change affects the evolution of sectoral productivity, increase in temperature e.g. has an adverse effect on rice crop yields

1.3 Our Approach

- We use a **dynamic general equilibrium (DGE) framework**, where
  - the model equations are explicitly derived from the optimization behaviour of representative agents
  - due to the dynamic nature of the model, the optimal decisions are derived at every point in time and thus show optimal reactions of agents to changing fundamentals
  - model provides a consistent framework for the interpretation of domestic and foreign economic shocks and the channels through which they affect variables

Key Characteristics of the Model (1)

- Small open-economy model
- Non-stationary model with direct mapping to statistical variables
- Regional and sectoral perspective:
  - Regional and sectoral shares of output, employment, wage bill
  - Regional and sectoral production functions
- National perspective: population, national output, taxes, government expenditure
- Three different agents: firms, households and government
- Climate variables are exogenous to economic variables:
  Temperature, wind speed, precipitation and sea level

Key Characteristics of the Model (2)

- The impact of climate (change) on the economy is modeled with the help of so-called “damage functions”
- These sector and/or region-specific equations link climate variables to economic outcomes
- “Damage” in that context refers to the fraction of potential production “lost” due to climate (change)
- Sectoral damage functions are calibrated based on particular sub-models that translate climate related events into economic impacts
- We will use results from the literature and econometric methods to calibrate the damage functions

Key Characteristics of the Model (3)

- DGE models can easily be built and run in Dynare, an **open source** pre-processor for Matlab (and its open source alternative Octave) that is **continuously supported online** and widely used
- Dynare allows the user to simulate and estimate the underlying structural model
- It is also possible to run Dynare with other open source software such as Julia, Python or R
- Translating model equations into Dynare code is **intuitive**

1.4 Advantages of the modeling approach

- Parameters can either be calibrated based on comparable studies, economic theory as well as period averages of actual data or be estimated using data up to the most recent period
- It is possible to track the transition dynamics from the initial state to the final state. It shows precisely how the economy “reacts” to the induced changes
- DGE models allow for the possibility of (transitory) disequilibria, e.g. unemployment or under-utilization of installed capital
- DGE models are parsimonious and more frequent updates are possible with respect to data requirements
- DGE models can be estimated (and simulated) with a subset of the data
- DGE models are estimated on the basis of national accounts data, and hence, they are capable to gauge the initial state of the economy, i.e. before climate change or adaptation policies come into effect, timelier and more precisely

1.4 Advantages of the Modeling Approach (cont.)

- The number of regions and sectors can be easily modified.
- Extensions to the baseline model can be tested easily.
- Our damage functions allow a transparent and replicable approach to include climate into the model.
- Expectations are formed in a model consistent way.

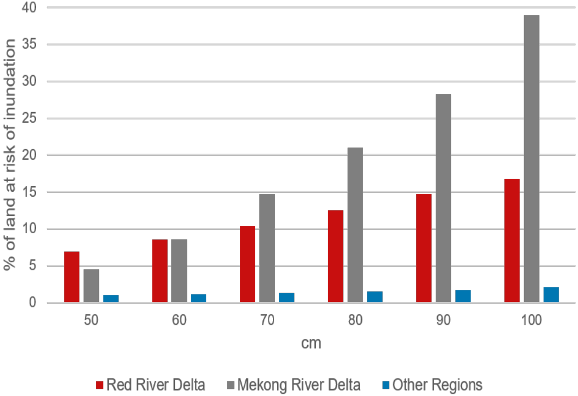
1.4 Limitations of the Modeling Approach

- Demand for computational power increases tremendously with the number of regions and sectors
- The number factors is smaller compared to I-O models and static CGE models
- Deviations from perfect foresight setup is not straightforward
- Specifying a functional relationship between climate and sectoral productivity restricts the functional form considered

1.5 Possible Applications

- Simulation of different climate change scenarios to determine upper bounds for costs of adaptation policies
- Decompose general equilibrium effects to identify relative contributions by individual climate variables
- Evaluate the impact of different tax rates for different sectors and regions to foster the transition process
- Cost-Benefit analysis example: building of a dike

Cost-Benefit of Building a Dike in the Mekong River Delta (1)

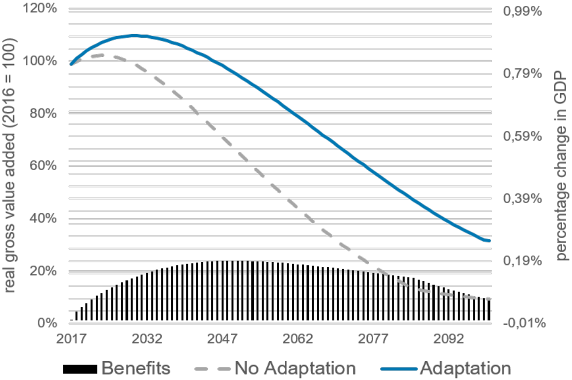


Source: Thuc et al. 2016

Cost-Benefit of Building a Dike in the Mekong River Delta (2)

- Facts and assumptions
  - Coast line of Mekong River Delta is 600 km
  - Dike needs to be of the same height along the entire coastline
  - Estimated costs: 40,000 EUR for one meter height and one meter length
  - Height increases with sea level each year
  - Damages associated with sea level rise in Mekong Delta River are zero, if and only if the height of the dike exceeds the sea level rise
- Benefit is defined as: difference between a scenario with dike (Adaptation) and without dike (Sea Level) relative to the Baseline scenario

Cost-Benefit of Building a Dike in the Mekong River Delta (3)



Source: own exhibition.

Note: Lines depict simulation of agricultural production and bars indicate the difference between Adaptation and No-Adaptation scenario for total GDP.

Outline

2. Model building steps
   - Definition of Scenarios
   - Identification of Climate Hazards
   - Calibration of Model Parameters

2.1 Definition of Climate Change Scenarios

- Determine climate change scenarios
- We use results from Thuc et al. 2016 to define regional specific climate change variables
- Scenarios are based on the IPCC scenarios RCP 4.5 and 8.5.
- It is possible to define regional specific paths for:
  - annual average temperature
  - annual average precipitation
  - sea level rise

2.1 Weather Extremes

- Thuc et al. 2016 do not explicitly report results for the frequency of weather extremes
- Define the relationship between annual averages and the frequency of weather extremes, e.g.
  - cyclones
  - droughts
- Previous studies report changes in the frequency of weather extremes for different climate change scenarios (see e.g. Arndt et al. 2015).

2.1 Baseline Scenario (1)

- Determine a hypothetical reference path
- For this path no further change in climate variables
- Define the evolution of sectoral TFP
- Disentangle the impact of different impact chains on the economy

2.1 Baseline Scenario (2)

- Model parameters are set to match characteristics of Vietnamese economy in 2016
- Catch-up process and transition to developed economy (Japan selected as plausible benchmark)
  - Geometric average annual growth rate set to 4%
  - Determine sectoral TFP shocks to match an assumed new long-run composition of gross value added by economic activity (agriculture 1%, industry 35%, services 65%)
  - Long-run dynamics of labour productivity to ensure employment shares converge to values observed for developed economies
  - Exogenous evolution of sectoral labour and total factor productivity is completed after 120 periods
- No consideration of climate (change) effects on economic development

2.2 Climate Hazards

- Vietnam will suffer from a rise of the sea level
- Cyclones will occur more frequently than before

2.3 Calibration

- It is necessary to define structural parameters of the model to reflect ....

Outline

3. Introduction to Dynare
   - What is Dynare?
   - Implementing a Model in Dynare
   - Steady State in Dynare
   - Deterministic Simulations in Dynare
   - Remarks and Examples
   - Macro Processor

3.1 What is Dynare?

- Dynare is an open-source program for dynamic general equilibrium modeling:
  - mainly a collection of different functions written for MATLAB
  - pre-processor translates Dynare files (.mod files) into MATLAB code

3.2 1 Model File

- A model file or mod-file (filename.mod) contains commands and blocks. Each command and each element of a block is terminated by a semicolon (;). Blocks are terminated by `end;`
- The model file complementary to these slides is *Introduction_Dynare.mod*
- Code lines within the mode file can be deactivated using %, // or /* ... */
- In order to run a model file:
  - The Dynare path has to be added to the search path of MATLAB

    ```
    addpath('C:\dynare\4.6.1\matlab')
    ```
  - Dynare executed to run the model in MATLAB

    ```
    dynare filename
    ```

3.2 Implementing a Model in Dynare

Neoclassical Growth Model (1)

- Households maximize lifetime utility subject to their budget constraint

$$\max_{\{c_t, k_t\}_{t=1}^{\infty}} = \sum_{t=1}^{\infty} \beta^{t-1} \frac{c_t^{1-\sigma}}{1-\sigma}$$

$$\text{s.t.} \quad c_t + k_t = A_t \, k_{t-1}^{\alpha} + (1-\delta) \, k_{t-1}$$

- It follows from the first order condition of the above problem with respect to consumption that the Lagrange multiplier is defined as

$$\lambda_t = \beta^{t-1} \, \sigma_t^{-\sigma},$$

representing the marginal utility of consumption

3.2 2 Neoclassical Growth Model (2)

- The resulting first order conditions are:

$$c_t^{-\sigma} = \beta\, c_{t+1}^{-\sigma} (\alpha\, A_{t+1}\, k_t^{\alpha-1} + 1 - \delta)$$

$$c_t + k_t = A_t\, k_{t-1}^{\alpha} + (1 - \delta)\, k_{t-1}$$

- By the definition $c_t = c_{t+1} = \bar{c}$ and $k_t = k_{t+1} = \bar{k}$ has to hold in the steady state
- Therefore, the first order conditions in the steady state case become:

$$\bar{c}^{-\sigma} = \beta\, \bar{c}^{-\sigma} (\alpha\, \bar{A}\, \bar{k}^{\alpha-1} + 1 - \delta)$$

$$\bar{c} + \bar{k} = \bar{A}\, \bar{k}^{\alpha} + (1 - \delta)\, \bar{k}$$

3.2 2 Neoclassical Growth Model (3)

- The steady state can be obtained analytically by solving the first equation for $\bar{k}$ and the second equation for $\bar{c}$:

$$\bar{k} = \left(\frac{1 - \beta(1-\delta)}{\beta\,\alpha \bar{A}} \right)^{\frac{1}{\alpha - 1}}$$

$$\bar{c} = \bar{A}\,\bar{k}^{\alpha} - \delta\,\bar{k}$$

- The neoclassical growth model considered in these slides, as well as the DGE-CRED model are *deterministic models*. There are no stochastic elements requiring expectancy terms or probability distributions. Therefore, the remainder of this introduction to Dynare focuses on this class of models.

3.2 3 Variables and Parameters (1)

- At the beginning of each model file, the endogenous (`var`) and exogenous (`varexo`) variables as well as parameters (`parameters`) have to be defined:

```
var = c k;
varexo = A;
parameters alpha_p beta_p gamma_p delta_p;
```

- Then, the parameter values have to be assigned:

```
alpha_p = 0.5;
beta_p = 0.95;
gamma_p = 0.5;
delta_p = 0.02;
```

3.2 3 Variables and Parameters (2)

- Optionally, a LaTeX name and long name can be assigned to a variable using the convention:

  ```
  variable_name $tex_name$ (long_name = 'quoted_string')
  ```

  - Example:

    ```
    var
    k $k$ (long_name = 'capital'),
    c $c$ (long_name = 'consumption');

    varexo
    A $A$ (long_name = 'technology');
    ```

3.2 3 Variables and Parameters (3)

- There are some restrictions to be kept in mind, when choosing variable and parameter names in Dynare:
  - Avoid names of built-in functions and commands
  - Minimize interface with Matlab or Octave functions
  - Example: Do not use correctly-spelled greek letters like `beta`, instead e.g. `betta`
- Note that by convention in Dynare, the time indice of a variable refers to the period when its value is determined. The typical example is for capital stock:
  - Since the capital stock entering the production function in the current period is determined in the previous period, the capital stock becomes `k(-1)`, and the law of motion of capital must be written as: `k = i + (1-delta)*k(-1)`
  - This convention can be modified using the `predetermined_variables` setting

3.2 4 The `model` block

- A model is declared inside a `model` block. In general, there must be as many equations as there are endogenous variables in the model. A great advantage of using Dynare is that the equations can be written almost as on paper.

```
model;
c + k = A*k(-1)^alpha_p + (1-delta_p)*k(-1);
c^(-gamma_p) = beta_p*c(+1)^(-gamma_p)
        *(alpha_p*A(+1)*k^(alpha_p-1) + 1 - delta_p);
end;
```

- Now, the model is set up and one can begin with the deterministic simulation.

