ECONSTOR
Make Your Publications Visible.

A Service of

ZBW Leibniz-Informationszentrum Wirtschaft
Leibniz Information Centre for Economics

Canova, Fabio; Paulsen, Kenneth Sæterhagen

Working Paper

Symbolic stationarization of dynamic equilibrium models

Working Paper, No. 18/2021

Provided in Cooperation with:
Norges Bank, Oslo

Suggested Citation: Canova, Fabio; Paulsen, Kenneth Sæterhagen (2021) : Symbolic stationarization of dynamic equilibrium models, Working Paper, No. 18/2021, ISBN 978-82-8379-217-1, Norges Bank, Oslo,
https://hdl.handle.net/11250/2835495

This Version is available at:
https://hdl.handle.net/10419/264937

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.

https://creativecommons.org/licenses/by-nc-nd/4.0/deed.no

WWW.ECONSTOR.EU

Mitglied der Leibniz-Gemeinschaft

WORKING PAPER

Symbolic Stationarization of Dynamic Equilibrium Models

NORGES BANK RESEARCH

18 | 2021

Fabio Canova
Kenneth Sæterhagen Paulsen

NORGES BANK

Working papers fra Norges Bank, fra 1992/1 til 2009/2 kan bestilles over e-post:
111facility@norges-bank.no

Fra 1999 og senere er publikasjonene tilgjengelige på www.norges-bank.no

Working papers inneholder forskningsarbeider og utredninger som vanligvis ikke har fått sin endelige form. Hensikten er blant annet at forfatteren kan motta kommentarer fra kolleger og andre interesserte. Synspunkter og konklusjoner i arbeidene står for forfatternes regning.

Working papers from Norges Bank, from 1992/1 to 2009/2 can be ordered by e-mail
FacilityServices@norges-bank.no

Working papers from 1999 onwards are available on www.norges-bank.no

Norges Bank's working papers present research projects and reports (not usually in their final form) and are intended inter alia to enable the author to benefit from the comments of colleagues and other interested parties. Views and conclusions expressed in working papers are the responsibility of the authors alone.

ISSN 1502-8190 (online)
978-82-8379-217-1 (online)

Symbolic Stationarization of Dynamic Equilibrium Models

Fabio Canova* Kenneth Sæterhagen Paulsen †

December 20, 2021

Abstract

Dynamic equilibrium models are specified to track time series with unit root-like behavior. Thus, unit roots are typically introduced and the optimality conditions adjusted. This step requires tedious algebra and often leads to algebraic mistakes, especially in models with several unit roots. We propose a symbolic algorithm that simplifies the step of rendering non-stationary models stationary. It is easy to implement and works when trends are stochastic or deterministic, exogenous or endogenous. Three examples illustrate the mechanics and the properties of the approach. A comparison with existing methods is provided.

Keywords: DSGE models, unit roots, endogenous growth, symbolic computation.

*BI Norwegian Business School and CEPR, fabio.canova@bi.no

†Norges Bank, corresponding author; kenneth.paulsen@norges-bank.no. This working paper should not be reported as representing the views of Norges Bank. The views expressed in this paper are those of the authors and do not necessarily reflect those of Norges Bank.

1 Introduction

Dynamic Stochastic General Equilibrium (DSGE) models are popular in academic and policy institutions and used for a variety of purposes: informally matching stylized facts present in the data, structural estimation, scenario analyses, and forecasting. Often these models just consider stationary disturbances and a standard production structure with constant returns to scale, making the output a vector of stationary time series. When this is the case, one has to choose how to compare the model to data displaying near non-stationary behavior and how to design estimation and inferential procedures which can cope with the mismatch (see e.g. Canova (2014)).

Alternatively, one could assume that some of the disturbances are non-stationary, for example, assume permanent shocks to total factor productivity. In this case, a balanced growth path must be found prior to the computation of the solution and the optimality conditions transformed so that a stationary equilibrium exits. Standard solution procedures such as Blanchard and Kahn (1980), Klein (2000) or Sims (2010), in fact, work under the assumption that the variables entering the optimality conditions are stationary. Manually performing the stationarization step is cumbersome, it involves tedious algebra and, more importantly, it is prone to mistakes, especially in medium or large scale models featuring a number of non-stationary disturbances.

This paper presents a fast and easy-to-implement algorithm that mechanically performs the stationarization step in models featuring unit roots or deterministic trends, when the sources of the trends are exogenous or endogenous, such as in Comin and Gertler (2006). Because the approach is symbolic, the cost of computations is trivial. Thus a researcher may explore the consequences of alternative assumptions regarding the location of the unit roots (preferences, technologies, policy shocks), examine the equations that are affected, and the implications they have for the observable variables. Such an exercise allows one to improve the specification and the empirical content by confronting the long run implications of the model and of the data systematically.

The output of the algorithm consists of a system of equations containing the stationary version of the original non-stationary model, and it can be written to a text file or exported to LaTeX. The user can thus check that the solution makes sense and then perform a standard dynamic analysis. The algorithm is contained in a package of routines, called NB toolbox (Paulsen (2021)), which also has a number of ready-built functions a researcher can employ for standard dynamic analysis. Given that the stationary model can be written to a text file, the output of the algorithm can also be used as an input in other toolboxes, such as DYNARE, Dynare (2021); IRIS, GPMN (2020); or RISE, Maih (2021), for filtering or estimation, if the user wishes to do so.

IRIS and DYNARE have the capability to deal with non-stationary disturbances in dynamic models. Our approach is more general than both. The first toolbox uses a numerical approach to perform the

stationarization step. Thus, the solution features approximation errors. In addition, the user needs to provide the balanced growth path for the approximation to be performed. The second toolbox needs the user to provide both the growth factors and the variables which are associated with the growth factors. In addition, at the moment, it can handle only exogenous unit root processes.

The rest of the paper is organized as follows. The next section describes the problem to be solved. The algorithm is in Section 3. Section 4 presents the syntax of the model file and the commands. Section 5 has examples. In Section 5.1 we consider a RBC model driven by non-stationary labor augmenting technology shocks. We stationarize the model manually (Section 5.1.1); with our algorithm (Section 5.1.2) and compare the two solutions. In Section 5.2 we repeat the exercise for a larger-scale model with unit root in the labor-augmenting and the investment-specific technologies, as in Justiniano *et al.* (2011)). We compare its long run properties to those of a model where unit roots appear in the production technology and in preferences. Section 5.3 considers a model with endogenous growth and compares manual and automatic solutions. Section 6 discusses the relationship between our approach and existing methods. Section 7 concludes. A number of on-line appendices contain the details and the files mentioned in the text.

2 The problem

The optimality conditions of a DSGE model can be written as:

$$E_t[F(X_{t+1}, X_t, X_{t-1}, A_t, A_{t-1}, U_t, \theta)] = 0, \tag{1}$$

where $F(.)$ is a $M \times 1$ vector of continuous non-linear functions, and X_t is a $M \times 1$ vector of endogenous variables, with the convention that the first R variables are non-stationary, and the remaining M-R are stationary, and U_t is a $N \times 1$ vector of stationary exogenous forcing variables. A_t is a $Q \times 1$ vector of non-stationary forcing variables which, for the sake of the presentation, are assumed to enter without leads, even though this is not a limitation of the approach, as leads can always be considered by extending the vector A_t. A_t may be exogenous or endogenous and, in this latter case, may depend on a subset of U_t, θ and X_t. Finally, θ is a $p \times 1$ vector of structural parameters.

The presence of the A_t in (1) rules out the possibility of using standard perturbation methods to solve the problem, since the Blanchard and Kahn condition fails. However, if the problem has a stationary representation, we can redefine the R endogenous non-stationary variables as $Z_t^i = \frac{X_t^i}{H^i(A_t)}$, $\forall i \in [1, R]$ and the M-R stationary variables as $Z_t^i = X_t^i$, $\forall i \in [R+1, M]$, and proceed in the standard way. Here H^i are continuous differentiable functions of A_t and represents the balanced growth path of the model. Let $\Delta Z_t^i \equiv \frac{H^i(A_t)}{H^i(A_{t-1})} \frac{X_{t-1}^i}{X_t^i}$ $\forall i \in [1, R]$ and $\Delta A_t = \frac{A_t}{A_{t-1}}$ and set $\frac{X_{t-1}}{X_t} = 1$.

Rather than finding the $H^i(A_t)$ functions, which is complicated when R is large, it is easier to compute the functions G^i satisfying $G^i(log(\Delta Z_t), log(\Delta A_t)) = 0, \forall i \in [1, R]$. These functions also characterize the

balanced growth path and are obtained with a set of restrictions, stated in Section 3.1. If we let

$$Y_t = \begin{bmatrix} Z_t \\ \Delta Z_t \\ \Delta A_t \end{bmatrix}, \tag{2}$$

a stationary representation of the optimality conditions of the model is

$$E_t[\hat{F}(Y_{t+1}, Y_t, Y_{t-1}, U_t, \theta)] = 0, \tag{3}$$

where $\hat{F}(.)$ is a function representing the non-linear equations, and Y_t has dimension $M + R + Q$. Note that ΔZ_t and ΔA_t belong to a block exogenous system when A_t is exogenous.

In words, our algorithm creates the Y_t from any set of (X_t, A_t, U_t), and transforms (1) into (3). To perform the transformation one needs four steps:

1. Find the restrictions holding in the balanced growth path.
2. Identify the R non-stationary variables.
3. Compute the functions $G^i, \quad \forall i \in [1, R]$.
4. Rewrite (1) into (3).

Clearly, trends must be properly placed for the procedure to succeed, as the restrictions obtained from the first step need not be consistent with balanced growth. For example, it is well know that, with a unit root in government expenditure, there are no restrictions on preferences and technologies producing balanced growth in a standard real business cycle (RBC) model.

3 The algorithm

This section describes the algorithm that automatizes the four steps outlined in Section 2.

3.1 Finding the restrictions holding in the balanced growth

One method to quickly identify which variables are non-stationary is to find the set of restrictions that must hold to rewrite the model in a stationary form. These restrictions are generically given by:

$$K(X_{ss}^{\Delta}, A_{ss}^{\Delta}, \theta) = 0, \tag{4}$$

where $X_t^{\Delta} = log(\frac{X_t}{X_{t-1}})$ and $A_t^{\Delta} = log(\frac{A_t}{A_{t-1}})$, and X_{ss}^{Δ} and A_{ss}^{Δ} are the values along the balanced growth path. Here the function K represents a vector of L linear restrictions. In general, $L \geq M$. Let θ_p and θ_d be two parameters, h and g two generic functions and $h^{\pm}$ a function formed using $h \pm g$. Let c and

v be functions obtained applying the rules described below to h and g, respectively. We can find the restrictions given by K, by applying the following rules to the equations of (1) by forward accumulation, and eliminating all duplicated restrictions [1].

Equal (=)

- If $X_{ss} = Y_{ss}$, then $X_{ss}^{\Delta} = Y_{ss}^{\Delta}$.
- If $X_{ss} = h(Y_{ss}, Z_{ss})$, then $X_{ss}^{\Delta} = c(Y_{ss}^{\Delta}, Z_{ss}^{\Delta})$.
- If $X_{ss} = h^{\pm}(Y_{ss}, Z_{ss})$, then $X_{ss}^{\Delta} = c(Y_{ss}^{\Delta}, Z_{ss}^{\Delta})$ and $X_{ss}^{\Delta} = v(Y_{ss}^{\Delta}, Z_{ss}^{\Delta})$.
- If $h(Y_{ss}, Z_{ss}) = \theta_p$ then $c(Y_{ss}^{\Delta}, Z_{ss}^{\Delta}) = 0$.

To understand the meaning of these restrictions, note that the first one states that if a model has an equation of the form $X_{ss} = Y_{ss}$, X_{ss} and Y_{ss} need to grow at the same rate along the balanced growth path. Similarly, the second restriction forces the two sides of the equation to have the same growth rate. The third expression consists of two terms; $X_{ss} = h(Y_{ss}, Z_{ss}) \pm g(Y_{ss}, Z_{ss})$ and forces them to grow at the same rate along the balanced growth path. The last condition requires $h(Y_{ss}, Z_{ss})$ to display no growth, as θ_p is a constant. The logic of the next expressions are similar:

Plus/Minus (±)

- If $X_{ss} \pm Y_{ss}$, then $X_{ss}^{\Delta} = Y_{ss}^{\Delta}$.
- If $X_{ss} \pm Y_{ss} \pm Z_{ss}$, then $X_{ss}^{\Delta} = Y_{ss}^{\Delta}$ and $X_{ss}^{\Delta} = Z_{ss}^{\Delta}$.
- If $X_{ss} \pm h(Y_{ss}, Z_{ss})$, then $X_{ss}^{\Delta} = c(Y_{ss}^{\Delta}, Z_{ss}^{\Delta})$.
- If $X_{ss} \pm h^{\pm}(Y_{ss}, Z_{ss})$, then $X_{ss}^{\Delta} = c(Y_{ss}^{\Delta}, Z_{ss}^{\Delta})$ and $X_{ss}^{\Delta} = v(Y_{ss}^{\Delta}, Z_{ss}^{\Delta})$.
- If $h(Y_{ss}, Z_{ss}) \pm \theta_p$, then $g(Y_{ss}^{\Delta}, Z_{ss}^{\Delta}) = 0$.
- If $\theta_p \pm \theta_d$, then $\theta_p \pm \theta_d$.

The last rule allows us to keep track of parameters. Since a parameter is an object of an equation, we need to ensure that, when applying the relevant operators, they are properly accounted for.

Times (*)

- If $X_{ss}Y_{ss}$, then $X_{ss}^{\Delta} + Y_{ss}^{\Delta}$.
- If $X_{ss}Y_{ss}Z_{ss}$, then $X_{ss}^{\Delta} + Y_{ss}^{\Delta} + Z_{ss}^{\Delta}$.
- If $X_{ss}h(Y_{ss}, Z_{ss})$, then $X_{ss} + c(Y_{ss}^{\Delta}, Z_{ss}^{\Delta})$.

[1] In a forward accumulation exercise, one first fixes the independent variable with respect to which the scaling is performed and then computes the trend of each sub-expression recursively.

- If $X_{ss}h^{\pm}(Y_{ss}, Z_{ss})$, then $c(Y_{ss}^{\Delta}, Z_{ss}^{\Delta}) = v(Y_{ss}^{\Delta}, Z_{ss}^{\Delta})$ and $X_{ss}^{\Delta} + c(Y_{ss}^{\Delta}, Z_{ss}^{\Delta})$.
- If $X_{ss}\theta_p$, then X_{ss}^{Δ}.
- If $\theta_p \cdot \theta_d$ then $\theta_p \cdot \theta_d$.

The last two rules imply that multiplying X_{ss} by a parameter leaves its growth rate unchanged, and that parameters do not grow.

Divide (/)

- If $\frac{X_{ss}}{Y_{ss}}$, then $X_{ss}^{\Delta} - Y_{ss}^{\Delta}$.
- If $\frac{(\frac{X_{ss}}{Y_{ss}})}{Zss}$, then $X_{ss}^{\Delta} - Y_{ss}^{\Delta} - Z_{ss}^{\Delta}$.
- If $\frac{X_{ss}}{(\frac{Y_{ss}}{Z_{ss}})}$, then $X_{ss}^{\Delta} - Y_{ss}^{\Delta} + Z_{ss}^{\Delta}$.
- If $\frac{X_{ss}}{h^{\pm}(Y_{ss}, Z_{ss})}$, then $c(Y_{ss}^{\Delta}, Z_{ss}^{\Delta}) = v(Y_{ss}^{\Delta}, Z_{ss}^{\Delta})$ and $X_{ss}^{\Delta} - c(Y_{ss}^{\Delta}, Z_{ss}^{\Delta})$.
- If $\frac{X_{ss}}{\theta_p}$, then X_{ss}^{Δ}.
- If $\frac{\theta_p}{\theta_d}$ then $\frac{\theta_p}{\theta_d}$.

Power (^)

- If $(X_{ss})^{\theta_p}$ then $\theta_p * X_{ss}^{\Delta}$.
- If $\theta_p^{Y_{ss}}$, then $Y_{ss}^{\Delta} = 0$, i.e. Y_t must be stationary.
- If $\theta_p^{h(Y_{ss}, Z_{ss})}$, then $c(Y_{ss}^{\Delta}, Z_{ss}^{\Delta}) = 0$, i.e. $h(Y_t, Z_t)$ must be a stationary.
- If $\theta_p^{\theta_d}$ then $\theta_p^{\theta_d}$.

Expression of the form $(X_{ss})^{Y_{ss}}$ or $(X_{ss})^{h(Y_{ss}, Z_{ss})}$ are not supported. This is because, for example, the first expression leads to a combination of two restrictions $Y_{ss}^{\Delta} = 0$ and $Y_{ss}X_{ss}^{\Delta}$ that prevent us from solving for the balanced growth path separately from the stationary steady state, as Y_{ss} appears in the system in (4). Situations of this type however are rare in the current wave of DSGEs.

Exponential (exp)

- If $e^{X_{ss}}$ then $X_{ss}^{\Delta} = 0$, i.e. X_t must be stationary.
- If $e^{h(Y_{ss}, Z_{ss})}$ then $c(Y_{ss}^{\Delta}, Z_{ss}^{\Delta}) = 0$, i.e. $h(Y_{ss}, Z_{ss})$ must be a stationary.
- If e^{θ_p} then e^{θ_p}.

Natural logarithm (log)

- If $log(X_{ss})$ then $X_{ss}^{\Delta} = 0$, i.e. X_t must be stationary.

- If $log\left(h(Y_{ss}, Z_{ss})\right)$ then $c(Y_{ss}^{\Delta}, Z_{ss}^{\Delta}) = 0$, i.e. $h(Y_{ss}, Z_{ss})$ must be stationary.
- If $log(\theta_p)$ then $log(\theta_p)$.

Example To illustrate how the rules work, suppose we want to find the restrictions provided by:

$$X_t \times Y_t + Z_t^{\theta_1} - Q_t. \tag{5}$$

Define steady state auxiliary variables as follows:

$$\begin{aligned} & X_{ss} \times Y_{ss} + Z_{ss}^{\theta_1} - Q_{ss} \\ & = (w_1 \times w_2) + w_3^{\theta_1} - w_4 \\ & \equiv w_5 + w_6 - w_4 = w_7. \end{aligned} \tag{6}$$

where $w_5 = w_1 \times w_2$ and $w_6 = w_3^{\theta_1}$. Let $\dot{w}$ be the operator obtained applying the above rules. The forward accumulation step consists of the followings:

1. $w_1 = X_{ss}$ leads to $\dot{w}_1 = X_{ss}^{\Delta}$
2. $w_2 = Y_{ss}$ leads to $\dot{w}_2 = Y_{ss}^{\Delta}$
3. $w_3 = Z_{ss}$ leads to $\dot{w}_3 = Z_{ss}^{\Delta}$
4. $w_4 = Q_{ss}$ leads to $\dot{w}_4 = Q_{ss}^{\Delta}$
5. $w_5 = w_1 \times w_2$ leads to $\dot{w}_5 = \dot{w}_1 + \dot{w}_2$
6. $w_6 = w_3^{\theta_1}$ leads to $\dot{w}_6 = \theta_1 \dot{w}_3$
7. $w_7 = w_5 + w_6 - w_4$ leads to $\dot{w}_5 = \dot{w}_6$ and $\dot{w}_5 = \dot{w}_4$

To apply the rules, we need an initial condition. Since $w_1 = X_{ss}$, no rules applies, and we must normalize by creating a seed. We set $\dot{w}_1 = X_{ss}^{\Delta}$, i.e. applying the dot operator to X_{ss}, gives its growth rate. Then:

1. $\dot{w}_5 = \dot{w}_6$ gives $\dot{w}_1 + \dot{w}_2 = \theta_p \dot{w}_3$ or, $X_{ss}^{\Delta} + Y_{ss}^{\Delta} = \theta_p Z_{ss}^{\Delta}$.
2. $\dot{w}_5 = \dot{w}_4$ gives $\dot{w}_1 + \dot{w}_2 = \dot{w}_4$ or $X_{ss}^{\Delta} + Y_{ss}^{\Delta} = Q_{ss}^{\Delta}$.

Applying the above rules to all the equations present in a model give us L_1 restrictions.

3.2 Identify the non-stationary variables

To the L_1 restrictions found in the last section, we add L_2 equations of the form $A^{\Delta}_{ss} = \gamma$; that is, A_t has a drift equal to γ. This allows the identification of the non-stationary variables without knowing the features of the forcing process A_t, which is necessary when the unit root is endogenous. In the end, we have $L = L_1 + L_2$ constraints. This produces a system of equations linear in X^{Δ}_{ss} and A^{Δ}_{ss}, that is:

$$C(\theta)\begin{bmatrix} X^{\Delta}_{ss} \\ A^{\Delta}_{ss} \end{bmatrix} = \begin{bmatrix} C^X(\theta) & C^A(\theta) \\ 0 & I \end{bmatrix}\begin{bmatrix} X^{\Delta}_{ss} \\ A^{\Delta}_{ss} \end{bmatrix} = \begin{bmatrix} 0 \\ \gamma \end{bmatrix}, \tag{7}$$

This system can be solved if $rank(C(\theta)) = M + Q$. In some cases a solution can be found even when $rank(C(\theta)) < M + Q$, in which case some of the endogenous variables are unrestricted. To allow for this possibility, we constrain all unrestricted endogenous variables to have no growth in steady state; that is for any X^1 in this group; we set $(X^1)^{\Delta}_{ss} = 0$. If a solution to (7) exists, it gives us the balanced growth path for each value of γ. We use this solution to identify the R non-stationary variables.