3.3 Steady State (1)

- By definition a steady state of a model satisfies:

$$y_t = y_{t+1} = \bar{y} \text{ and } u_t = u_{t+1} = \bar{u},$$

  where y is the vector of endogenous variables and u the vector of exogenous shocks
  - In the context of the neoclassical growth model: $y_t = (c_t \quad k_t)'$ and $u_t = A_t$
- Note that a steady state is conditional to:
  - The steady state values of exogenous variables $\bar{u}$
  - The values of parameters (implicit in the above definition)
- Even for a given set of exogenous and parameter values, some (nonlinear) models have several steady states

3.3 Steady State (2)

- The steady state is an important concept in the framework of the DGE-CRED model:
  - In this model it is assumed that the economy is in steady state at the beginning of the initial period and then transits towards a new steady state, reached in the terminal period.
  - While the trajectories of the exogenous variables are given, those of the endogenous variables are determined within the model.
- There are three approaches to calculate the steady state in Dynare
- The steady state values are stored in the MATLAB matrix `oo_.steady_state`

3.3 Steady State: Approach 1

- Idea: Provide an initial guess for the steady state in the `initval` block and then conduct the steady state calculation using `steady`
- Example, considering the neoclassical growth model:

```
initval;
c = 2;
k = 30;
A = 1;
end;
steady;
```

3.3 Steady State: Approach 2

- Idea: Use a `steady_state_model` block, in which the steady state values are calculated
- Example, considering the neoclassical growth model:

```
initval;
A = 1;
end;
steady_state_model;
k = ((1 - beta_p*(1 - delta_p))/
        (beta_p*alpha_p*A))^(1/(alpha_p - 1));
c = A*k^alpha_p - delta_p*k;
end;
```

3.3 Steady State: Approach 2

- Note that the steady state values of the exogenous variables have to be assigned in an `initval` block
- In cases where the steady state can be solved analytically, using a `steady_state_model` block is a suitable approach.

3.3 Steady State: Approach 3

- Idea: Use an explicit steady state file, which is an external MATLAB-file that must conform with a certain structure and naming convention: `NAMEofMODfile_steadystate.m`
- In this steady state file, you must provide the exact steady state values as in the case of the `steady_state_model` block
- Advantage: Flexibility, can call build-in MATLAB functions, allows for changing parameters to take parameter dependencies into account without resorting to model-local variables
- Drawback: The additional flexibility offered by a steady state file increases the scope for errors
- Note: A steady state file is used in the DGE–CRED model

3.4 1 Deterministic Simulation (1)

- The deterministic simulation builds up on the concept of *perfect foresight*, in which agents have full knowledge and perfectly anticipate all future shocks
- More precisely, we assume that:
  - agents learn the value of all future shocks;
  - since there is shared knowledge of the model and of future shocks, agents can compute their optimal plans for all future periods;
  - optimal plans are not adjusted in periods 2 and later

    $\Rightarrow$ the model behaves as if it were deterministic
- Cost of this approach: The effect of future uncertainty is not taken into account
- Advantage: Numerical solutions can be computed exactly (up to rounding errors) and nonlinearities are fully taken into account

3.4 1 Deterministic Simulation (2)

- The general problem in the deterministic, perfect foresight, case can be expressed as:

$$f(y_{t+1}, y_t, y_{t-1}, u_t) = 0,$$

where y is the vector of endogenous variables and u the vector of exogenous shocks

- Identification rule: There must be as many equations in $f(...)$ as there are endogeneous variables in y
- The general perfect foresight problem for the neoclassical growth model is:

$$f(y_{t+1}, y_t, y_{t-1}, u_t) = \begin{pmatrix} c_t^{-\sigma} - \beta\, c_{t+1}^{-\sigma}(\alpha\, A_{t+1}\, k_t^{\alpha-1} + 1 - \delta) \\ c_t + k_t - A_t\, k_{t-1}^{\alpha} - (1-\delta)\, k_{t-1} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

where $y_t = (c_t \;\; k_t)'$ and $u_t = A_t$

3.4 1 Deterministic Simulation (3)

- The aim of a deterministic simulation is to examine the trajectories of the model variables over the time period $t = 1, ..., T$
- Consequently, the stacked system for a perfect foresight simulation over T periods becomes a two-boundary value problem:

$$\begin{cases} f(y_2, y_1, y_0, u_1) = 0 \\ f(y_3, y_2, y_1, u_2) = 0 \\ \vdots \\ f(y_{T+1}, y_T, y_{T-1}, u_T) = 0 \end{cases},$$

  where y_0 and y_{T+1} as well as $u_1, ..., u_T$ are given
- Dynare uses a Newton-type methode to solve this stacked system

3.4 1 Deterministic Simulation (4)

- The Newton method numerically solves the two-boundary value problem and hence computes trajectories for given shocks over a *finite* number of periods.
- If one is rather interested in solving an *infinite*-horizon problem, the easiest way is to approximate the solution by a finite-horizon problem (large T). The drawback of this approach is that the solution is specific to a given sequence of shocks, and not generic.
- In case there is more than one lead or lag, Dynare automatically transforms the model in the form with one lead and one lag using auxiliary variables. For example, if there is a variable with two leads x_{t+2}:
  - create a new auxiliary variable a
  - replace all occurrences of x_{t+2} by a_{t+1}
  - add a new equation: $a_t = x_{t+1}$

3.4 2 Newton Method (1)

- The following slides aim to provide an intuition for the Newton method used by the perfect foresight solver in Dynare
- Start from an initial guess $Y^{(0)}$, where $Y = [y_1' \quad y_2' \quad \dots \quad y_T']'$
- Iterate: Updated solutions $Y^{(k+1)}$ are obtained by solving a linear system

$$F(Y^k) + \left[\frac{\partial F}{\partial Y}\right](Y^{(k+1)} - Y^{(k)}) = 0$$

3.4 2 Newton Method (2)

- Terminal condition for the solver:

$$\|Y^{(k+1)} - Y^{(k)}\| < \epsilon_Y$$

or

$$\|F(Y^{(k)})\| < \epsilon_F$$

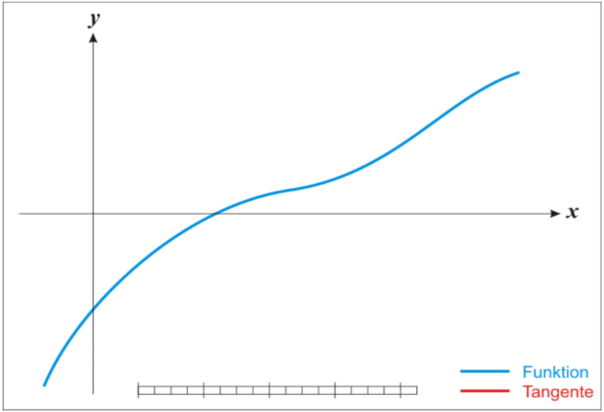
- Convergence may never happen if function is ill-behaved or initial guess $Y^{(0)}$ is too far from solution
$\Rightarrow$ to avoid an infinite loop, abort after a given number of iterations

3.4 2 Newton Method (3)

- The following options to the `perfect_foresight_solver` can be used to control the Newton algorithm:
  - `maxit`: Maximum number of iterations before aborting (default: 50)
  - `tolf`: Convergence criterion based on function value (ϵ_F) (default: 10^{-5})
  - `tolx`: Convergence criterion based on change in function argument (ϵ_Y) (default: 10^{-5})
  - `stack_solver_algo`: select between the different flavors of Newton algorithms

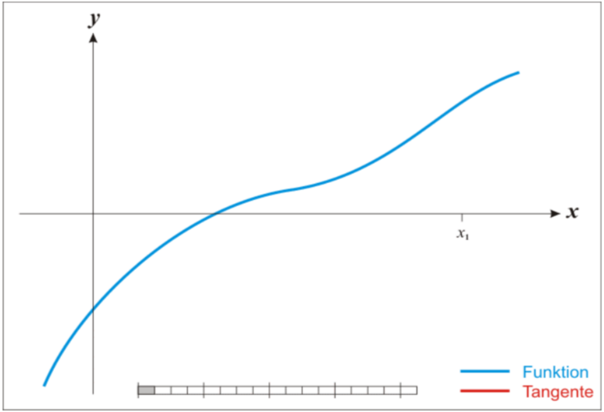
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



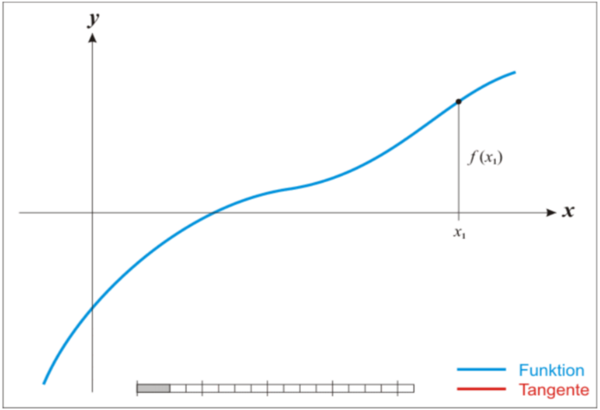
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



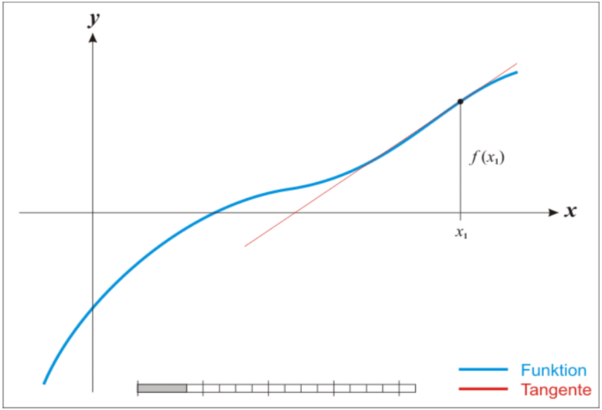
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



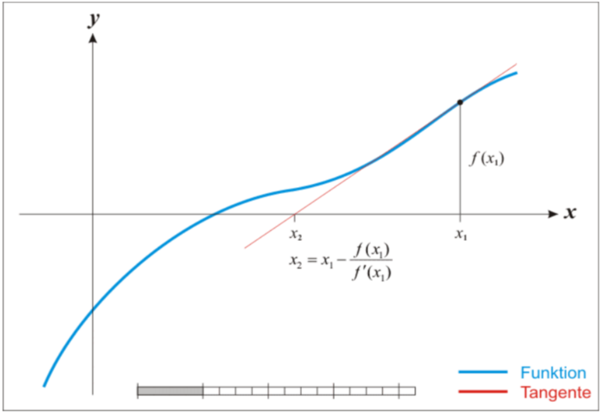
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



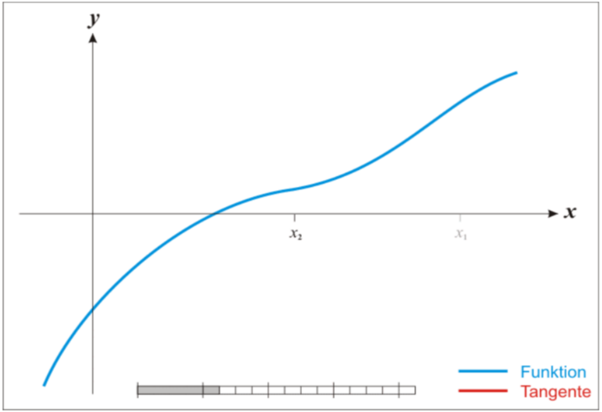
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



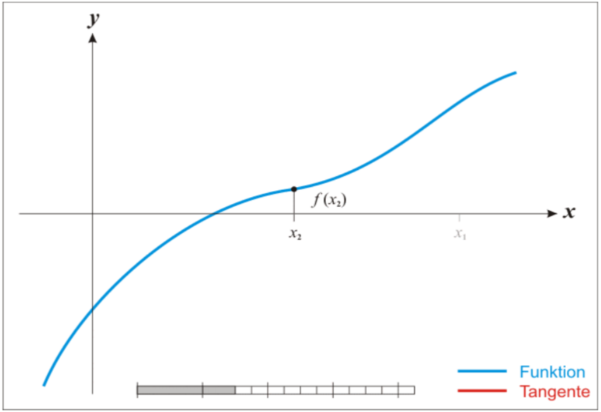
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



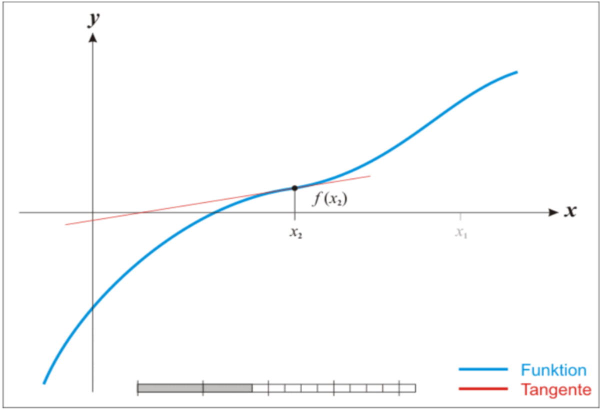
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



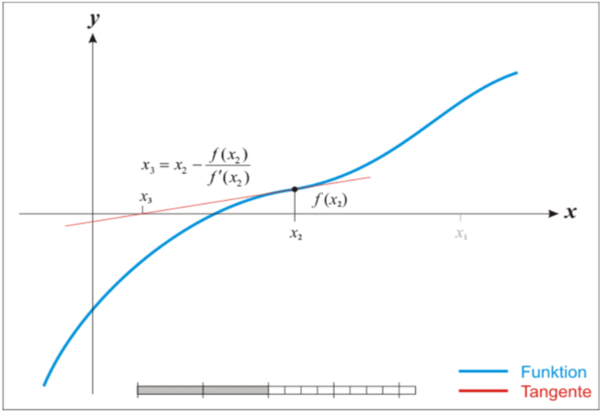
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