3.3 Obtaining the G^i functions

From (7) it is also possible to identify the variables dependent on G^i functions. Let $\hat{C}(\theta)$ be the matrix consisting of the columns corresponding to the R non-stationary variables in $\begin{bmatrix} C^X(\theta) & C^A(\theta) \end{bmatrix}$. Generally, the number of rows of $(\hat{C}(\theta)) > R$, and thus have more equations than non-stationary variables. Applying Gaussian elimination to the rows of $\hat{C}(\theta)'$, produces the row echelon form:

$$C(\theta)^{echelon} = \begin{bmatrix} c(\theta)^{echelon}_{1,1} & \cdots & c(\theta)^{echelon}_{1,R} \\ 0 & \ddots & \vdots \\ 0 & 0 & c(\theta)^{echelon}_{R,R} \\ 0 & 0 & 0 \end{bmatrix}. \tag{8}$$

During the process, we record the first R linearly independent columns of $\hat{C}(\theta)'$. Since the columns of $\hat{C}(\theta)'$ represent a function, we can use this index to select the R linearly independent equations in R non-stationary endogenous variables. This gives the required G^i functions.

3.4 Representing the stationary system

After recording the non-stationary variables, we "divide through" the optimality conditions. Noticing that $Z^i_{t+1} = \frac{X^i_{t+1}}{H^i(A_t)} = X^i_{t+1}\Delta Z^i_{t+1}$, $Z^i_{t-1} = \frac{X^i_{t-1}}{H^i(A_t)} = \frac{X^i_{t-1}}{\Delta Z^i_t}$, $\frac{A^i_{t+1}}{A^i_t} = \Delta A^i_{t+1}$, $\frac{A^i_t}{A^i_t} = 1$, $\frac{A^i_t}{A^i_{t-1}} = (\Delta A^i_t)^{-1}$, we can substitute X^i_{t+1} with $X^i_{t+1}\Delta Z^i_{t+1}$, X^i_{t-1} with $\frac{X^i_{t-1}}{\Delta Z^i_t}$, A^i_{t+1} with ΔA^i_{t+1}, A^i_t with 1 and A_{t-1} with $(\Delta A^i_t)^{-1}$, whenever they appear. These equations together with the G^i functions will give us (3).

4 The model file language

In order to run the algorithm with the NB toolbox, one needs to write the equations of the model to a file with .nb (or .mod) extension. The file consists of six blocks, describing the endogenous and exogenous variables, the parameters, the model, the unit root processes and what to report.

endogenous

A list of the endogenous variables. They may appear with lags and leads in the model, but a maximum of one lead is imposed. Variables are separated by a space or starting a new line.

unitrootvars

A list of the variables that generate non-stationary dynamics. They can have lags and leads, but a maximum of one lead is imposed. Variables are separated by a space or starting a new line.

exogenous

A list of the exogenous variables. They cannot have lags or leads and are assumed to be white noise with unit standard deviation. Variables are separated by a space or stating a new line.

parameters

A list of the parameters. This is block is optional, as one could write parameter values directly into the equations. Parameters are separated by a space or starting a new line.

model

A list of the equations. Each equation must end with a semicolon and may span more than one line. It must be typed-in using the endogenous and the exogenous variable names, and the parameter names or numbers (hard-coded parameters). For lead or lags use varName(+1) and varName(-1).

reporting

A list of the variables to be reported. For example, although the consumption share does not appear in a model, one may want to compute its response function to shocks or other summary statistics. The reporting command allows to set variables which do not appear directly in the model. The expression consists of the endogenous variables and/or the parameters and any MATLAB (or owned MATLAB) function that acts on a double vector, and returns an output of the same dimension as the input. Each equation must start with a variable name, there should be an equal sign followed by an expression, and it must end with a semicolon; for example, cy = c/y; or csx = cumsum(x).

Note that the specification of processes of A_t must be included in the model declaration block in its non-stationary form. If A_t does not depend on the exogenous forcing variables, there is a deterministic trend in the model; otherwise a stochastic trend is assumed. The processes for A_t may also depend on the

endogenous variables. This way, the model language accommodates many different assumption about the drivers of the trend. If no `unitrootvars` command is used in the file, the toolbox assumes that all the shocks are stationary and solves the model with standard tools.

A number handy functions may be used to simplify in the model file declaration. For example, steady_state(M) returns the steady state of the variable M and bgp(M) returns its growth rate along the balanced growth path. Both these functions are used in the example in Section 5.2. The algorithm is implemented as an object oriented MATLAB code. A description of how the algorithm works is in the Appendix A.

5 Examples

To show that the algorithm is easy to use and produces reliable results, we consider three models. The first is a real business cycle (RBC) model with a unit root in labor augmenting productivity; the second is a version of the larger-scale New Keynesian (NK) model of Justiniano *et al.* (2011) with multiple unit roots; the third is a closed economy model with R&D generated endogenous labor augmenting productivity.

5.1 A RBC model

There is exogenous labor supply, endogenous capital accumulation, and δ is the capital depreciation rate. Production is Cobb-Douglas in capital and labor, and the labor augmenting term A_t is a unit root:

$$log(\Delta a_t) = (1-\lambda)log(g) + \lambda log(\Delta a_{t-1}) + u_t. \tag{9}$$

where $\Delta a_t = A_t/A_{t-1}$, g is the long-run mean, λ is a persistence parameter, and $u \sim iid(0,\sigma^2)$. Preferences are logarithmic in consumption and agents discount the future at the rate β. The details of the model and the optimality conditions are in the Appendix B. We first show how to solve the model manually, and then implement the algorithm we propose.

5.1.1 Using a manual approach

Steps 1-3: Finding the balanced growth path; the non-stationary variables and the G^i functions.

With only one shock, output, investment, consumption, and capital all grow at the rate log(g). To verify the guess, we apply the rules of Section 3.1. The production function gives

$$Y_{ss}^{\Delta} = \gamma(A_{ss}^{\Delta} + L_{ss}^{\Delta}) + (1-\gamma)K_{ss}^{\Delta}. \tag{10}$$

where γ is a share parameter. The capital accumulation equation and the resource constraint give:

$$K_{ss}^{\Delta} = I_{ss}^{\Delta}. \tag{11}$$

$$Y_{ss}^{\Delta} = C_{ss}^{\Delta} \tag{12}$$

$$Y_{ss}^{\Delta} = I_{ss}^{\Delta} \tag{13}$$

The Euler equation and the definition of the real rate give:

$$r_{ss}^{\Delta} = 0, \tag{14}$$

$$r_{ss}^{\Delta} = Y_{ss}^{\Delta} - K_{ss}^{\Delta} \tag{15}$$

Finally, because labor supply is inelastic, it must be the case that

$$L_{ss}^{\Delta} = 0. \tag{16}$$

These equations imply:

$$C_{ss}^{\Delta} = I_{ss}^{\Delta} = K_{ss}^{\Delta} = Y_{ss}^{\Delta} = A_{ss}^{\Delta} \equiv log(g), \tag{17}$$

Hence, for these five variables $H^i(A_t) = A_t$, and equations (10)-(13), (15) (16) determine the G^i functions.

Step 4: The stationary system

For any M_t, let $m_t = \frac{M_t}{H^m(A_t)}$ be its stationary version, and $\Delta m_t = exp(M_t^{\Delta}) = \frac{H^m(A_t)}{H^m(A_{t-1})}$ its growth rate. The equations of the stationary system are:

Production function:

$$y_t = L_t^{\gamma} \left(\frac{k_{t-1}}{\Delta k_t} \right)^{1-\gamma}. \tag{18}$$

Capital accumulation equation:

$$k_t = (1-\delta)\frac{k_{t-1}}{\Delta k_t} + i_t, \tag{19}$$

Market clearing condition:

$$y_t = c_t + i_t, \tag{20}$$

Euler equation:

$$\frac{E[c_{t+1}\Delta c_{t+1}]}{c_t} = \beta(1+r), \tag{21}$$

Definition of the real rate:

$$(1-\gamma)\frac{E[y_{t+1}\Delta y_{t+1}]}{k_t} = r + \delta. \tag{22}$$

(21), (18), (19) (20) determine the steady state of the stationary system:

$$r_{ss} = \frac{g}{\beta} - 1. \tag{23}$$

$$y_{ss} = \left(\frac{k_{ss}}{g} \right)^{(1-\gamma)}, \tag{24}$$

$$i_{ss} = \left(1 - \frac{1-\delta}{g} \right) k_{ss}, \tag{25}$$

$$c_{ss} = y_{ss} - i_{ss}, \text{ and} \tag{26}$$

Moreover, using (24) into (22) we get

$$k_{ss} = \left(\frac{r+\delta}{1-\gamma}\right)^{(-\frac{1}{\gamma})} g. \tag{27}$$

The steady state for the five unknowns is the solution to these five equations.

To compute the linear approximation to the solution of the stationary system around the steady state, one can employ standard package such as DYNARE (Dynare (2021)), GENSYS (Sims (2010)) or the NB toolbox (Paulsen (2021)). With the linear approximate solution, interesting statistics such as impulse responses, variance or historical decompositions can be easily computed.