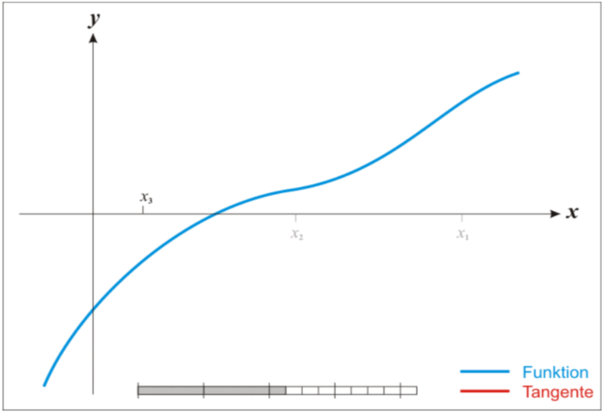
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



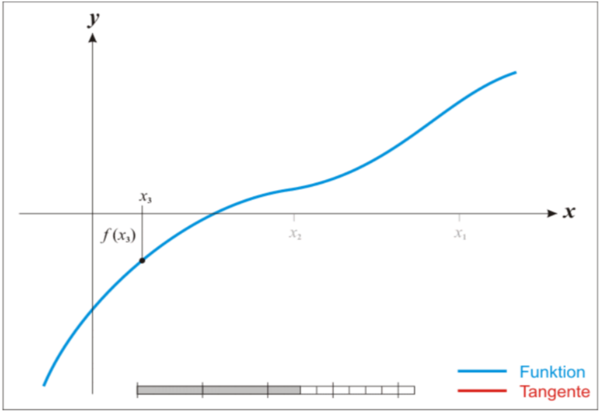
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



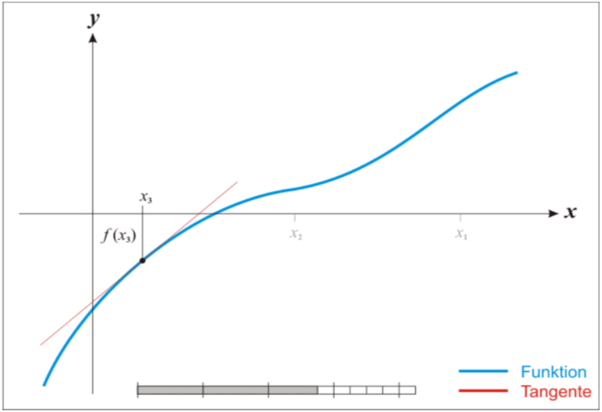
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



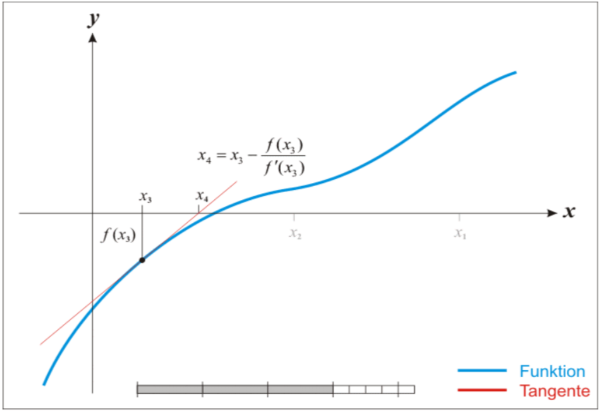
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



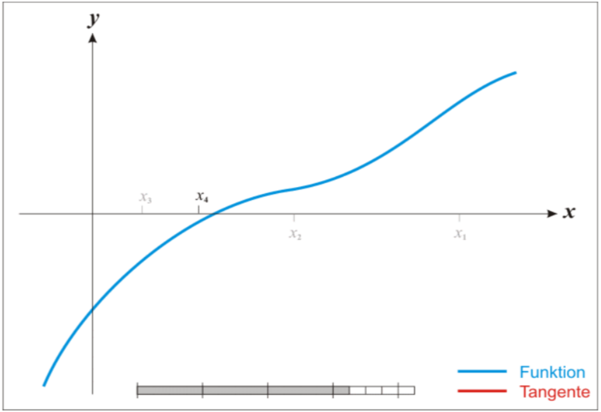
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



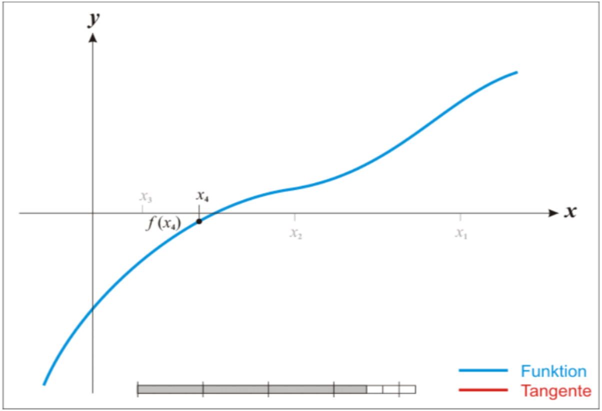
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



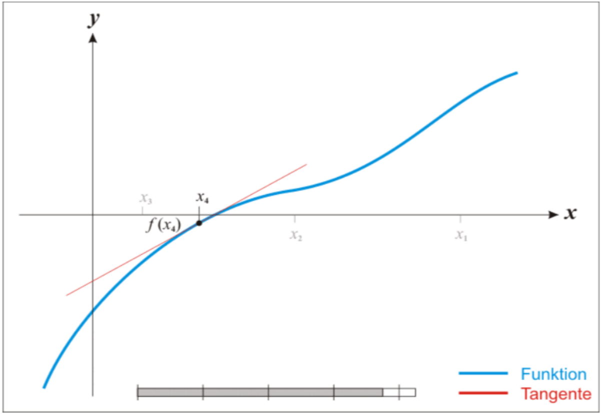
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



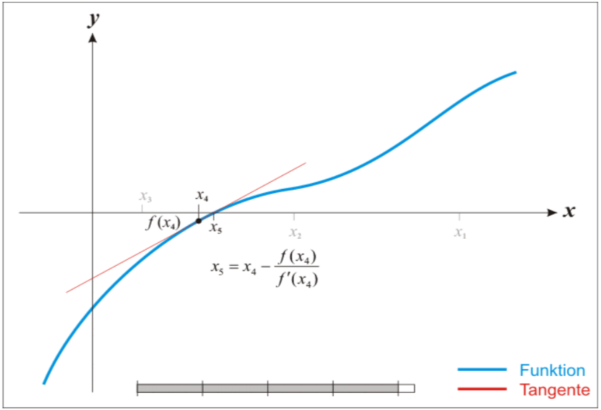
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



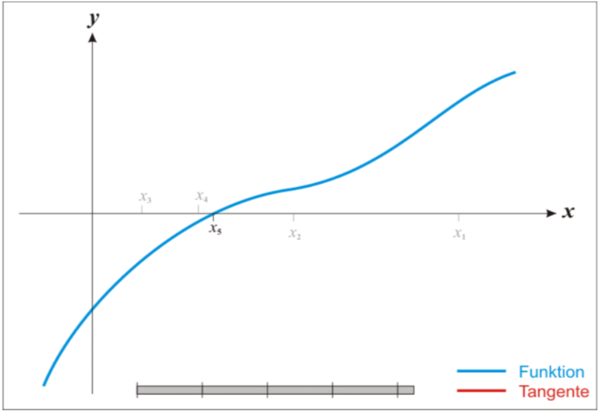
3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



3.4 2 Newton Method (4)

- Graphical illustration of the Newton method (unidimensional):



3.4 3 Perfect Foresight

- In order to perform the deterministic simulation, one has to specify (i) the constraints of the stacked system y_0, y_{T+1} and $u_1, \dots, u_T$ and (ii) provide an initial guess $y_1, \dots, y_T$ for the Newton algorithm
- The path for the endogenous and exogenous variables are stored in two MATLAB/Octave matrices:
  - `oo_.endo_simul` $= (y_0 \;\; y_1 \;\; \dots \;\; y_T \;\; y_{T+1})$
  - `oo_.exo_simul'` $= (y_0 \;\; y_1 \;\; \dots \;\; y_T \;\; y_{T+1})$
- The `perfect_foresight_setup` initializes those matrices, given the `shocks`, `initval`, `endval` and `histval` blocks
- Then, the `perfect_foresight_solver` replaces $y_1, \dots, y_T$ by the solution

3.4 3 Transition from an Initial to a Terminal Steady State (1)

- The following slides examine a specific type of deterministic simulation, which is needed in the DGE-CRED framework.
- The DGE-CRED Model allows its user to investigate the trajectories of the endogenous variables, given the parameter values and pathways of the exogenous variables.
- It is assumed that the economy is in an inital steady state at the beginning of the model period ($t = 0$) and transits towards a new steady state in the terminal period ($t = T + 1$).

giz Deutsche Gesellschaft für Internationale Zusammenarbeit (GIZ) GmbH

3.4 3 Transition from an Initial to a Terminal Steady State (2)

- In order to implement this specific type of deterministic simulation, the following can be conducted in Dynare:
  - Use a steady state file. This enables a steady state computation for varying values of the exogenous variables and allows using numeric solvers.
  - Declare the initial values of the exogenous variables in an `initval` block followed by `steady`.
  - Declare the terminal values of the exogenous variables in an `endval` block followed by `steady`.
  - Note: This is basically the same idea as presented in “approach 3” of the section covering the steady state calculation.

3.4 3 Transition from an Initial to a Terminal Steady State (3)

- The initial steady state values are determined based on the values of the exogenous variables assigned in the `initval` block.
  - These values are then stored as initial values in `oo_.endo_simul` and `oo_.exo_simul`.
- The terminal steady state values are determined based on the values of the exogenous variables assigned in the `endval` block.
  - These values are then stored as terminal values as well as initial guess for the numerial solver in `oo_.endo_simul` and `oo_.exo_simul`.

3.4 3 Transition from an Initial to a Terminal Steady State (4)

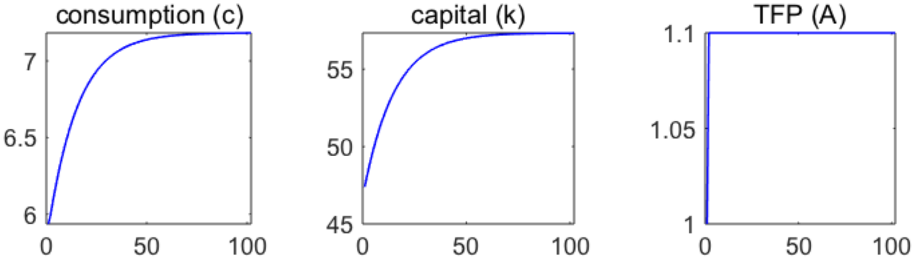
- Example: The economy starts from the initial steady state, where $A_0 = 1$. In the terminal steady state, the total factory productivity is 10% higher.

```
initval;
A = 1;
end;
steady;

endval;
A = 1.1;
end;
steady;
```

3.4 3 Transition from an Initial to a Terminal Steady State (5)

- The following trajectories are obtained:



3.4 3 The `initval` Block (1)

- While the previous slides examined the usage of the `initval` and `endval` block in combination with `steady`, these blocks have more functionalities than encountered so far.
- In presence of the `initval` and absence of other blocks:
  - `oo_.endo_simul` and `oo_.exo_simul` variables storing the endogenous and exogenous variables will be filled with the values provided by this block.
  - It will therefore provide the initial and terminal conditions for all the endogenous and exogenous variables.
  - For the intermediate simulation periods it provides the starting values for the solver.
- It is important to be aware that if some variables, endogenous or exogenous, are not mentioned in the `initval` block, a zero value is assumed.

3.4 3 The `initval` Block (2)

- Example: Let us assume that we want to set $c_0 = c_{T+1} = 4$ and $k_0 = k_{T+1} = 20$ in the neoclassical growth model. Furthermore, we assume that TFP is constant at $A_t = 1$.

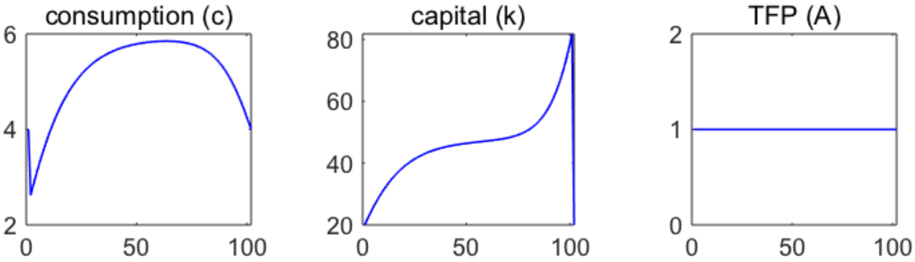
```
initval;
c = 4;
k = 20;
A = 1;
end;
```

- Note that the purpose of this example is to illustrate the usage of the `initval` block, rather than adressing a meaningful question.
- In order to run the deterministic simulation for $T = 100$ periods, enter:

```
perfect_foresight_setup(periods=100);
perfect_foresight_solver;
```

3.4 3 The `initval` Block (3)

- The following trajectories are obtained:



- Comments:
  - As consumption is a forward looking variable in this model, its initialization c_0 does not affect the trajectory. However, its terminal value does.
  - The opposite holds for capital.