5.1.2 Using the automatic approach

Rather than proceeding manually, we write the equations of the non-stationary model to a file and let a software take care of the rest. The file is as follows:

```
endogenous
c dA i k l r y

exogenous
u

unitrootvars
A

parameters
beta  delta g gamma lambda std_u

model
    c(+1)/c           = beta*(1 + r);
    l                 = 1;
    y                 = (A*l)^gamma * k(-1)^(1-gamma);
    (1-gamma)*y(+1)/k = r + delta;
    k                 = (1-delta)*k(-1) + i;
    y                 = c + i;
    dA                = A/A(-1);
    A/A(-1)           = g^(1-lambda)*(A(-1)/A(-2))^lambda*exp(std_u*u);
```

```
reporting
    A         = exp(cumsum(log(dA) - log(steady_state(dA)))); % level variables
    C         = c*A;     I  = i*A;      K  = k*A;     Y = y*A;
    c_log_dev = exp(log(c) - log(steady_state(c)));  % deviations  from steady state
    i_log_dev = exp(log(i) - log(steady_state(i)));
    k_log_dev = exp(log(k) - log(steady_state(k)));
    r_dev     = r - steady_state(r);
    y_log_dev = exp(log(y) - log(steady_state(y)));
```

With this file (saved as non_stationary.nb), the NB commands needed to stationarize the system and to compute impulse responses:

```
modelNS = nb_dsge('nb_file','non_stationary.nb'); %Read the model file

modelNS = set(modelNS,'name','Stationarize automatically'); % Name it

param        = struct( );    % Assign parameter values
param.g      = 1.03;   param.gamma  = 0.60;  param.delta  = 0.10;
param.beta   = 0.97;  param.lambda = 0;   param.std_u  = 0.01;
modelNS      = assignParameters(modelNS,param);

% Find the balanced growth path, (steps 1-3 of the algorithm):
modelNS  = solveBalancedGrowthPath(modelNS);

% Make  the model  stationary (step 4 of the algorithm):
modelNS = stationarize(modelNS)

% Find the steady state and return it as  a cell matrix:
modelNS = checkSteadyState(modelNS,...
    'solver',               'fsolve',...
    'steady_state_solve',   true,...
    'steady_state_default', @ones);
ss = getSteadyState(modelNS)

% Return the balanced growth path solution as a cell matrix:
bgp = getBalancedGrowthPath(modelNS)

% Obtain the linear solution: y(t) = A y(t-1) + B eps(t)
% and  compute  impulse responses
modelNS = solve(modelNS);

% Compute and plot  impulse responses
% Note modelS object represents the model stationarized manually.
[irfs,~,plotter] = irf([modelNS,modelS],...
   'variables',[modelNS.dependent.name(1:7),modelNS.reporting(:,1)'],...
   'settings',{'legBox','off','legFontSize',18,'subPlotSize',[4,3],...
               'figureTitle',false});
nb_graphInfoStructGUI(plotter)
```

The stationary model file produced by these commands is in Appendix B. If a user wants to save the stationary model for later purposes or to use in other packages, it is possible to write it to a file using:

```
writeModel2File(modelNS,'stationarized.nb')
```

We plot the dynamics induced by a one standard deviation increase in the labor-augmenting shock in the manual and automated solutions in Figure 1; the top part has the stationary variables, the bottom part the level variables. If the automated solution is accurate, the responses should be identical.

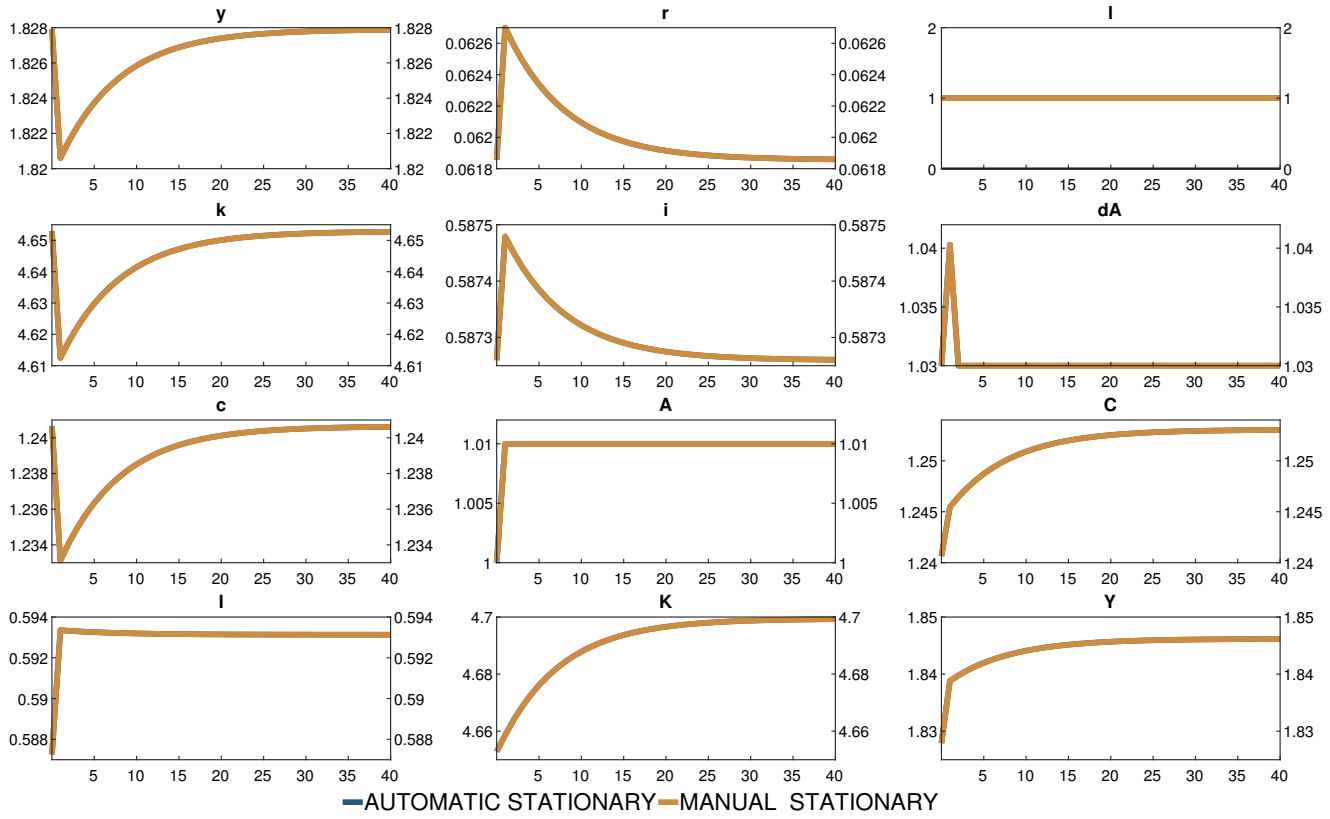

Figure 1: Impulse responses. A is the labour augmenting process. y, r, l, k, i, dA and c are the stationary variables in log deviation from the steady state; C, I, K and Y are the non-stationary levels of the variables.

The two impulse responses are clearly indistinguishable. As the economy becomes permanently more productive, output (Y), consumption (C), investment (I) and capital (K) jump on impact and slowly reach the new long run level. Because the expected output to capital ratio increases on impact, the real interest rate (r) rises and this induces a fall in transitory consumption (c). As the economy moves to the

new steady state the real interest rate falls and transitory variables return to their original levels.

5.2 A New Keynesian model

This section compares the manual and the automated decision rules of a slightly modified version of the Justiniano *et al.* (2011) model and studies the consequences of assuming unit roots in different shocks. The model allows us to illustrate how the algorithm works when the trend path is driven by multiple disturbances. The model deviates from Justiniano *et al.* (2011) in three ways: we assume Rotemberg rather than Calvo prices and wages adjustments; we use different functional forms for capital utilization and for investment-adjustment costs; and we set government purchases of goods and services to zero.

There are six types of agents: final and intermediate good producing firms, consumers, investment and capital producers, and entrepreneurs. Their problems and the optimality conditions are in the Appendix C. The model features nine disturbances, two of which are assumed to be non-stationary: Z_t, the labor-augmenting technology process, and Υ_t, the investment-specific technology process. Z_t grows at the gross rate $\pi_t^z = \frac{Z_t}{Z_{t-1}}$, while Υ_t grows at the gross rate $\pi_t^{\Upsilon} = \frac{\Upsilon_t}{\Upsilon_{t-1}}$. The law of motion of all the disturbances is also described in the Appendix C.

Let P_t^X be the nominal price of the variable X in period t. The consumption good is the numeraire and its price is P_t, which will be non-stationary. Let $\pi_t = \frac{P_t}{P_{t-1}}$ be the inflation rate, W_t the nominal wage, $R_t^X \equiv 1 + r_t^X$ the gross interest rate in sector or variable X and r_t^X the net rate of interest. All other variables are expressed in real terms, unless otherwise stated. Finally, let $\widetilde{X}_t$ denote the stationary version of X and $\widetilde{X}_{ss}$ its steady state.

5.2.1 Computing the stationary system manually

Justiniano *et al.* (2011) state that $Z_t \Upsilon_t^{\frac{\alpha}{1-\alpha}}$ is the balanced growth path for this economy. To show that this is indeed the case, we scale the optimality conditions of by this factor and show that they still hold after the transformation. From the final and intermediate goods sectors' conditions we get:

i) Final good production function:

$$\widetilde{A}_t = \widetilde{Q}_t. \tag{28}$$

ii) Pricing of final good:

$$\widetilde{P}_t = \widetilde{P}_t^Q. \tag{29}$$

iii) Intermediate goods production function:

$$\widetilde{Q}_t \quad = \quad (z_t^L L_t)^{1-\alpha} \left(\widetilde{\overline{K}}_t\right)^{\alpha} \tag{30}$$

iv) Wage and nominal rate definitions:

$$\widetilde{W}_t \equiv \frac{W_t}{P_t \Upsilon_t^{\frac{\alpha}{1-\alpha}} Z_t}; \; \widetilde{R}_{K,t} \equiv \frac{\Upsilon_t R_{K,t}}{P_t} \tag{31}$$

v) Labor demand function:

$$L_t = (1-\alpha)\frac{\widetilde{MC}_t}{\widetilde{W}_t}\widetilde{Q}_t. \tag{32}$$

vi) Capital demand function:

$$\widetilde{K}_t = \alpha\frac{\widetilde{MC}_t}{\widetilde{R}_t^K}\widetilde{Q}_t. \tag{33}$$

vii) Using $P_t^Q = P_t^Q(n)$ and $\pi_t^Q \equiv \frac{P_t^Q}{P_{t-1}^Q}$, the cost of adjusting prices is

$$\gamma_{PQ,t} = \frac{\phi^{PQ}}{2}\left[\frac{\pi_t^Q}{\pi_{t-1}^Q} - 1\right]^2. \tag{34}$$

viii) The Euler equation for intermediate firms is:

$$\begin{aligned}\widetilde{Q}_t - \theta_t^H\widetilde{Q}_t + \widetilde{MC}_t\theta_t^H\frac{\widetilde{Q}_t}{\widetilde{P}_t^Q} - \phi^{PQ}\left[\frac{\pi_t^Q}{\pi_{t-1}^Q} - 1\right]\frac{\pi_t^Q}{\pi_{t-1}^Q}\widetilde{Q}_t \\ +E_t\left\{\Lambda\phi^{PQ}\left[\frac{\pi_{t+1}^Q}{\pi_t^Q} - 1\right]\frac{(\pi_{t+1}^Q)^2}{\pi_t^Q}\widetilde{Q}_{t+1}\left(\pi_{t+1}^{\Upsilon}\right)^{\frac{\alpha}{1-\alpha}}\pi_{t+1}^z\right\} = 0.\end{aligned} \tag{35}$$