3.4 3 The `endval` Block (1)

- In the absence of an `initval` block, the `endval` block fills both `oo_.endo_simul` and `oo_.exo_simul`. In this case it, therefore, has the same effect as if only an `initval` block was present.
- However, if an `initval` and `endval` block are both present, the former assigns the initial conditions in $t = 0$ while the latter provides the terminal conditions in $t = T + 1$ as well as the initial guess for the perfect foresight solver.
- Example: Let us assume that we want to set $c_0 = 4$, $c_{T+1} = 6$, $k_0 = 20$, $k_{T+1} = 30$ and $A_t = 1$.
- As before, the purpose of this example is to illustrate the usage of the `endval` block, rather than adressing a meaningful question.

3.4 3 The `endval` Block (2)

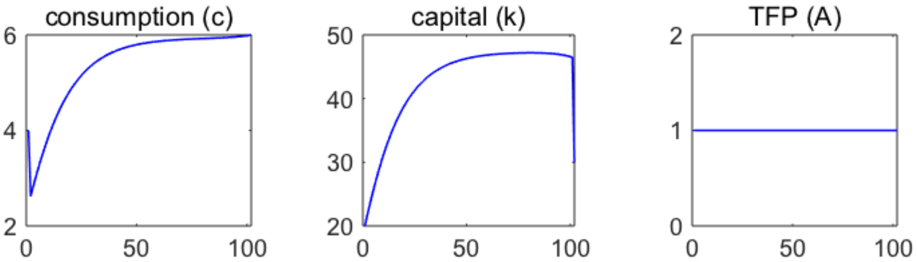
- Leading the following code:

```
initval;
c = 4;
k = 20;
A = 1;
end;

endval;
c = 6;
k = 30;
A = 1;
end;
```

3.4 3 The `endval` Block (3)

- The following trajectories are obtained:



- Comment: As in the previous example, consumption is forward looking, while capital is backward looking.

3.4 3 The `histval` Block (1)

- The usage of the `histval` block is particularly interesting, if there are variables with more than one lead or lag.
- In a deterministic simulation the `histval` block must be combined with an `initval` block. A `histval` and `endval` block cannot be combined.
- The `histval` block assigns the initial condition, while the `initval` block provides the terminal condition and inital guess for the perfect foresight solver.

3.4 3 The `histval` Block (2)

- The previous example can, therefore, also be implemented by:

```
histval;
c(0) = 4;
k(0) = 20;
A(0) = 1;
end;

initval;
c = 6;
k = 30;
A = 1;
end;
```

3.4 3 Shocks on Exogenous Variables (1)

- For deterministic simulations, the `shocks` block specifies temporary changes in the value of exogenous variables. For permanent shocks, use an `endval` block.
- It is possible to specify shocks which last several periods and which can vary over time. The `periods` keyword accepts a list of several dates or date ranges, which must be matched by as many shock values in the `values` keyword.
- Note that a range in the `periods` keyword can be matched by only one value in the `values` keyword. If values represents a scalar, the same value applies to the whole range. If values represents a vector, it must have as many elements as there are periods in the range.

3.4 3 Shocks on Exogenous Variables (2)

- The `shock` block has the following structure:

```
shocks;
var ... ;
periods ... ;
values ... ;
end;
```

- Examples using the `shock` block will be examined at the end of this section.

3.5 1 Remarks

- Because of the various functions of the `initval`, `endval` and `histval` blocks, it is strongly recommended to check the constructed `oo_.endo_simul` and `oo_.exo_simul` variables after running `perfect_foresight_setup` and before running `perfect_foresight_solver` to see whether the desired outcome has been achieved.
- `simul(periods = T)` executes both, `perfect_foresight_setup` and `perfect_foresight_solver` at the same time and can also be used.
- The following slides cover some examples based on the neoclassical growth model. It is recommendable to take a look at the file *Introduction_Dynare.mod* while proceeding with these examples.

3.5 2 Example No. 1

- Scenario 1: Return to equilibrium starting from $k_0 = 0.5\bar{k}$.

```
...
steady;

ik = varlist_indices('k',M_.endo_names);
kstar = oo_.steady_state(ik);

histval;
k(0) = kstar/2;
end;
...
```

3.5 2 Example No. 2

- Scenario 2: The economy starts from the steady state. There is an unexpected positive productivity shock at the beginning of period 1: $A_1 = 1.1$.

```
...
steady;

shocks;
var A;
periods 1;
values 1.1;
end;
...
```

3.5.2 Example No. 3

- Scenario 3: The economy starts from the steady state. There is a sequence of shocks to A_t: 10% in period 5 and an additional 5% during the 4 following periods.

```
...
steady;

shocks;
var A;
periods 4, 5:8;
values 1.1, 1.05;
end;
...
```

3.5 2 Example No. 4

- Scenario 4: The economy starts from the initial steady state. In period 6, TFP increases by 10% permanently.
  - Same `initval` and `endval` blocks as in Scenario 4.
  - A shock block is used to maintain TFP at its initial level during periods 1-5.

```
...
shocks;
var A;
periods 1:5;
values 1;
end;
...
```

3.6 1 Macro-processing Language (1)

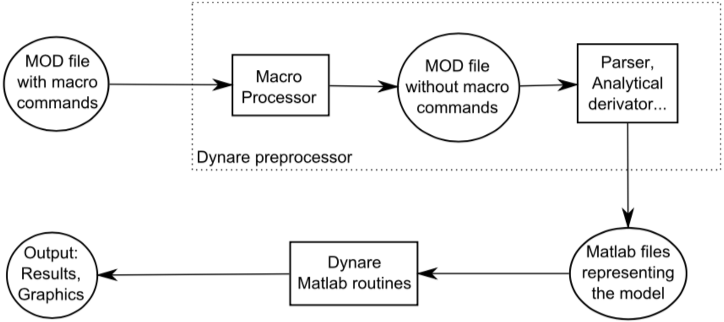
- So far, we have encountered the **Dynare language** (used in mod-files), which is well suited for many economic models.
  - The Dynare language is a markup language for MATLAB/Octave that defines models,
  - but itself lacks a programmatic element.
- The **Dynare macro language** adds a programmatic element to Dynare.
  - It allows to replicate blocks of equations through loops (`for` structure), conditionally executing some code (`if/then/else` structure), writing indexed sums or products inside equations, etc.
  - It is used to speed model development,
  - and useful in various situations. Examples: Multi-region models, creation of modular mod-files, variable flipping, conditional inclusion of equations, etc.

3.6 1 Macro-processing Language (2)

- The Dynare macro language provides a new set of **macro commands** that can be used in mod-files.
- Technically, this macro language is totally independent of the basic Dynare language, and is processed by a seperate component of the Dynare pre-processor.
- This macro processor transforms a mod-file with macros into a mod-file without macros (doing text expansion/inclusions) and then feeds it to the Dynare parser.
- The key point is to understand that the macro processor only does text substitution (like the C preprocessor or the PHP language).

3.6 1 Macro-processing Language (3)

- The flowchart below illustrates the relation between the Dynare macro language, macro processor, Dynare language and MATLAB/Octave.



Source: Villemot, S. and H. Bastani: “The Dynare Macro Processor - Dynare Summer School 2019”.

3.6 2 Introduction to the Syntax (1)

- The macro-processor is invoked by placing *macro directives* in the mod-file.
- Directives begin with: `@#`
- The main directives are:
  - `@#include`: file inclusion
  - `@#define`: definition of a macro processor variable
  - `@#if`, `@#ifdef`, `@#ifndef`, `@#else`, `@#endif`: conditional statements
  - `@#for`/`@#endfor`: loop statements
- Most directives fit on one line. If needed however, two backslashes at the end of a line indicate that the directive is continued on the next line.

3.6 2 Introduction to the Syntax (2)

- The macro processor has its own list of variables which are different than model variables and MATLAB/Octave variables.
- There are 4 types of macro-variables:
  - integer
  - string
  - integer array
  - string array
- Note that there is no boolean type:
  - false is represented by integer zero
  - true is any non-zero integer
- As the macro-processor cannot handle non-integer real numbers, integer division results in the quotient with the fractional part truncated (hence, $5/3 = 3/3 = 1$).

3.6 2 Introduction to the Syntax (3)

- The following slides introduce the directives used in the DGE-CRED model. In order to be able to use and modify the DGE-CRED, it is sufficient to master these commands.
- The sources listed below encompass a comprehensive explanation of the *entire* syntax of the macro language:
  - Villemot, S. and H. Bastani: “The Dynare Macro Processor - Dynare Summer School 2019”.
  - Chapter 4.24 in Dynare Reference Manual Version 4.

3.6 2 Introduction to the Syntax (4)

- The value of a macro-variable can be defined with the `@#define` directive.

  ```
  @#define Regions = 3
  ```

- Macro-expressions can be used in two places:
  - Inside a macro directive; no special markup is required (as in the example above).
  - In the body of the mod-file, between an at sign and curly braces (like `@expr`); the macro processor will substitute the expression with its value:

    ```
    parameters z;
    z = @{Regions};
    ```

    In the example above the value of 3 is assigned to parameter z.

3.6 2 Introduction to the Syntax (5)

- The `include` directive simply inserts the text of another file in its place.

```
@#include "ModFiles/DGE_CRED_Model_Equations.mod"
```

- It is equivalent to copy/paste of the content of the included file.
- Note that it is possible to nest includes (i.e. to include a file with an include file).
- Files to include are searched for in the current directory. Other directories can be added with the `@#includepath` directive.

3.6 2 Introduction to the Syntax (6)

- Loops can be implemented using the `@#for...@#endfor` directive.
- In a model encompassing multiple regions, loops can faciliate setting up a model block:

```
model;
@#for reg in 1:Regions
      Y_@{reg} = A_@{reg} * (K_@{reg})^alpha
                          * (L_@{reg})^(1-alpha);
@#endfor
...
end;
```

3.6 2 Introduction to the Syntax (7)

- The previous example also illustrates a great advantage of the macro language compared to the MATLAB/Ocatve syntax.
  - In MATLAB it is not possible to change the index of a variable, for example Y_r, K_r, L_r.
  - Intuition: The Dynare macro as well as basic language rather have the character of a text than code, which is then transforemed by Dynare and MATLAB into a code.
  - It is also possible to use the MATLAB syntax directly in a mod-file. Dynare simply does not have to translate this part of the file further.

3.6 2 Introduction to the Syntax (8)

- The most basic conditional directive is the if-statement. It is executed by the directive `@#if,...,@#else,@#endif`.

```
@#define RCP_scenario = 4.5 // choose: 4.5 or 8.5
parameters Temp_2100;
...
@#if RCP_scenario == 4.5
    Temp_2100 = 2.3;
@#else
    Temp_2100 = 3.9;
@#endif
```

3.6 2 Introduction to the Syntax (9)

- In the example on the previous slide, `Temp_2100` is a parameter representing the temperature in 2100. Its value is assigned using a conditional statement.
- The lines between `@#if`, `@#else` or `@#endif` are executed only if the condition evaluates to a non-null integer (i.e. is true). The `@#else` branch is optional and, if present, is only evaluated if all other conditions evaluate to 0 (i.e. do not hold).

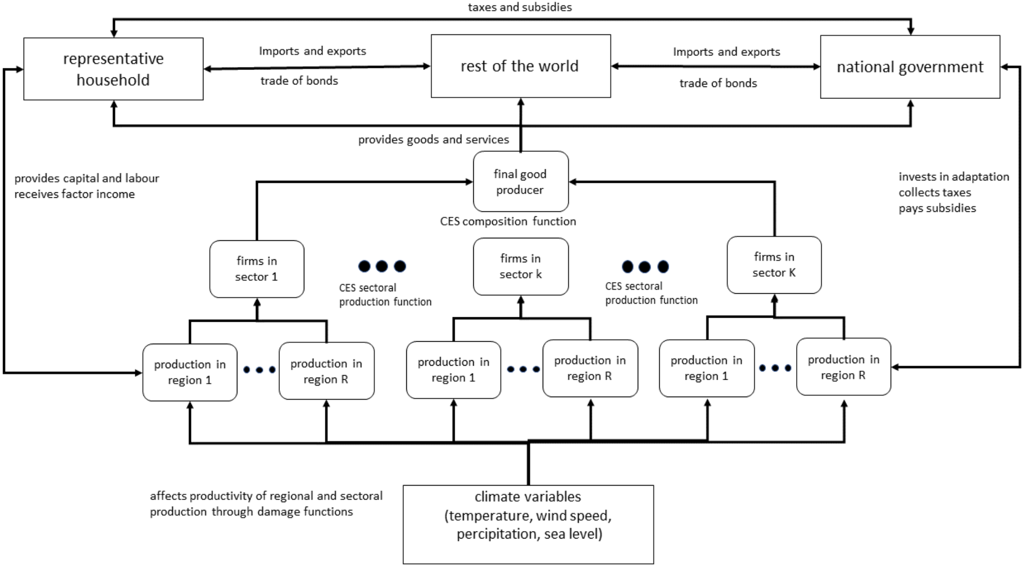
Outline

4. DGE–CRED Model
   - Introduction
   - Folder Structure
   - Demand
   - Production
   - Climate Variables
   - Aggregation and Identities

4.1 Introduction

- Dynamic general equilibrium (DGE) model with optimizing agents
- Differentiates between several regions and main economic activities.
- Our model is implemented in the open source environment Dynare and can be run using Matlab or Octave.
  - Sectors in the model correspond to economic activities and the classification by the General Statistical Office (GSO).
  - Regions are based on the statistical regions.
- We extend the approach by Nordhaus 1993 to model the impact of climate change through damage functions.