The first order conditions for the investment and capital producer give:

ix) Production function for investment goods:

$$\widetilde{I}_t = \widetilde{Y}_t^I. \tag{36}$$

x) Investment price:

$$\widetilde{P}_t^I = 1. \tag{37}$$

xi) Production function for capital goods:

$$\widetilde{K}_t^{new} = z_{I,t}\left(1 - S\left(\frac{\widetilde{I}_t}{\widetilde{I}_{t-1}}\right)\right)\widetilde{I}_t. \tag{38}$$

xii) Optimality condition for investments:

$$\begin{aligned}\left[\Lambda\frac{1}{\pi_{t+1}^{\Upsilon}}\widetilde{P}_{t+1}^K z_{I,t+1}S_2'\left(\frac{\widetilde{I}_{t+1}}{\widetilde{I}_t}\right)\widetilde{I}_{t+1}\right] - \\ \widetilde{P}_t^K z_{I,t}\left[1 - S\left(\frac{\widetilde{I}_t}{\widetilde{I}_{t-1}}\right) - S_1'\left(\frac{\widetilde{I}_t}{\widetilde{I}_{t-1}}\right)\widetilde{I}_t\right] - \widetilde{P}_t^I = 0\end{aligned} \tag{39}$$

xiii) Investment adjustment costs:

$$S\left(\frac{\widetilde{I}_t}{\widetilde{I}_{t-1}}\right) = \frac{\phi_{I1}}{2}\left[\frac{\widetilde{I}_t}{\widetilde{I}_{t-1}}\left(\pi_t^{\Upsilon}\right)^{\frac{1}{1-\alpha}}\pi_t^z - \left(\pi_{ss}^{\Upsilon}\right)^{\frac{1}{1-\alpha}}\pi_{ss}^z\right]^2. \tag{40}$$

and thus

$$S_1'\left(\frac{\widetilde{I}_t}{\widetilde{I}_{t-1}}\right)\widetilde{I}_t = \begin{array}{c}\phi_{I1}\left[\frac{\widetilde{I}_t}{\widetilde{I}_{t-1}}\left(\pi_t^{\Upsilon}\right)^{\frac{1}{1-\alpha}}\pi_t^z - \left(\pi_{ss}^{\Upsilon}\right)^{\frac{1}{1-\alpha}}\pi_{ss}^z\right] \\ \frac{\widetilde{I}_t}{\widetilde{I}_{t-1}}\left(\pi_t^{\Upsilon}\right)^{\frac{1}{1-\alpha}}\pi_t^z\end{array}. \tag{41}$$

$$S_2'\left(\frac{\widetilde{I}_t}{\widetilde{I}_{t-1}}\right)\widetilde{I}_t = -\begin{array}{c}\phi_{I1}\left[\frac{\widetilde{I}_t}{\widetilde{I}_{t-1}}\left(\pi_t^{\Upsilon}\right)^{\frac{1}{1-\alpha}}\pi_t^z - \left(\pi_{ss}^{\Upsilon}\right)^{\frac{1}{1-\alpha}}\pi_{ss}^z\right] \\ \left(\frac{\widetilde{I}_t}{\widetilde{I}_{t-1}}\left(\pi_t^{\Upsilon}\right)^{\frac{1}{1-\alpha}}\pi_t^z\right)^2\end{array}. \tag{42}$$

The first order conditions for the households and the entrepreneurs give:

Xiv) Marginal utility of consumption:

$$\left(\Upsilon_t^{\frac{\alpha}{1-\alpha}}Z_t\right)u'\left(C_t\right) \equiv \widetilde{u}'\left(\widetilde{C}_t\right) = z_t^u\left[\frac{\widetilde{C}_t - \widetilde{C}_{t-1}\left(\frac{b^c}{\left(\pi_t^{\Upsilon}\right)^{\frac{\alpha}{1-\alpha}}\pi_t^z}\right)}{\left(1-\frac{b^c}{\left(\pi_{ss}^{\Upsilon}\right)^{\frac{\alpha}{1-\alpha}}\pi_{ss}^z}\right)}\right]^{-1}. \tag{43}$$

xv) Marginal disutility of labor:

$$v'\left(L_t\right) = \left[\frac{L_t\left(j\right) - b^l L_{t-1}}{1-b^l}\right]^{\zeta}. \tag{44}$$

xvi) The stochastic discount factor:

$$\Lambda_{t,t+i} \equiv \beta^i \frac{\rho_{t+1}\widetilde{u}'\left(\widetilde{C}_{t+i}\right)}{\rho_t\widetilde{u}'\left(\widetilde{C}_t\right)}\frac{1}{\pi_{t+i}}\frac{1}{\left(\pi_{t+i}^{\Upsilon}\right)^{\frac{\alpha}{1-\alpha}}\pi_{t+i}^z}, \tag{45}$$

Then the one-period ahead stochastic discount factor is then

$$\Lambda_{t,t+1} \equiv \beta \frac{\rho_{t+1}\widetilde{u}'\left(\widetilde{C}_{t+1}\right)}{\rho_t\widetilde{u}'\left(\widetilde{C}_t\right)}\frac{1}{\pi_{t+1}}\frac{1}{\left(\pi_{t+1}^{\Upsilon}\right)^{\frac{\alpha}{1-\alpha}}\pi_{t+1}^z}. \tag{46}$$

xvii) Optimality condition for bond holdings:

$$E_t\left[\Lambda_{t,t+1}\right]R_t = 1, \tag{47}$$

xviii) Labor supply equation:

$$\widetilde{W}_t = \psi_t MRS(L_t, \widetilde{C}_t)\begin{bmatrix}\left[\psi_t - 1\right]\left[1-\gamma_t\right] \\ +\phi^W\frac{\pi_t^W}{\pi_{t-1}^W}\left(\frac{\pi_t^W}{\pi_{t-1}^W} - 1\right) \\ -E\left[\Lambda\phi^W\frac{L_{t+1}}{L_t}\frac{\pi_{t+1}^W}{\pi_t^W}\left(\frac{\pi_{t+1}^W}{\pi_t^W} - 1\right)\right]\end{bmatrix}^{-1}. \tag{48}$$

where

$$MRS(L_t, C_t) = \frac{\Upsilon_t^{\frac{\alpha}{1-\alpha}}Z_t v'\left(L_t\right)}{\widetilde{u}'\left(\widetilde{C}_t\right)} = \Upsilon_t^{\frac{\alpha}{1-\alpha}}Z_t MRS(L_t, \widetilde{C}_t). \tag{49}$$

xix) Wage adjustment equation:

$$\gamma_t \equiv \frac{\phi^W}{2}\left[\frac{\pi_t^W}{\pi_{t-1}^W}-1\right]^2. \tag{50}$$

xx) Optimality condition for capital:

$$\widetilde{P}_t^K = E_t\left[\Lambda\pi_{t+1}\left(\begin{array}{c}\frac{1}{\pi_{t+1}^{\Upsilon}}\widetilde{P}_{t+1}^K(1-\delta)\\ +\frac{1}{\pi_{t+1}^{\Upsilon}}\widetilde{R}_{K,t+1}U_{t+1}-\frac{1}{\pi_{t+1}^{\Upsilon}}\widetilde{\gamma}(U_{t+1})\end{array}\right)\right]. \tag{51}$$

xxi) Optimality condition for capital utilization:

$$\widetilde{R}_{K,t} = \widetilde{\gamma}'(U_t), \tag{52}$$

where we have used that

$$\widetilde{\gamma}(U_t) \equiv \Upsilon_t\gamma(U_t) = \frac{\widetilde{R}_{ss}^K}{\phi_u}\left[e^{\phi_u(U_t-1)}-1\right], \tag{53}$$

$$\widetilde{\gamma}'(U_t) \equiv \Upsilon_t\gamma'(U_t) = \widetilde{R}_{K,ss}e^{\phi_u(U_t-1)}. \tag{54}$$

xxii) Effective capital definition:

$$\widetilde{\overline{K}}_t = U_t\frac{\widetilde{K}_{t-1}}{\left(\pi_t^{\Upsilon}\right)^{\frac{1}{1-\alpha}}\pi_t^z}. \tag{55}$$

xxiii) Physical capital accumulation equation:

$$\widetilde{K}_t = \frac{(1-\delta)\widetilde{K}_{t-1}}{\left(\pi_t^{\Upsilon}\right)^{\frac{1}{1-\alpha}}\pi_t^z}+\widetilde{K}_t^{new}. \tag{56}$$

xxiv) Final goods market clearing:

$$\widetilde{A}_t = \widetilde{C}_t + \widetilde{Y}_t^I. \tag{57}$$

xxv) Final goods inflation:

$$\pi_t \equiv \frac{P_t}{P_{t-1}}. \tag{58}$$

xxvi) Intermediate goods inflation:

$$\pi_t^Q = \pi_t\frac{\widetilde{P}_t^Q}{\widetilde{P}_{t-1}^Q}. \tag{59}$$

xxvii) Wage inflation:

$$\pi_t^W = \frac{\widetilde{W}_t}{\widetilde{W}_{t-1}}\pi_t\pi_t^z\left(\pi_t^{\Upsilon}\right)^{\frac{\alpha}{1-\alpha}}. \tag{60}$$

xxviii) Output definition:

$$\widetilde{Y}_{NAT,t} = \widetilde{A}_t. \tag{61}$$

xxix) Output growth definition:

$$\triangle \widetilde{Y}_{NAT,t} = \frac{\widetilde{Y}_{NAT,t}}{\widetilde{Y}_{NAT,t-1}} \pi_t^z \left(\pi_t^{\Upsilon}\right)^{\frac{\alpha}{1-\alpha}} . \tag{62}$$

A file with these equations and a set of instructions used to solve the system with the NB toolbox are in the Appendix D.

5.2.2 Computing the stationary system automatically

The steps are the same as in the RBC example. First, we write the equations of the non-stationary model to a file and save it with name jpt_non_stationary.nb. This file is reproduced in the Appendix E together with the commands needed to make the model stationary and compute impulse responses. The resulting file with the stationary model is also reproduced in the Appendix F. To check for the accuracy of the automated solution, we compare impulse responses to an investment-specific technology shock in the stationary version of the model computed manually and automatically and plot them in figure 2. The two responses are identical also in this case.

An investment-specific shock permanently raises consumption, investment, and output although the level of consumption and level of output temporarily move in different directions. As an investment unit is permanently transformed into more capital, investment, consumption and output as a deviation from the steady state temporarily fall and converge slowly. The responses of stationary variables are persistent because price adjustment costs are important. Hours also fall but temporarily while price inflation and wage inflation initially fall and then rebound as demand for transitory investment builds up. Finally, given the Taylor rule, the dynamics of the interest rate track the dynamics of inflation along the adjustment path.