4.1 Model Structure



4.2 Main Folder Structure

- The main file containing all necessary mod-files is `DGE_CRED_Model.mod`. This file includes the following files stored in the `ModFiles` folder
- Subroutines responsible for finding the initial and terminal conditions are located in the subfolder `Functions`
- To define scenarios and structural parameters you need to create an Excel file located in the subfolder `ExcelFiles`

4.2 Mod Files

- The main file containing all necessary mod-files is `DGE_CRED_Model.mod`. This file includes the following files stored in the `ModFiles` folder:
  - `DGE_CRED_Model_Declarations.mod` declares all endogenous and exogenous variables of the model and structural parameters.
  - `DGE_CRED_Model_Parameters.mod` assigns values to the structural parameters of the model.
  - `DGE_CRED_Model_Equations.mod` contains the equations of the model.
  - `DGE_CRED_Model_LatexOutput.mod` produces latex output for documentation of the declared variables and model equations.
  - `DGE_CRED_Model_SteadyState.mod` computes initial and terminal condition for the dynamic simulation.
  - `DGE_CRED_Model_Simulations.mod` starts the dynamic simulation.

4.2 Functions

- Subroutines responsible for finding the initial and terminal conditions are located in the subfolder `Functions`:
  - `Calibration.mat` finds the initial conditions to reflect a specific year of the economy.
  - `FindA.mat` looks for exogenous productivity shocks across sectors and regions to meet the terminal conditions.
  - `FindK.mat` looks for a capital allocation across sectors and regions to fulfill the static equations of the model.
  - `rng.mat` random number generator function necessary for Octave users.
  - `LoadExogenous.mat` reads exogenous variables for different scenarios.

4.2 Excel Files

- To define scenarios and structural parameters you need to create an Excel file located in the subfolder `ExcelFiles`:
  - `ModelSimulationandCalibrationforKSectorsandRregions.xlsx` has multiple sheets:
    - initial `Start`
    - terminal `Terminal`
    - parameters to define rigidity parameters `Dynamics`
    - elasticity parameters and tax rates `Structural Parameters`
    - coefficients for regional and sector specific damage functions `Climate Damage Functions (Labour, Capital, TFP)`
    - `Baseline` scenario and other optional scenario sheets `Adaptation and Extremes` defining paths for exogenous variables
    - `Data` to load external data sources

4.2 Mod Files

- We will now define and implement all model equations into Dynare.
- For each Equation we will explain its derivation and how to implement it.
- The folder ModFiles contains the file `DGE_CRED_Model_Equations.mod`.
- `DGE_CRED_Model_Equations.mod` contains all model equations.
- We need to use the symbols defined in `DGE_CRED_Model_Declarations.mod`.

4.3 Households

- Representative households h providing labour N and capital K to domestic firms f
- Maximize discounted utility over an infinite horizon by choosing consumption $C_t(h)$, capital $K_{k,r,t+1}(h)$, investments $I_{k,r,t}(h)$, labour $N_{k,r,t}(h)$ and foreign net wealth B_{t+1}
- Optimization problem of the representative household is

$$\max_{C_t(h),\, K_{k,r,t+1}(h),\, I_{k,r,t}(h),,\, N_{k,r,t}(h),\, B_{t+1}} \sum_{t=0}^{\infty} \beta^t \left(\frac{C_t(h)^{1-\sigma^C}}{1-\sigma^C} - \sum_{k=1}^{K} \sum_{r=1}^{R} A^N_{k,r,t}\, \phi^L_{k,r} \frac{N_{k,r,t}(h)^{1+\sigma^L}}{1+\sigma^L} \right)$$

$$\text{s.t.} P_t\, C_t(h)\,(1+\tau^C) + \sum_{k=1}^{K} \sum_{r=1}^{R} P_{k,r,t} I_{k,r,t}(h) + B_{t+1}(h) =$$

$$\sum_{k=1}^{K} \sum_{r=1}^{R} (1-\tau^N)\, W_{k,r,t} N_{k,r,t}(h) + \sum_{k=1}^{K} \sum_{r=1}^{R} P_{k,r,t}\, r_{k,r,t}\,(1-\tau^K)\, K_{k,r,t}(h) + S^f_t\, \phi^B_t\,(1+r^f_t)\, B_t(h)$$

4.3 Households Lagrangian

- We set-up the Lagrangian for the optimization problem to derive the first order conditions.

$$
\begin{aligned}
&\sum_{t=0}^{\infty}\beta^t\Bigg[\left(\frac{C_t(h)^{1-\sigma^C}}{1-\sigma^C}-\sum_{k=1}^{K}\sum_{r=1}^{R}A^N_{k,r,t}\,\phi^L_{k,r}\frac{N_{k,r,t}(h)^{1+\sigma^L}}{1+\sigma^L}\right)\\
&-\lambda_t(h)\Big(P_t\,C_t(h)\,(1+\tau^C)+\sum_{k=1}^{K}\sum_{r=1}^{R}P_{k,r,t}I_{k,r,t}(h)+B_{t+1}(h)-\sum_{k=1}^{K}\sum_{r=1}^{R}(1-\tau^N)\,W_{k,r,t}N_{k,r,t}(h)\\
&-\sum_{k=1}^{K}\sum_{r=1}^{R}P_{k,r,t}\,r_{k,r,t}\,(1-\tau^K)\,K_{k,r,t}(h)-S^f_t\,\phi^B_t\,(1+r^f_t)\,B_t(h)\Big)\\
&-\sum_{k=1}^{K}\sum_{r=1}^{R}\lambda_t(h)\omega^I_{k,r,t}(h)\left\{K_{k,r,t+1}-(1-\delta-D^K_{k,r,t})\,K_{k,r,t}-I_{k,r,t}\,\Gamma\left(\frac{I_{k,r,t}}{I_{k,r,t-1}}\right)\right\}\Bigg].
\end{aligned}
$$

4.3 Households First Order Conditions – Intratemporal

- Marginal utility of consumption

$$\lambda_t = \frac{C_t(h)^{-\sigma^C}}{P_t\,(1+\tau^C)}$$

- Labour supply curve

$$\phi^L_{k,r}\,A^N_{k,r,t}\,N_{k,r,t}(h)^{\sigma^L} = \lambda_t(h)\,W_{k,r,t}\,(1-\tau^N)$$

4.3 Implementation of Households First Order Conditions – Intratemporal

```
model;
...
    @# for sec in 1:Sectors
        @# for reg in 1:Regions
        ...
        [name = 'HH FOC labour',mcp = 'N_@{sec}_@{reg}>0']
        (1 - tauN_p)*W_@{sec}_@{reg}*(P*C/PoP)^(-sigmaC_p)/(1+tauC_p) =
        A_N_@{sec}_@{reg}*phiL_@{sec}_@{reg}_p*(N_@{sec}_@{reg})^(sigmaL_p);
        ...
        @# endfor // end loop over regions
    @# endfor // end loop over sectors
...
end;
```

4.3 Households First Order Conditions – Intertemporal

- Euler equation for capital

$$\lambda_{t+1}(h)\,\beta\,\left(P_{k,r,t+1}\,r_{k,r,t+1}\,(1-\tau^K)+(1-\delta-D^K_{k,r,t+1})\,\omega^I_{k,r,t+1}\right)=\lambda_t(h)\,\omega^I_{k,r,t}.$$

- Euler equation for investment

$$P_{k,r,t}\,\lambda_t(h)=\lambda_t(h)\,\omega^I_{k,r,t}\left(\Gamma(\frac{I_{k,r,t}}{I_{k,r,t-1}})+\frac{\partial\Gamma(\frac{I_{k,r,t}}{I_{k,r,t-1}})}{\partial(\frac{I_{k,r,t}}{I_{k,r,t-1}})}\,\frac{I_{k,r,t}}{I_{k,r,t-1}}\right)-\beta\lambda_{t+1}(h)\,\omega^I_{k,r,t+1}\,\frac{\partial\Gamma(\frac{I_{k,r,t+1}}{I_{k,r,t}})}{\partial(\frac{I_{k,r,t+1}}{I_{k,r,t}})}\left(\frac{I_{k,r,t+1}}{I_{k,r,t}}\right)^2$$

- Investment adjustment cost

$$\Gamma(\frac{I_{k,r,t}}{I_{k,r,t-1}})=3-exp\left\{\sqrt{\phi^K/2}\left(\frac{I_{k,r,t}}{I_{k,r,t-1}}-1\right\}\right)-exp\left\{-\sqrt{\phi^K/2}\left(\frac{I_{k,r,t}}{I_{k,r,t-1}}-1\right)\right\}$$

4.3 Implementation of Households First Order Conditions – Intertemporal (1)

```
...
# Gamma_@{sec}_@{reg} = (3 - (exp(sqrt(phiK_p/2)*(I_@{sec}_@{reg}/I_@{sec}_@{reg}(-1)-1))
                        +exp(-sqrt(phiK_p/2)*(I_@{sec}_@{reg}/I_@{sec}_@{reg}(-1)-1))));

# Gammapr_@{sec}_@{reg} = -I_@{sec}_@{reg}/I_@{sec}_@{reg}(-1)*sqrt(phiK_p/2)*
                          (exp(sqrt(phiK_p/2)*(I_@{sec}_@{reg}/I_@{sec}_@{reg}(-1)-1))
                          -exp(-sqrt(phiK_p/2)*(I_@{sec}_@{reg}/I_@{sec}_@{reg}(-1)-1)));

# Gammaprp1_@{sec}_@{reg} = I_@{sec}_@{reg}(+1)^2/(I_@{sec}_@{reg})^2*sqrt(phiK_p/2)*
                            (exp(sqrt(phiK_p/2)*(I_@{sec}_@{reg}(+1)/I_@{sec}_@{reg}-1))
                            -exp(-sqrt(phiK_p/2)*(I_@{sec}_@{reg}(+1)/I_@{sec}_@{reg}-1)));

# Gammap1_@{sec}_@{reg} = (3 - (exp(sqrt(phiK_p/2)*(I_@{sec}_@{reg}/I_@{sec}_@{reg}(-1)-1))
                          +exp(-sqrt(phiK_p/2)*(I_@{sec}_@{reg}/I_@{sec}_@{reg}(-1)-1))));
...
```


giz Deutsche Gesellschaft für Internationale Zusammenarbeit (GIZ) GmbH

4.3 Implementation of Households First Order Conditions – Intertemporal (2)

```
...
[name = 'HH FOC capital',mcp = 'K_@{sec}_@{reg}>0']
(C(+1)/PoP(+1))^(-sigmaC_p)/(P(+1)*(1+tauC_p))*beta_p*r_@{sec}_@{reg}(+1)*P_@{sec}_@{reg}(+1)*(1 - tauK_p)
+ (C(+1)/PoP(+1))^(-sigmaC_p)/(P(+1)*(1+tauC_p))*omegaI_@{sec}_@{reg}(+1)
* beta_p*(1 - delta_p - D_K_@{sec}_@{reg}(+1))
= omegaI_@{sec}_@{reg}*(C/PoP)^(-sigmaC_p)/(P*(1+tauC_p));

[name = 'HH FOC investment',mcp = 'I_@{sec}_@{reg} > 0']
(C/PoP)^(-sigmaC_p)/(P*(1+tauC_p))*P_@{sec}_@{reg} =
(C/PoP)^(-sigmaC_p)/(P*(1+tauC_p))*omegaI_@{sec}_@{reg}*(Gamma_@{sec}_@{reg}+Gammapr_@{sec}_@{reg})
+ beta_p*(C(+1)/PoP(+1))^(-sigmaC_p)/(P(+1)*(1+tauC_p))*omegaI_@{sec}_@{reg}(+1)*Gammaprp1_@{sec}_@{reg};

[name = 'LOM capital',mcp = 'I_@{sec}_@{reg} > 0']
K_@{sec}_@{reg} = (1 - delta_p - D_K_@{sec}_@{reg})*K_@{sec}_@{reg}(-1)+I_@{sec}_@{reg}*Gamma_@{sec}_@{reg};
...
```

4.3 Rest of the world

- Euler equation foreign bonds

$$\lambda_{t+1}\,\beta\,S^f_{t+1}\,\phi^B_{t+1}\left(1+{r^f}_{t+1}\right)=\lambda_t$$

- Effective exchange rate S^f and the world interest rate r^f.
- The required interest rate is above the world interest rate if the foreign debt ($B_{t+1}<0$)/ foreign claims ($B_{t+1}>0$) relative to GDP increases/decreases and future net exports relative to GDP will decrease.

$$\phi^B_{t+1}=exp\left(-\phi^B\,(S^f_{t+1}\,r^f_{t+1}\,\frac{B_{t+1}}{Y_{t+1}}+\frac{NX_{t+1}}{Y_{t+1}})\right)$$

4.3 Implementation Rest of the World

```
...
# phiB = exp(-phiB_p*((Sf(+1)*rf(+1)*B/Y(+1)+NX(+1)/Y(+1))))

[name = 'Foreign Assets']
(C(+1)/PoP(+1))^(-sigmaC_p)/(P(+1)*(1+tauC_p))*beta_p*Sf(+1)*phiB*(1+rf(+1))
= (C/PoP)^(-sigmaC_p)/(P*(1+tauC_p));
...
```