It is instructive to analyze the G^i functions produced. They are characterized by 18 equations:

$$A_t^{\triangle} = Q_t^{\triangle}, \tag{63}$$

$$T_t^{\triangle} - \alpha \bar{K}_t^{\triangle} = (1-\alpha) Z_t^{\triangle}, \tag{64}$$

$$\bar{K}_t^{\triangle} + {R^K}_t^{\triangle} = T_t^{\triangle}, \tag{65}$$

$$\left(\frac{W}{P}\right)_t^{\triangle} = T_t^{\triangle}, \tag{66}$$

$$I_t^{\triangle} - \Upsilon_t^{\triangle} = {Y^I}_t^{\triangle}, \tag{67}$$

$$\left(\frac{P^I}{P}\right)_t^{\triangle} = -\Upsilon_t^{\triangle}, \tag{68}$$

$${K^{new}}_t^{\triangle} = I_t^{\triangle}, \tag{69}$$

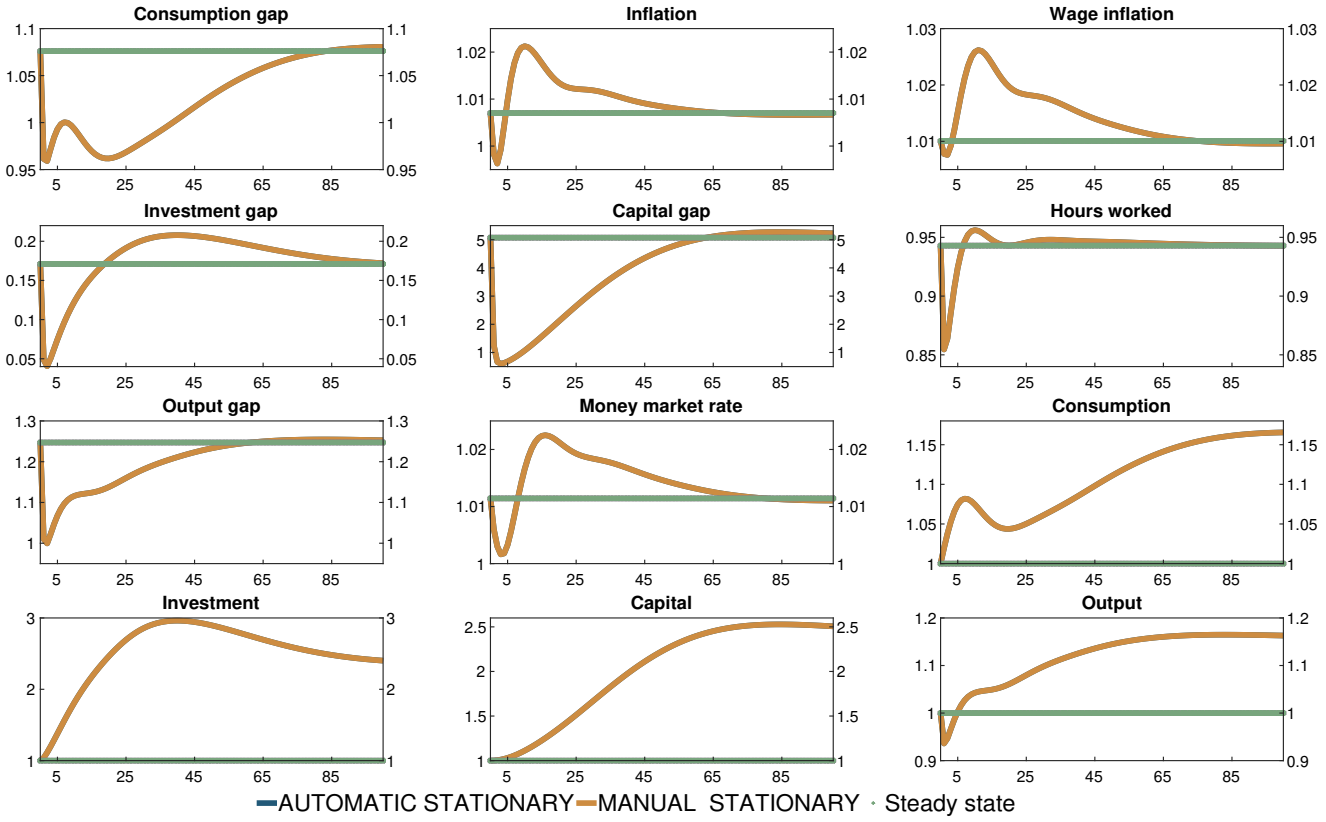

Figure 2: Responses of variables to an temporary shock to investment-specific technology. The green line gives the steady state/inital value.

$$\left(\frac{P^K}{P}\right)_t^{\triangle} = \left(\frac{P^I}{P}\right)_t^{\triangle}, \tag{70}$$

$$C_t^{\triangle} - u'(C)_t^{\triangle} = 0, \tag{71}$$

$$\left(\frac{W}{P}\right)_t^{\triangle} = \left(\frac{v'(L)}{u'(C)}\right)_t^{\triangle}, \tag{72}$$

$$\left(\frac{v'(L)}{u'(C)}\right)_t^{\triangle} = -u'(C)_t^{\triangle}, \tag{73}$$

$$\left(\frac{P^K}{P}\right)_t^{\triangle} = R^K{}_t^{\triangle}, \tag{74}$$

$$\left(\frac{P^K}{P}\right)_t^{\triangle} = \gamma(U)_t^{\triangle}, \tag{75}$$

$$R^{K}{}_{t}^{\triangle} = \gamma'(U)_{t}^{\triangle}, \tag{76}$$

$$K_{t}^{\triangle} = K^{new}{}_{t}^{\triangle}, \tag{77}$$

$$\bar{K}_{t}^{\triangle} = K_{t}^{\triangle}, \tag{78}$$

$$A_{t}^{\triangle} = C_{t}^{\triangle}, \tag{79}$$

$$Y_{t}^{\triangle} = A_{t}^{\triangle}. \tag{80}$$

(63)-(64) come from the final good and the intermediate production functions, (65)-(66) from the intermediate producer conditions, (67)-(68) from the investment good production function and optimization, (69)-(70) from capital production and optimization, (71)-(78) from the consumer optimality conditions, while (79)-(80) are obtained from market clearing and the resource constraint.

Recall that these equations characterize the trending variables. Note that, while capital, its real rate and the real wage are on the list, hours are not. Also, since the relative price of capital is trending, the marginal rate of substitution between leisure and consumption is also trending. We contrast these facts next with those implied by a model with unit root in preferences.

5.2.3 Changing the location of the unit root

To investigate the consequences of assuming unit roots in different disturbances, we let the preference shock, rather than the investment-specific technology shock, be a unit root process. In practice, this requires removing Υ_t from the list of processes generating non-stationary dynamics, and adding X_t to it, where X_t is the preference shock, entering the utility as follows

$$u\left(C_t\left(j\right)\right) = X_t z_t^u \left(1 - \frac{b^c}{\pi_{ss}^z}\right) \ln\left[\frac{C_t\left(j\right) - b^c C_{t-1}}{\left(1 - \frac{b^c}{\pi_{ss}^z}\right)}\right], \tag{81}$$

The model file is in the Appendix G. The automatically stationarized file in the Appendix I.

Clearly, when different disturbances feature unit root, the G^i function will generally change as different variables display trending behavior. Thus, a comparison of the G^i functions allows a researcher to understand the specification that is more likely to be compatible with the data. When the preference shock features a unit root equations (63), (69), (72), (77), (78), (79) and (80) are unchanged. Equations (67),(68), (70), (74)-(76) are no longer relevant. From the intermediate production function, we now have:

$$(\alpha - 1)L_{t}^{\triangle} + T_{t}^{\triangle} - \alpha\bar{K}_{t}^{\triangle} = (1 - \alpha)\, Z_{t}^{\triangle}, \tag{82}$$

while equations (65), (66) and (71) become

$$\bar{K}_{t}^{\triangle} + R^{K}{}_{t}^{\triangle} = T_{t}^{\triangle}, \tag{83}$$

Table 1 Balanced growth, selected variables.

	Model with a unit root in investment technology	**Model with a unit root in preferences**
C	1.0030	1.0029
I	1.0090	1.0029
K	1.0090	1.0029
L	1.0000	1.0011
Y	1.0030	1.0029
$\frac{P^I}{P}$	0.9941	1.0000
W	1.0030	1.0018

$$L_t^{\triangle} + \left(\frac{W}{P}\right)_t^{\triangle} = T_t^{\triangle}, \tag{84}$$

$$C_t^{\triangle} - u'(C)_t^{\triangle} = X_t^{\triangle}. \tag{85}$$

Compared with the version with investment specific unit root, the return on capital is now stationary while hours worked are trending to allow for the permanent increase in preferences for consumption. As a consequence, the marginal utility of labor is also trending and the marginal rate of substitution is also affected because with trending hours, real wages do not grow at same rate as output.

$$\zeta L_t^{\triangle} = v'(L)_t^{\triangle}, \tag{86}$$

$$\left(\frac{v'(L)}{u'(C)}\right)_t^{\triangle} = v'(L)_t^{\triangle} - u'(C)_t^{\triangle}. \tag{87}$$

Finally, the market clearing condition imply an additional restriction:

$$A_t^{\triangle} = Y^I{}_t^{\triangle}. \tag{88}$$

We report the growth rates of few variables of the model in table 1. In the economy with investment-specific technology shocks, investment and capital increase faster than output over time, as the real price of investment decreases over time, while this is not the case in the economy with unit roots in preferences. Thus, this set of facts can be used to select which specification is more in line with the data. Alternatively, suppose that the labor share is of interest. If, in the data, it is trending and hours are stationary, which happens in US, France, Germany and Canada (see e.g. Canova and Matthes (2021)) then a model with unit root in investment-specific shocks should be preferred, as it implies trending output and stationary hours. If, on the other hand, the labor share and hours are both trending, as it happens, for example, in the UK, then a model with unit root in preferences is more appealing, as both output and hours display a

Table 2 Steady states and unconditional variances, selected variables.

	A unit root in investment technology	**A unit root in preferences**
	Steady states	
$\tilde{C}$	1.0760	1.1324
π	1.0070	1.0070
π^W	1.0101	1.0089
$\tilde{I}$	0.1711	0.2333
$\tilde{K}$	5.0737	8.3790
$\tilde{L}$	0.9430	0.9499
$\tilde{Y}$	1.2471	1.3657
$1 + r_t$	1.0114	1.0053
	Variances	
$\tilde{C}$	4.4702	5.7502
π	0.4738	0.4815
π^W	0.1692	0.1699
$\tilde{I}$	3.5098	6.6931
$\tilde{K}$	2740.7912	6541.5216
$\tilde{L}$	1.6128	1.8329
$\tilde{Y}$	6.6636	7.8057
$1 + r_t$	0.2721	0.2817

unit root without being cointegrated. In general, with stylized growth facts in hand, one can choose which disturbances should be made non-stationary, and this helps to improve the specification of the model.

Table 2 reports the steady-state values and the unconditional variances of two stationary versions of the model for selected variables. Clearly, the steady states generally differ. For a given set of parameters, average consumption, investment, hours and output are typically higher in the economy with a unit root in preference, while wages and the nominal rate are smaller. This is because preferences and technologies are simultaneously trending. As a consequence, the model with a unit root in preference also displays higher variability. For example, capital is almost two and half times and consumption is one fourth more volatile in this case.

Impulse responses to a labor augmenting shock in the two systems are in figure 3. While the growth rates and the steady states of the two models differ, the dynamics in response to other shocks present in the model should be qualitatively independent to the location of the other unit roots. Indeed, the responses in the two systems are similar. The main quantitative difference comes from the fact already mentioned that capital grows faster in the economy with investment specific unit roots and this produces a different steady state for consumption, investment, and output.

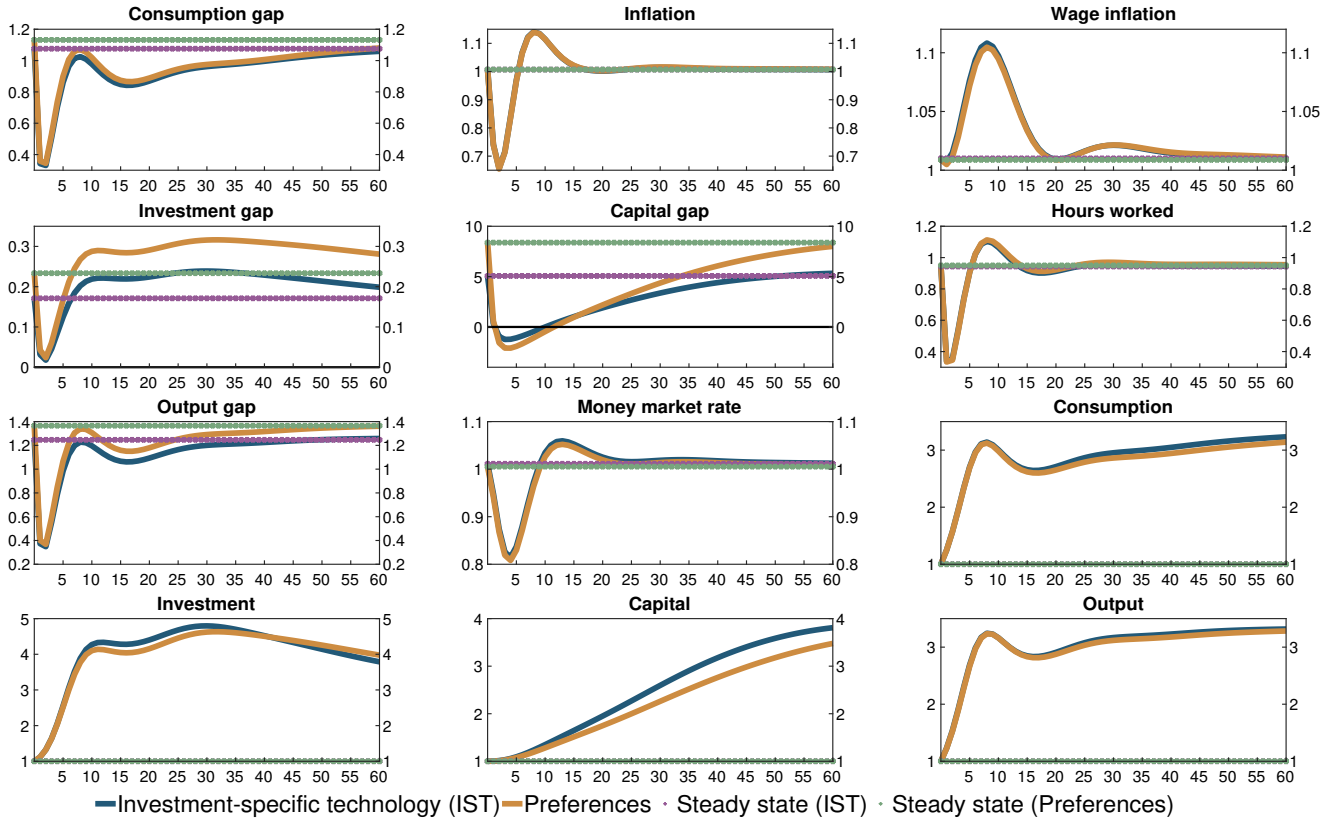

Figure 3: Responses to a labor augmenting technology shock. The purple line gives the steady state/inital value in the economy with investment-specific technology, while the green gives the steady state/inital value in the economy with unit roots in preferences.

5.3 Models with endogenous unit roots

As mentioned one of the advantages of our approach relative to existing ones, is that it can be employed also in models where the process responsible for the growth dynamics is endogenous rather than exogenous.

To illustrate that indeed the mechanical approach works in this case, we consider a closed economy model with exogenous labor supply, endogenous physical and R&D capital accumulation; preferences are logarithmic in consumption. The production function is

$$Y_t = (A_{t-1}L_t)^{\gamma}K_{t-1}^{1-\gamma}, \tag{89}$$

where Y_t is output, $L_t = 1$ is labor, K_t is physical capital, A_t is R&D capital and γ is the labor share.

Physical capital accumulates according to

$$K_t = (1-\delta)K_{t-1} + I_t, \tag{90}$$

where I_t is physical investment and δ the depreciation rate. The representative household maximizes

$$E\left[\sum_{s=0}^{\infty}\beta^s U(C_{t+s})\right], \tag{91}$$

where β is the discount factor, subject to (89), (90) and the constraint

$$Y_t + B_t + P_t^A(1-\delta^{RD})A_{t-1} = C_t + I_t + (1+r_{t-1})B_{t-1} + P_t^A A_t + T_t, \tag{92}$$

where B_t are one period bond, T_t are the lump-sum taxes and P_t^A the relative price of R&D capital and δ^{RD} is the depreciation rate of R&D capital. Optimization implies:

$$E\left[\beta\frac{U'(C_{t+1})}{U(C_t)}\right](1+r_t) = 1, \tag{93}$$

$$r_t = (1-\gamma)\frac{E[Y_{t+1}]}{K_t} - \delta, \tag{94}$$

$$1 + r_t = E\left[\frac{P_{t+1}^A}{P_t^A}(1-\delta^{RD}) + \frac{1}{P_t^A}\gamma\frac{Y_{t+1}}{A_t}\right]. \tag{95}$$

A R&D capital producer maximizes profits by choice of investment S_t

$$max_{\{S_t\}}\left[P_t^A A_t - P_t^A(1-\delta^{RD})A_{t-1} - (1-s_t^{RD})S_t\right], \tag{96}$$

subject to the R&D accumulation equation

$$A_t = \left[1 - \delta^{RD} + \upsilon\left(\frac{S_t}{A_t}\right)e^{u_t}\right]A_{t-1}, \tag{97}$$

where $\upsilon(.)$ is a function on the form

$$\upsilon\left(\frac{S_t}{A_t}\right) = g + \frac{\theta}{1-\frac{1}{\zeta}}\left(\frac{S_t}{A_t}\right)^{1-\frac{1}{\zeta}}. \tag{98}$$

We assume $u_t \sim iid(0,\sigma^2)$. The R&D government subsidy s_t^{RD} is stochastic and follows a AR(1) process:

$$log(s_t^{RD}) = (1-\lambda)log(s_{ss}^{RD}) + \lambda log(s_{t-1}^{RD}) + \epsilon_t. \tag{99}$$

The optimality condition for R&D capital producer is:

$$P_t^A\frac{A_{t-1}}{A_t}\theta\left(\frac{S_t}{A_t}\right)^{-\frac{1}{\zeta}}e^{u_t} = 1 - s_t^{RD}. \tag{100}$$

The government finances R&D subsidies $s_t^{RD} S_t$ using lump-sump taxes T_t and issuing one period real bonds B_t. The budget constraint is

$$B_t - (1 + r_{t-1})B_{t-1} + T_t = s_t^{RD} S_t, \tag{101}$$

(93)-(95) , (89), (90), (97) , (99), (100), (101) determine the equilibrium.

In this model both u_t and s_t^{RD} are stationary but A_t is a unit root with endogenous drift. Applying the rules of Section 3 one finds that output, consumption, physical investment and physical capital, R&D investment and R&D capital all grow at the same rate which is given by $(1-\delta^{RD}+\upsilon(s_{ss}))$, where $s_{ss} = \frac{S^{ss}}{A^{ss}}$. Hence for these six variables $H^i(A_t) = A_t$, and the G^i functions are given by

$$Y_{ss}^{\Delta} = \gamma(A_{ss}^{\Delta} + L_{ss}^{\Delta}) + (1-\gamma)K_{ss}^{\Delta}. \tag{102}$$

$$A_{ss}^{\Delta} = log(1 - \delta^d + \upsilon\,(s_{ss})). \tag{103}$$

$$K_{ss}^{\Delta} = I_{ss}^{\Delta}. \tag{104}$$

$$Y_{ss}^{\Delta} = C_{ss}^{\Delta} \tag{105}$$

$$r_{ss}^{\Delta} = 0, \tag{106}$$

$$S_{ss}^{\Delta} = A_{ss}^{\Delta}. \tag{107}$$

The stationary system is represented by eight equations:

Production function:

$$y_t = \left(\frac{1}{\Delta a_t}\right) L_t^{\gamma} \left(\frac{k_{t-1}}{\Delta k_t}\right)^{1-\gamma}. \tag{108}$$

Capital accumulation equation:

$$k_t = (1-\delta)\frac{k_{t-1}}{\Delta k_t} + i_t, \tag{109}$$

Market clearing condition:

$$y_t = c_t + i_t + s_t, \tag{110}$$

Euler equation:

$$\frac{E[c_{t+1}\Delta c_{t+1}]}{c_t} = \beta(1+r), \tag{111}$$

Definition of the real rate:

$$(1-\gamma)\frac{E[y_{t+1}\Delta y_{t+1}]}{k_t} = r + \delta. \tag{112}$$

Optimality condition with respect to R&D capital:

$$1 + r_t = \frac{E[p_{t+1}^A]}{p_t^A}(1-\delta^{RD}) + \frac{1}{p_t^A}\gamma E[y_{t+1}\Delta y_{t+1}] \tag{113}$$

Optimality condition for the capital producer:

$$\frac{p_t^A}{\Delta a_t} \upsilon\left(s_t\right)' e^{u_t} = 1 - s_t^{RD}. \tag{114}$$

Finally, the R&D capital accumulation equation:

$$\Delta a_t \equiv \frac{A_t}{A_{t-1}} = \left[1 - \delta^d + \upsilon\left(s_t\right) e^{u_t}\right]. \tag{115}$$

We compute the linear approximation to the solution of this system and the responses to an impulse in the government subsidy s_t^{RD}. We repeat the exercise using the mechanical approach. The file with the automatically stationarized model is in Appendix J. Figure 4 shows that the two solutions are identical.