4.3 Government Budget Constraint

- We are interested in different policy measures taken by the government to adapt to a new climate regime.
- Government behaviour is not a result of an optimization problem.

$$G_t + \sum_k^K \sum_r^R \sum_z^Z G^{A,z}_{k,r,t} + B^G_{t+1} = \sum_k^K \sum_r^R \left\{ \left(\tau^K + \tau^K_{r,k,t}\right) P_{k,r,t}\, r_{k,r,t}\, K_{k,r,t} + \left(\tau^N + \tau^N_{k,r,t}\right) W_{k,r,t}\, N_{k,r,t}\, Pop_t \right\}$$
$$+ \left(1 + r^f_t\right) S^f_t \phi^B_t\, B^G_t$$

4.3 Implementation of Government Budget Constraint

```
...
[name = 'Government Budget Constraint']
G+BG
@# for sec in 1:Sectors
    @# for reg in 1:Regions
        @# for z in ["T", "WS", "PREC", "SL", "CYC", "DRO"]
            + G_A_@{z}_@{sec}_@{reg}+G_A_PREC_@{sec}_@{reg}
        @# endfor
    @# endfor
@# endfor
= (1+rf)*Sf*exp(-phiB_p*((Sf*rf*B(-1)/Y+NX/Y)))*BG(-1)+tauC_p*C
@# for sec in 1:Sectors
    @# for reg in 1:Regions
        +(tauN_p+tauN_@{sec}_@{reg})*W_@{sec}_@{reg}*N_@{sec}_@{reg}*PoP/P
        +(tauK_p+tauK_@{sec}_@{reg})*P_@{sec}_@{reg} /P*r_@{sec}_@{reg}*K_@{sec}_@{reg}
    @# endfor
@# endfor
;

[name = 'Government Budget Deficit']
BG = exo_BG;
...
```

4.3 Government Policy Instruments

- Governments can invest into adaptation capital stocks

$$K_{k,r,t+1}^{A,z} = \eta_{k,r,t}^{A,z}$$

- Evolution of adaptation capital stocks

$$K_{k,r,t+1}^{A,z} = (1 - \delta_{K^{A,z},k,r})\, K_{k,r,t}^{A,z} + G_{k,r,t}^{A,z}$$

- Tax on capital expenditures paid by firms

$$\tau_{k,r,t}^{K} = \tau_{k,r,0}^{K} + \eta_{k,r,t}^{\tau^K}$$

- Tax rate on wage bill paid by firms

$$\tau_{k,r,t}^{N} = \tau_{k,r,0}^{N} + \eta_{k,r,t}^{\tau^N}$$

4.3 Implementation of Government Policy Instruments

```
...
@# for z in ["T", "WS", "PREC", "SL", "CYC", "DRO"]
 [name = 'sector specific adaptation expenditures by the government against sea level rise']
 K_A_@{z}_@{sec}_@{reg} = exo_GA_@{z}_@{sec}_@{reg};

 [name = 'sector specific adaptation capital against sea level rise']
 K_A_@{z}_@{sec}_@{reg} = (1-deltaKA@{z}_@{sec}_@{reg}_p)*K_A_@{z}_@{sec}_@{reg}(-1)+G_A_@{z}_@{sec}_@{reg};
@# endfor
[name = 'sector specific corporate tax rate paid by firms']
tauK_@{sec}_@{reg} = tauK_@{sec}_@{reg}_p + exo_tauK_@{sec}_@{reg};

[name = 'sector specific labour tax rate paid by firms']
tauN_@{sec}_@{reg} = tauN_@{sec}_@{reg}_p + exo_tauN_@{sec}_@{reg};
...
```

4.3 Resource constraint

- Households and government use domestic final goods Y_t produced by firms for consumption, investment and for exports X_t and can also use imports M_t for consumption and investment

$$Y_t = C_t + I_t + G_t + \underbrace{X_t - M_t}_{NX_t} \tag{1}$$

- The aggregation of the budget constraints of the representative households also states that positive net exports are used to increase net financial wealth to the rest of the world.

$$NX_t = B_{t+1} - (1 + r_t^f) S_t^f \phi_t^B B_t \tag{2}$$

4.3 Implementation of Resource Constraint

```
...
[name = 'Resource Constraint']
Y = C + I + G + NX
@# for sec in 1:Sectors
    @# for reg in 1:Regions
                    @# for z in ["T", "WS", "PREC", "SL", "CYC", "DRO"]
          + G_A_@{z}_@{sec}_@{reg}
                    @# endfor
    @# endfor
@# endfor
;

[name = 'Net Exports']
NX = (B - (1 + rf) * exp(-phiB_p*((Sf*rf*B(-1)/Y+NX/Y))) * Sf * B(-1));
...
```

4.4 Sectoral Decomposition

- Final domestic goods Y_t are created combining goods from different sectors $Y_{k,t}$ using a CES production function.

$$\min_{Y_{k,t}} \sum_k Y_{k,t} P_{k,t} \tag{3}$$

$$Y_t = \left(\sum_k \omega_k^{Q\,\frac{1}{\eta^Q}} Y_{k,t}^{\frac{\eta^Q - 1}{\eta^Q}} \right)^{\frac{\eta^Q}{\eta^Q - 1}} \tag{4}$$

- Therefore, the demand for sectoral products correspond to the first order conditions of the above optimization problem.

$$\frac{P_{k,t}}{P_t} = \omega_k^{Q\,\frac{1}{\eta^Q}} \left(\frac{Y_{k,t}}{Y_t} \right)^{\frac{-1}{\eta^Q}}$$

4.4 Implementation of Sectoral Decomposition

```
...
[name = 'demand for sector output']
P_@{sec}/P = omegaQ_@{sec}_p^(1/etaQ_p)*(Y_@{sec}/Y)^(-1/etaQ_p);
...
```

```
[name = 'aggregate gross value added']
P*Y =
@# for sec in 1:Sectors
    + P_@{sec}*Y_@{sec}
@# endfor
;
```

4.4 Regional Decomposition

- In order to model regional economic activity we further decompose the production process on a regional level.

$$\min_{Y_{k,r,t}} \sum_k Y_{k,r,t} P_{k,r,t}$$

$$Y_{k,t} = \left(\sum_k \omega_{k,r}^{Q\,\frac{1}{\eta_k^Q}} Y_{k,r,t}^{\frac{\eta_k^Q - 1}{\eta_k^Q}} \right)^{\frac{\eta_k^Q}{\eta_k^Q - 1}}$$

- Demand for sectoral and regional products correspond to the first order conditions of the above optimization problem.

$$\frac{P_{k,r,t}}{P_{k,t}} = \omega_{k,r}^{Q\,\frac{1}{\eta_k^Q}} \left(\frac{Y_{k,r,t}}{Y_{k,t}} \right)^{\frac{-1}{\eta_k^Q}}$$

4.4 Implementation of Regional Decomposition

```
...
[name = 'demand for regional sector output',mcp = 'Y_@{sec}_@{reg} > 0']
P_@{sec}_@{reg} = omegaQ_@{sec}_@{reg}_p^(1/etaQ_@{sec}_p)
                  *((Y_@{sec}_@{reg})/Y_@{sec})^(-1/etaQ_@{sec}_p)*P_@{sec};
...
```

```
...
[name = 'sector aggregate specific output']
Y_@{sec} = (
          @# for reg in 1:Regions
             +omegaQ_@{sec}_@{reg}_p^(1/etaQ_@{sec}_p)*((Y_@{sec}_@{reg}))^((etaQ_@{sec}_p-1)/etaQ_@{sec}_p)
          @# endfor
           )^(etaQ_@{sec}_p/(etaQ_@{sec}_p-1));
...
```

4.4 Regional Production

- At the regional and sectoral level are representative firms maximizing profits using capital $K_{k,r,t}$ and labour $L_{k,r,t} = N_{k,r,t}\, Pop_t$ provided by households to produce products.
- They charge a price $P_{k,r,t}$ for their products and have to pay households wages $W_{k,r,t}$, interest on rented capital $P_{r,k,t}\, r_{r,k,t}$, taxes related to the wage bill $\tau^N_{r,k,t}$ and on capital expenditure $\tau^K_{r,k,t}$.
- Representative firms have access to a regional and sector specific constant elasticity of substitution production function.
- The productivity of capital and labour of a firm in one sector and region depends on the climate variables, and the adaption measures by the government represented by a damage function affecting total factor productivity $A_{k,r,t}$ by $D_{k,r,t} = D_{k,r}\left(T_{r,t},\ PREC_{r,t},\ WS_{r,t},\ SL_{r,t},\ CYC_{r,t},\ DRO_{r,t},\ G^A_{r,k,t}\right)$.
- Further, we explicitly differentiate between climate induced damages affecting labour productivity $D_{N,k,r,t}$ and capital depreciation $D_{K,k,r,t}$.
- As in Nordhaus 1993, we assume a polynomial functional form of the damage functions, but the damages are different across regions and sectors.

4.4 Damages on TFP

$$D_{k,r\,t} = \Big\{$$

$$\underbrace{(a_{T,1,k,r}\,T_{r\,t} + a_{T,2,k,r}\,(T_{r\,t})^{a_{T,3,k,r}})}_{\text{impact of temperature}}\ \underbrace{exp(-\phi^{G^{A,T}}_{k,r}\ K^{A,T}_{k,r,t})}_{\text{impact of adaptation}} + \underbrace{(a_{SL,1,k,r}\,SL_t + a_{SL,2,k,r}\,(SL_t)^{a_{SL,3,k,r}})}_{\text{impact of sea level}}\ \underbrace{I(SL > \frac{K^{A,SL}_{k,r,t}}{\phi^{G^{A,SL}}_{k,r}})}_{\text{impact of adaptation}}$$

$$+ \underbrace{(a_{WS,1,k,r}\,WS_{r\,t} + a_{WS,2,k,r}\,(WS_{r\,t})^{a_{WS,3,k,r}})}_{\text{impact of wind speed}}\ \underbrace{exp(-\phi^{G^{A,WS}}_{k,r}\ K^{A,WS}_{k,r,t})}_{\text{impact of adaptation}}$$

$$+ \underbrace{(a_{PREC,1,k,r}\,PREC_{r\,t} + a_{PREC,2,k,r}\,(PREC_{r\,t})^{a_{PREC,3,k,r}})}_{\text{impact of precipitation}}\ \underbrace{exp(-\phi^{G^{A,PREC}}_{k,r}\ K^{A,PREC}_{k,r,t})}_{\text{impact of adaptation}}$$

$$+ \underbrace{(a_{CYC,1,k,r}\,CYC_{r\,t} + a_{CYC,2,k,r}\,(CYC_{r\,t})^{a_{CYC,3,k,r}})}_{\text{impact of cyclones}}\ \underbrace{exp(-\phi^{G^{A,CYC}}_{k,r}\ K^{A,CYC}_{k,r,t})}_{\text{impact of adaptation}}$$

$$+ \underbrace{(a_{DRO,1,k,r}\,DRO_{r\,t} + a_{DRO,2,k,r}\,(DRO_{r\,t})^{a_{DRO,3,k,r}})}_{\text{impact of droughts}}\ \underbrace{exp(-\phi^{G^{A,DRO}}_{k,r}\ K^{A,DRO}_{k,r,t})}_{\text{impact of adaptation}}$$

$$\Big\}.$$

4.4 Implementation of Damages on TFP

```
...
[name = 'sector specific damage function']
D_@{sec}_@{reg} = min(0.7,
(a_T_1_@{sec}_@{reg}_p * T_@{reg} + a_T_2_@{sec}_@{reg}_p * T_@{reg}^(a_T_3_@{sec}_@{reg}_p))
 * exp(-phiGAT_@{sec}_@{reg}_p*K_A_T_@{sec}_@{reg}(-1)) +
(a_SL_1_@{sec}_@{reg}_p * SL + a_SL_2_@{sec}_@{reg}_p * SL^(a_SL_3_@{sec}_@{reg}_p))
 * (SL > (K_A_SL_@{sec}_@{reg}(-1) / phiGASL_@{sec}_@{reg}_p)) +
(a_W_1_@{sec}_@{reg}_p * WS_@{reg} + a_W_2_@{sec}_@{reg}_p * WS_@{reg}^(a_W_3_@{sec}_@{reg}_p))
 * exp(-phiGAWS_@{sec}_@{reg}_p*K_A_WS_@{sec}_@{reg}(-1)) +
(a_P_1_@{sec}_@{reg}_p * PREC_@{reg} + a_P_2_@{sec}_@{reg}_p * PREC_@{reg}^(a_P_3_@{sec}_@{reg}_p))
 * exp(-phiGAPREC_@{sec}_@{reg}_p*K_A_PREC_@{sec}_@{reg}(-1)) +
(a_DR_1_@{sec}_@{reg}_p * DRO_@{reg} + a_DR_2_@{sec}_@{reg}_p * DRO_@{reg}^(a_DR_3_@{sec}_@{reg}_p))
 * exp(-phiGADRO_@{sec}_@{reg}_p*K_A_DRO_@{sec}_@{reg}(-1)) +
(a_CY_1_@{sec}_@{reg}_p * CYC_@{reg} + a_CY_2_@{sec}_@{reg}_p * CYC_@{reg}^(a_CY_3_@{sec}_@{reg}_p))
 * exp(-phiGACYC_@{sec}_@{reg}_p*K_A_CYC_@{sec}_@{reg}(-1))
                                                    );
...
```

4.4 Damages on Labour Productivity

$$
\begin{aligned}
D^N_{k,r\,t} = \Big(& \\
& \underbrace{a^N_{T,1,k,r}\, T_{r\,t} + a^N_{T,2,k,r}\, (T_{r\,t})^{a^N_{T,3,k,r}}}_{\text{impact of temperature}} + \underbrace{a^N_{SL,1,k,r}\, SL_t + a^N_{SL,2,k,r}\, (SL_t)^{a^N_{SL,3,k,r}}}_{\text{impact of sea level}} \\
& + \underbrace{a^N_{WS,1,k,r}\, WS_{r\,t} + a^N_{WS,2,k,r}\, (WS_{r\,t})^{a^N_{WS,3,k,r}}}_{\text{impact of wind speed}} + (\underbrace{a^N_{PREC,1,k,r}\, PREC_{r\,t} + a^N_{PREC,2,k,r}\, (PREC_{r\,t})^{a^N_{PREC,3,k,r}})}_{\text{impact of precipitation}} \\
& + \underbrace{a^N_{CYC,1,k,r}\, CYC_{r\,t} + a^N_{CYC,2,k,r}\, (CYC_{r\,t})^{a^N_{CYC,3,k,r}}}_{\text{impact of cyclones}} + \underbrace{a^N_{DRO,1,k,r}\, DRO_{r\,t} + a^N_{DRO,2,k,r}\, (DRO_{r\,t})^{a^N_{DRO,3,k,r}}}_{\text{impact of droughts}} \\
& \Big).
\end{aligned}
$$

4.4 Implementation of Damages on Labour Productivity

```
...
[name = 'sector specific damage function on labour productivity ']
D_N_@{sec}_@{reg} = min(1,aN_T_1_@{sec}_@{reg}_p * T_@{reg} + aN_T_2_@{sec}_@{reg}_p * T_@{reg}^(aN_T_3_@{sec}
      _@{reg}_p) +
aN_SL_1_@{sec}_@{reg}_p * SL + aN_SL_2_@{sec}_@{reg}_p * SL^(aN_SL_3_@{sec}_@{reg}_p) +
aN_W_1_@{sec}_@{reg}_p * WS_@{reg} + aN_W_2_@{sec}_@{reg}_p * WS_@{reg}^(aN_W_3_@{sec}_@{reg}_p) +
aN_P_1_@{sec}_@{reg}_p * PREC_@{reg} + aN_P_2_@{sec}_@{reg}_p * PREC_@{reg}^(aN_P_3_@{sec}_@{reg}_p) +
aN_DR_1_@{sec}_@{reg}_p * DRO_@{reg} + aN_DR_2_@{sec}_@{reg}_p * DRO_@{reg}^(aN_DR_3_@{sec}_@{reg}_p) +
aN_CY_1_@{sec}_@{reg}_p * CYC_@{reg} + aN_CY_2_@{sec}_@{reg}_p * CYC_@{reg}^(aN_CY_3_@{sec}_@{reg}_p)
);
...
```

4.4 Damages on Capital

$$
\begin{aligned}
D^K_{k,r\,t} = \Big(& \\
& \underbrace{a^K_{T,1,k,r}\, T_{r\,t} + a^K_{T,2,k,r}\, (T_{r\,t})^{a^K_{T,3,k,r}}}_{\text{impact of temperature}} + \underbrace{a^K_{SL,1,k,r}\, SL_t + a^K_{SL,2,k,r}\, (SL_t)^{a^K_{SL,3,k,r}}}_{\text{impact of sea level}} \\
& + \underbrace{a^K_{WS,1,k,r}\, WS_{r\,t} + a^K_{WS,2,k,r}\, (WS_{r\,t})^{a^K_{WS,3,k,r}}}_{\text{impact of wind speed}} + \underbrace{a^K_{PREC,1,k,r}\, PREC_{r\,t} + a^K_{PREC,2,k,r}\, (PREC_{r\,t})^{a^K_{PREC,3,k,r}}}_{\text{impact of precipitation}} \\
& + \underbrace{a^K_{CYC,1,k,r}\, CYC_{r\,t} + a^K_{CYC,2,k,r}\, (CYC_{r\,t})^{a^K_{CYC,3,k,r}}}_{\text{impact of cyclones}} + \underbrace{a^K_{DRO,1,k,r}\, DRO_{r\,t} + a^K_{DRO,2,k,r}\, (DRO_{r\,t})^{a^K_{DRO,3,k,r}}}_{\text{impact of droughts}} \\
& \Big).
\end{aligned}
$$

4.4 Implementation of Damages on Capital

```
...
[name = 'sector specific damage function on capital formation']
D_K_@{sec}_@{reg} = min(1,aK_T_1_@{sec}_@{reg}_p*T_@{reg}+aK_T_2_@{sec}_@{reg}_p*T_@{reg}^(aK_T_3_@{sec}_@{reg
      }_p)+
aK_SL_1_@{sec}_@{reg}_p*SL+aK_SL_2_@{sec}_@{reg}_p*SL^(aK_SL_3_@{sec}_@{reg}_p)+
aK_W_1_@{sec}_@{reg}_p*WS_@{reg}+aK_W_2_@{sec}_@{reg}_p*WS_@{reg}^(aK_W_3_@{sec}_@{reg}_p)+
aK_P_1_@{sec}_@{reg}_p*PREC_@{reg}+aK_P_2_@{sec}_@{reg}_p*PREC_@{reg}^(aK_P_3_@{sec}_@{reg}_p)+
aK_DR_1_@{sec}_@{reg}_p*DRO_@{reg}+aK_DR_2_@{sec}_@{reg}_p*DRO_@{reg}^(aK_DR_3_@{sec}_@{reg}_p)+
aK_CY_1_@{sec}_@{reg}_p*CYC_@{reg}+aK_CY_2_@{sec}_@{reg}_p*CYC_@{reg}^(aK_CY_3_@{sec}_@{reg}_p)
);
...
```

4.4 Profit Maximization

- Firms in each region and sector have access to a constant elasticity of substitution production function with production factors labour and capital.

$$\max_{Y_{k,r,t},N_{k,r,t},K_{k,r,t}} P_{k,r,t}\, Y_{k,r,t} - W_{k,r,t}\, N_{k,r,t}\, Pop_t\, (1+\tau^N_{k,r,t}) - r_{k,r,t}\, P_{k,r,t}\, K_{k,r,t}\, (1+\tau^K_{k,r,t})$$

$$\text{s.t. } Y_{k,r,t} = A_{k,r,t}(1-D_{k,r,t}) \left[{\alpha^N_{k,r}}^{\frac{1}{\eta^{NK}_{k,r}}} \left(A^N_{k,r,t}\, (1-D^N_{k,r,t})\, Pop_t\, N_{k,r,t} \right)^{\rho_{k,r}} + {\alpha^K_{k,r}}^{\frac{1}{\eta^{NK}_{k,r}}} \left(K_{k,r,t} \right)^{\rho_{k,r}} \right]^{\frac{1}{\rho_{k,r}}},$$

$$\text{where } \rho_{k,r} = \frac{\eta^{NK}_k - 1}{\eta^{NK}_k}.$$

4.4 Factor Demand

- Demand for production factors are given by the first order condition of the above optimization problem. The Lagrange multiplier is equal to the price charged by companies.

$$\frac{W_{k,r,t}}{P_{k,r,t}}\left(1+\tau^N_{k,r,t}\right) = {\alpha^N_{k,r}}^{\frac{1}{\eta^{NK}_{k,r}}} \left(A_{k,r,t}\left(1-D_{k,r,t}\right) A^N_{k,r,t}\left(1-D^N_{k,r,t}\right)\right)^{\rho_{k,r}} \left(\frac{Pop_t N_{k,r,t}}{Y_{k,r,t}}\right)^{-\frac{1}{\eta^{NK}_{k,r}}}$$

$$r_{k,r,t}\left(1+\tau^K_{k,r,t}\right) = {\alpha^K_{k,r}}^{\frac{1}{\eta^{NK}_{k,r}}} \left(A_{k,r,t}\left(1-D_{k,r,t}\right)\right)^{\rho_{k,r}} \left(\frac{K_{k,r,t}}{Y_{k,r,t}}\right)^{-\frac{1}{\eta^{NK}_{k,r}}}$$

- We use the more general case of the CES production function rather than the more commonly used Cobb-Douglas production function.
- The parameter $\eta^{NK}_{k,r}$ allows us to control the response of capital and labour demand to temporary productivity shocks.
- Temporary productivity shocks are in our set-up also weather extremes.

4.4 Implementation of Factor Demand

```
...
[name='sector specific output']
Y_@{sec}_@{reg} = (1-D_@{sec}_@{reg})*A_@{sec}_@{reg}*(alphaK_@{sec}_@{reg}_p^(1/etaNK_@{sec}_@{reg}_p)*(A_K_@
      {sec}_@{reg}*K_@{sec}_@{reg}(-1))^((etaNK_@{sec}_@{reg}_p-1)/etaNK_@{sec}_@{reg}_p)+(alphaN_@{sec}_@{reg}
      _p)^(1/etaNK_@{sec}_@{reg}_p)*((1-D_N_@{sec}_@{reg})*A_N_@{sec}_@{reg}*PoP*N_@{sec}_@{reg})^((etaNK_@{sec
      }_@{reg}_p-1)/etaNK_@{sec}_@{reg}_p))^(etaNK_@{sec}_@{reg}_p/(etaNK_@{sec}_@{reg}_p-1));

[name = 'Firms FOC capital',mcp = 'K_@{sec}_@{reg} > 0']
r_@{sec}_@{reg}*(1+tauK_@{sec}_@{reg}) = alphaK_@{sec}_@{reg}_p^(1/etaNK_@{sec}_@{reg}_p)*((1-D_@{sec}_@{reg})
      *A_@{sec}_@{reg})^((etaNK_@{sec}_@{reg}_p-1)/(etaNK_@{sec}_@{reg}_p))*A_K_@{sec}_@{reg}^((etaNK_@{sec}_@{
      reg}_p-1)/(etaNK_@{sec}_@{reg}_p))*(K_@{sec}_@{reg}(-1)/Y_@{sec}_@{reg})^(-1/etaNK_@{sec}_@{reg}_p);

[name = 'Firms FOC labour',mcp = 'N_@{sec}_@{reg} > 0']
W_@{sec}_@{reg}*(1+tauN_@{sec}_@{reg})/P_@{sec}_@{reg} = alphaN_@{sec}_@{reg}_p^(1/etaNK_@{sec}_@{reg}_p)*((1-
      D_N_@{sec}_@{reg})*A_N_@{sec}_@{reg}*(1-D_@{sec}_@{reg})*A_@{sec}_@{reg})^((etaNK_@{sec}_@{reg}_p-1)/(
      etaNK_@{sec}_@{reg}_p))*((PoP*N_@{sec}_@{reg})/Y_@{sec}_@{reg})^(-1/etaNK_@{sec}_@{reg}_p);
...
```

4.4 Trade with the Rest of the World

- The demand for domestic exports and foreign imports is not explicitly modeled in this version of the model.
- We assume that net exports follow an auto-regressive process of order one and that the long-run value of net exports depend on the long-run development of gross domestic product.

$$NX_t = \rho^{NX} NX_{t-1} + (1 - \rho^{NX})\omega^{NX} P_t \, Y_t \, exp\,(\eta_{NX\,t})$$

- The effective exchange rate S_t^f and the world interest rate r_t^f determine how much governments and households have to pay back in domestic currency as net lender or how much they receive as net borrower to the rest of the world.
- The world interest rate is independent of domestic developments and only the effective exchange rate adjusts.

4.4 Implementation of Trade with Rest of the World

```
...
[name = 'LOM Net Exports']
NX = rhoNX_p * NX(-1) + (1 - rhoNX_p) * exp(exo_NX) * omegaNX_p * Y * P;

[name = 'World interest rate']
rf = (1/beta_p-1);
...
```

4.5 Climate Variables

- Average annual regional temperature is given by:

$$T_{r,t} = T_{r,0} + \eta_t^{T}$$

- Average annual regional wind speed is given by:

$$WS_{r,t} = WS_{r,0} + \eta_t^{WS}$$

- Average annual regional precipitation is given by:

$$PREC_{r,t} = PREC_{r,0} + \eta_t^{PREC}$$

- Sea level:

$$SL_t = SL_0 + \eta_t^{SL}$$

4.5 Implementation of Climate Variables

```
...
[name = 'Temperature']
T_@{reg} = T0_@{reg}_p + exo_T_@{reg};

[name = 'Wind speed']
WS_@{reg} = WS0_@{reg}_p + exo_WS_@{reg};

[name = 'Precipitation']
PREC_@{reg} = PREC0_@{reg}_p + exo_PREC_@{reg};

[name = 'Sea Level']
SL = SL0_p + exo_SL;
...
```

4.5 Weather Extremes

- A score reflecting the intensity and number of annual droughts in the region.

$$DRO_{r,t} = DRO_{r,0} + \eta_t^{DRO}$$

- A score reflecting the intensity and number of annual cyclones in the region.

$$CYC_{r,t} = CYC_{r,0} + \eta_t^{CYC}$$

4.5 Implementation of Weather Extremes

```
...
[name = 'Drought']
DRO_@{reg} = DRO0_@{reg}_p + exo_DRO_@{reg};

[name = 'Cyclones']
CYC_@{reg} = CYC0_@{reg}_p + exo_CYC_@{reg};
...
```

4.6 National Aggregates

- National aggregate investment is given by:

$$P_t\, I_t = \sum_k^K P_{k,t}\, I_{k,t}$$

- National aggregate capital stock is given by:

$$P_t\, K_t = \sum_k^K P_{k,t}\, K_{k,t-1}$$

- National aggregate labour input:

$$N_t = \sum_k^K N_{k,t}$$

4.6 Implementation of National Aggregates

```
...
[name = 'aggregate investment']
P * I =
@# for sec in 1:Sectors
 + P_@{sec} * I_@{sec}
@# endfor
;

[name = 'aggregate capital']
P * K =
@# for sec in 1:Sectors
 + P_@{sec} * K_@{sec}(-1)
@# endfor
;

[name = 'aggregate labour']
N =
@# for sec in 1:Sectors
    + N_@{sec}
@# endfor
;
...
```

4.6 Sectoral Aggregates

- Sector aggregate investment is given by:

$$P_{k,t}\, I_{k,t} = \sum_{r}^{R} P_{k,r,t}\, I_{k,r,t}$$

- Sector aggregate capital stock is given by:

$$P_{k,t}\, K_{k,t} = \sum_{r}^{R} P_{k,r,t}\, K_{k,r,t-1}$$

- Sector aggregate wage bill:

$$W_{k,t} N_{k,t} = \sum_{r}^{R} W_{k,r,t} N_{k,r,t}$$

- Sector aggregate labour input:

$$N_{k,t} = \sum_{r}^{R} N_{k,r,t}$$

4.6 Implementation of Sectoral Aggregates

```
...
[name = 'aggergate sector labour']
N_@{sec} =
@# for reg in 1:Regions
+ N_@{sec}_@{reg}
@# endfor
;
[name = 'aggergate sector labour income']
W_@{sec} * N_@{sec} =
@# for reg in 1:Regions
+ W_@{sec}_@{reg} * N_@{sec}_@{reg}
@# endfor
;
[name = 'aggregate sector investment']
P_@{sec} * I_@{sec} =
@# for reg in 1:Regions
+ P_@{sec}_@{reg} * I_@{sec}_@{reg}
@# endfor
;
[name = 'aggregate capital']
P_@{sec} * K_@{sec} =
@# for reg in 1:Regions
+ P_@{sec}_@{reg} * K_@{sec}_@{reg}
@#
;
```

|· p:125 |

4.6 National Aggregates

- Sector aggregate investment is given by:

$$P_{k,t}\, I_{k,t} = \sum_{r}^{R} P_{k,r,t}\, I_{k,r,t}$$

- Sector aggregate capital stock is given by:

$$P_{k,t}\, K_{k,t} = \sum_{r}^{R} P_{k,r,t}\, K_{k,r,t-1}$$

- Sector aggregate wage bill:

$$W_{k,t} N_{k,t} = \sum_{r}^{R} W_{k,r,t} N_{k,r,t}$$

- Sector aggregate labour input:

$$N_{k,t} = \sum_{r}^{R} N_{k,r,t}$$

4.6 Implementation of Sectoral Aggregates

```
...
[name = 'aggergate sector labour']
N_@{sec} =
@# for reg in 1:Regions
+ N_@{sec}_@{reg}
@# endfor
;
[name = 'aggergate sector labour income']
W_@{sec} * N_@{sec} =
@# for reg in 1:Regions
+ W_@{sec}_@{reg} * N_@{sec}_@{reg}
@# endfor
;
[name = 'aggregate sector investment']
P_@{sec} * I_@{sec} =
@# for reg in 1:Regions
+ P_@{sec}_@{reg} * I_@{sec}_@{reg}
@# endfor
;
[name = 'aggregate capital']
P_@{sec} * K_@{sec} =
@# for reg in 1:Regions
+ P_@{sec}_@{reg} * K_@{sec}_@{reg}
@#
;
```

4.6 Population and Price Level

- Population is independent of other variables in the model and is given by:

$$pop_t = pop_{0,t} + \eta_t^{pop}$$

- National price level is given by:

$$P_t = P_{0,t} + \eta_t^{P}$$

4.6 Implementation of Population and Prices

```
...
[name = 'national price level']
P = P0_p * exp(exo_P);

[name = 'Population']
PoP = PoP0_p + exo_PoP;
...
```

Outline

5. Model Simulation and Calibration
   - Definitions
   - Calibration
   - Scenario Analyses

5.1 Main model file `DGE_CRED_Model.mod`

- Define number of sectors and regions
- Define number of steps in steady state calculation and simulation process
- Define excel files that include:
  - Data used for calibration
  - Explicit parameter values for parameters
  - Assumption related to scenarios
  - Results
- Declare additional `.mod` files needed for simulation and documentation of results

5.1 Definition of Sectors and Regions

- Number of sectors: 3 or 9
  - Agriculture, industry, services
  - Agriculture, manufacturing, construction, transportation and storage, accommodation and food service activities, further production activities, services, state-related sectors and other service activities.
- Number of regions: 2, 3 or 6
  - Coastal (Red River Delta, North Central and Central Coast, Southeast, Mekong river delta) and non-coastal regions
  - Red River Delta, Mekong river delta and remaining
  - All six individually

giz Deutsche Gesellschaft für Internationale Zusammenarbeit (GIZ) GmbH

5.2 Calibration of Parameters

- Regional and sectoral shares of GDP, employment and wages calibrated to match actual data
- Assumptions about a baseline trajectory of the Vietnamese economy (without climate change effects)
  - Projections for population dynamics
  - Long-term growth projections, reflecting general economic catch-up process
  - Speed and scale of structural change (transition to service economy)
- Calibration of structural parameters unaffected by long-term growth and/or climate change

5.2 Calibration of Damage Function Parameters

- Distinct damage functions for every region-sector combination
- Consideration of six climate change phenomena:
  - Temperature (T)
  - Wind Speed (W)
  - Percipitation (P)
  - Sea Level (SL)
  - Drought (DR)
  - Cyclone (CY)
- Each damage function includes parameters measuring
  - Loss of labour productivity
  - Depreciation of capital
  - Loss of TFP

w.r.t. to each of the six climate change phenomena

5.2 Notation of Damage Function Parameters

- Damage function parameters have the general form:

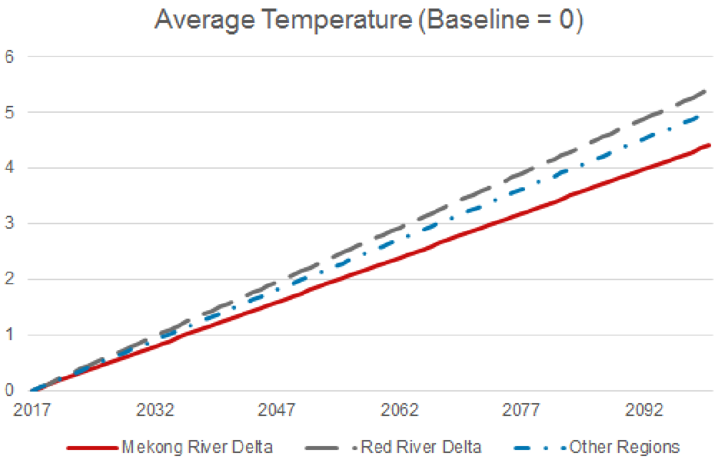
  `f\_c\_d\_s\_r\_p`

  - Input Factor (`f`): `aN`- labour, `aK` - capital, `a` - TFP
  - Climate change phenomenon (`c`): `T` - temperature, `W` - wind speed, `P` - percipitation, `SL` - sea level, `DR` - drought, `CY` - cyclone
  - Degree of polynomial (`d`): `1` - linear, `2` - quadratic, `3` - quadratic exponent
  - Sector (`s`): `1` - agriculture, `2` - industry, `3` - services
  - Region (`r`): `1` - region 1, `2` - region 2, `3` - region 3

5.3 Scenario 1 – „Temperature“

- Assumes that average temperature increases by X degrees in a given region until the year 2100
- Three regions example (Thuc et al, 2016):
  - Mekong river delta: +4.4 degrees
  - Red river delta: +5.4 degrees
  - Other regions 3: +5.0 degrees
- Linear increases in average temperature every year until final value is reached

5.3 Scenario 1 – „Temperature“



Source: DGE–CRED model

5.3 Scenario 1 – „Temperature“

- Damage functions are calibrated to match results of relevant meta studies
- Challinor et al. (2014):
  - Crop yields on average decrease by 4.5% in response to a 1 degree increase in temperature
  - Parameter that governs loss in production from temperature increases (`a_T_1_1_r_p`) is set to 0.045 for all regions
  - Can be set differently for different kinds of crop or, more generally, different regions (to reflect region-specific cultivation)

5.3 Scenario 2 – „Sea Level“

- Increase in average temperature as defined in scenario 1
- In addition, assumes that the sea level rises by 100 cm until the year 2100
- Percentage of agricultural land at risk of inundation following a sea level rise by 100 cm (Thuc et al, 2016)
  - Mekong river delta: 39%
  - Red river delta: 16%
  - Other regions: 2%

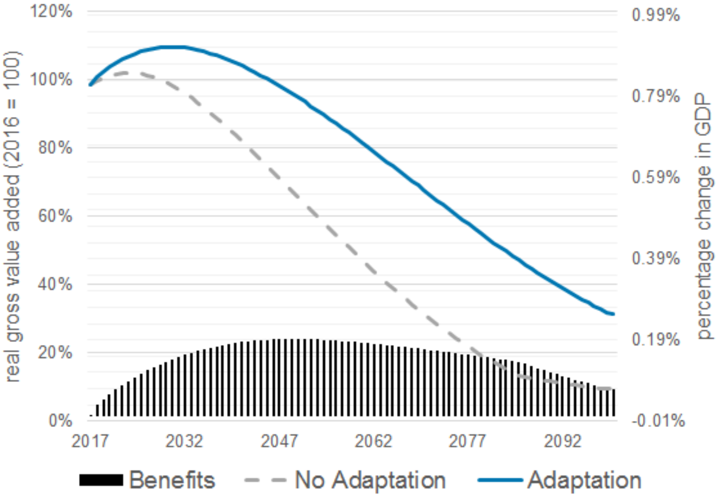
5.3 Scenario 2 – „Sea Level“

- Damage functions are calibrated to match percentage of agricultural land at risk of inundation
- Different calibration for different regions
  - Mekong river delta: `a_SL_1_1_1_p` set to 0.39
  - Red river delta: `a_SL_1_1_2_p` set to 0.16
  - Other regions: `a_SL_1_1_3_p` set to 0.02
- Effects of sea level rise on other sectors of production not yet included
- Can be extended e.g. to transportation (flooded/destroyed roads), energy etc.

5.3 Scenario 3 – „Adaptation“

- Increase in average temperature and sea level as defined in scenario 2
- In addition: government measures to reduce damage from (sea level rise), e.g. dike construction. Damage functions are calibrated to match percentage of agricultural land at risk of inundation
- Facts and assumptions
  - Coast line of Mekong River Delta is 600 km
  - Dike needs to be of the same height along the entire coastline
  - Estimated costs: 40,000 EUR for one meter height and one meter length
  - Height increases with sea level each year
  - Damages associated with sea level rise in Mekong Delta River are zero, if and only if the height of the dike exceeds the sea level rise
- Benefit calculated as difference in GDP between a scenario with dike (Adaptation) and without dike (Sea Level) relative to the Baseline

5.3 Scenario 3 – „Adaptation“



Source: DGE–CRED model

giz Deutsche Gesellschaft für Internationale Zusammenarbeit (GIZ) GmbH

5.3 Scenario 4 – „Extremes“

- Increase in average temperature and sea level as defined in scenario 2
- In addition: occurrence of cyclones and droughts
- Occurrence of cyclones and droughts as well as their intensity is modeled as a random process

5.3 Simulation Results

- Stored in Excel file
- Include
  - Paths for all model variables, stored to individual sheets for every scenario
  - Figures comparing the outcomes of different simulated scenarios

References I

- Arndt, Channing, Finn Tarp, and James Thurlow (2015). "The economic costs of climate change: a multi-sector impact assessment for Vietnam". In: Sustainability 7.4, pp. 4131–4145.
- Hsiang, Solomon (2016). "Climate econometrics". In: Annual Review of Resource Economics 8, pp. 43–75.
- Miller, Ronald E and Peter D Blair (2009). Input-output analysis: foundations and extensions. Cambridge university press.
- Nordhaus, William D (1993). "Optimal greenhouse-gas reductions and tax policy in the "DICE" model". In: American Economic Review 83.2, pp. 313–317.

References II

Thuc, Tran, Nguyen Van Thang, Huynh Thi Lan Huong, Mai Van Khiem, Nguyen Xuan Hien, and Doan Ha Phong (2016). “Climate change and sea level rise scenarios for Vietnam”. In: Ministry of Natural resources and Environment. Hanoi, Vietnam.

Wing, Ian Sue (2004). “Computable general equilibrium models and their use in economy-wide policy analysis”. In: Technical Note, Joint Program on the Science and Policy of Global Change, MIT.