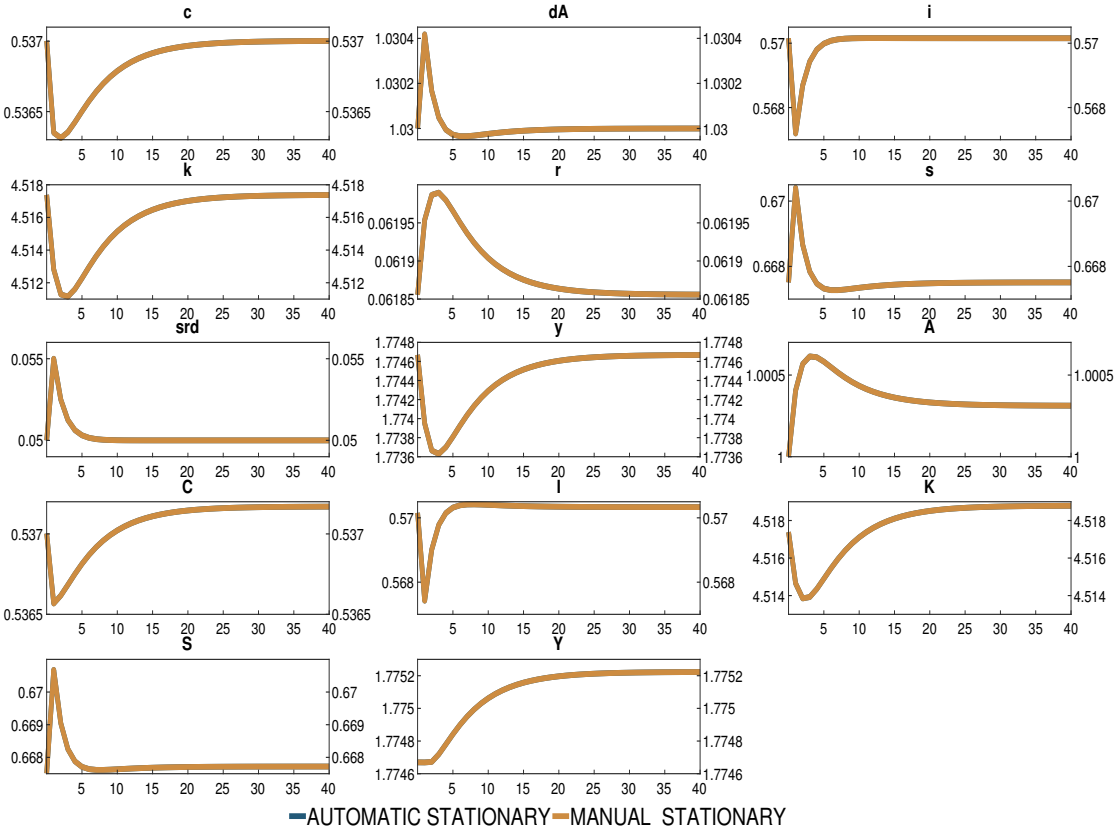

Figure 4: Responses to a subsidy shock

6 Comparison with other packages

Although other toolboxes have the capabilities to deal with non-stationary models, they are more restrictive in what the user is allowed to do and require crucial inputs in the stationarization process.

DYNARE has a symbolic facility designed to transform the first order conditions of a non-stationary model. However, to implement the transformation the user needs to provide both the growth factors and the endogenous variables that are trending. Furthermore, at least in its current format, DYNARE only accepts exogenous unit root process and can not deal e.g. with deterministic trends or endogenous growth. In Appendix K we provide a DYNARE code to solve the RBC example of Section 5.1 and the responses which as generated by an impulse in the labour augmenting disturbances.

IRIS, on the other hand, uses a numerical approach to transform the first-order conditions of the problem into a stationary format. Relative to IRIS, our algorithm has a number of advantages. First with IRIS the user needs to provide the growth path around which the numerical approximation is taken; this is automatically found with our approach. Second, we use a symbolic approach, which is free from numerical approximation error; third, our setup gives the user the possibility to inspect the G^i functions and understand how the model has been made stationary; finally the stationary model our approach produces can be used in other packages.

7 Conclusion

As Lafourcade and de Wind (2012) have noticed "... setting up, solving and estimating a DSGE that includes multiple theoretically-founded trends is a non-trivial exercise". The algorithm we provide makes this process less painful and less subject to algebraic mistakes. The approach is simple and straightforward; it allows a researcher to compare the consequences of different trend assumptions and to select the one which is more likely to characterize the available data. All of this, without tedious algebra and the risk of having to redo the computations several times. The algorithm we propose uses symbolic language and prepackaged MATLAB routines to generate the stationary system. Once this is obtained the model can be solved and standard dynamic analysis performed.

The codes used in the paper belong to the NB toolbox developed by Kenneth S. Paulsen. The toolbox is freely available, and downloadable at `https://github.com/Coksp1/NBTOOLBOX/`. The same location contains the files needed to replicate the examples of Sections 5.1, 5.2, 5.3 (`https://github.com/Coksp1/NBTOOLBOX/tree/main/Examples/Econometrics/DSGE/Non_stationary_model`).

References

Blanchard, O. J. and Kahn, C. M. (1980) The Solution of Linear Difference Models under Rational Expectations, *Econometrica*, **48**, 1305–1311.

Canova, F. (2014) Bridging dsge models and the raw data, *Journal of Monetary Economics*, **67**, 1–15.

Canova, F. and Matthes, C. (2021) A composite likelihood approach fro dynamic structural models, *Economic Journal*, **131**, 2447–2477.

Comin, D. and Gertler, M. (2006) Medium term business cycles, *American Economic Review*, **96**, 523–551.

Dynare (2021) https://www.dynare.org/, Dynare, CEPREMAP.

GPMN (2020) https://iris.igpmn.org/, Iris macroeconomic modeling toolbox, Global Projection Model Network.

Justiniano, A., Primiceri, G. and Tambalotti, A. (2011) Investment shocks and the relative price of investment, *Review of Economic Dynamics*, **14**, 101–121.

Klein, P. (2000) Using the generalized shur form to solve a multivariate linear rational expectation model, *Journal of Economic Dynamics and Control*, **24**, 1405–1423.

Lafourcade, P. and de Wind, J. (2012) Taking Trends Seriously in DSGE Models: An Application to the Dutch Economy, DNB Working Papers 345, Netherlands Central Bank, Research Department.

Maih, J. (2021) https://github.com/jmaih/RISE_toolbox, Rise, Norges Bank.

Paulsen, K. S. (2021) https://github.com/Coksp1/NBTOOLBOX, toolbox, Norges Bank.

Sims, C. (2010) Gensys, manuscript, Princeton University.

A Description of how the algorithm is implemented

The algorithm utilizes different classes existing in the NB toolbox:

1. The model file is parsed, and written into a MATLAB function handle.

2. The process of finding the restrictions imposed on the balanced growth path is performed utilizing the `nb_bgrowth` class. First, endogenous and exogenous variables, unit root process and parameters are transformed to a set of `nb_bgrowth` objects which are passed to the MATLAB function handle representing the equations of the model. The result is the set of restrictions holding along the balanced growth path.

3. The initial solution for the balanced growth path is obtained by transforming the equations found in step 2 to a linear system. From this solution it is possible to identify the R non-stationary variables.

4. The identification of the R the G^i functions is done with the function `nb_findBasis`. To get the symbolic representation of the G^i's, we first construct a function handle representing these equations, and then utilize the `nb_term` class and subclasses to transform it into symbolic equations.

5. The stationary representation of the model is constructed using `nb_stTerm`. This requires transforming the endogenous and exogenous variables and unit root variables to class `nb_stTerm` objects using the solution for the balanced growth. The parameters are transformed to `nb_stParam` objects. Then these objects are passed to the MATLAB function handle representing the non-stationary equations. The result is the stationary representation written in the model file syntax.

6. With the stationary representation, the model can be solved in a standard way. The examples presented in the paper utilizes the `myAD` [2], or `nb_mySD` classes. Again, endogenous and exogenous variables, unit root process and parameters are transformed to a set of objects which are passed to the MATLAB function handle representing the stationary equations of the model. The result is a set of matrices representing the linearized system, which is solved by Klein (2000) algorithm.

Step 1 uses the `nb_dsge` class constructor; steps 2-4 the `nb_dsge.solveBalancedGrowthPath` method[3], while step 5 employs the `nb_dsge.stationarize` method. The final step is performed by the `nb_dsge.solve` method. The following command allow one to perform model-based analysis once the solution is found:

IRFs Produces impulse response functions using the method `nb_dsge.irf`.

Theoretical moments Produce the theoretical moments of the model using the method `nb_dsge.theoreticalMome`

[2] Original code is written by SeHyoun Ahn, and can be found here https://github.com/sehyoun/MATLABAutoDiff

[3] A method is a function that is related to a class. For example the `nb_dsge.solveBalancedGrowthPath` method is a function that act on an object of class `nb_dsge`.

Filter If `nb_dsge` is assigned to data, using the `nb_dsge.set` method, you can run the Kalman filter using the `nb_dsge.filter` method. See also the `nb_dsge.getFiltered` method.

Forecast If the model is filtered, forecast can be produced using the `nb_dsge.forecast` method. See also `nb_dsge.getForecast` and `nb_dsge.plotForecast` methods.

B The details of the RBC model

The production function is

$$Y_t = (A_t L_t)^{\gamma} K_{t-1}^{1-\gamma}, \tag{116}$$

where Y_t is output, $L_t = 1$ is labor, K_t is capital, A_t is the labor augmenting productivity, and γ is the labor share. Let I_t be investment and δ is the depreciation rate. Capital accumulates according to

$$K_t = (1 - \delta)K_{t-1} + I_t, \tag{117}$$

A representative household maximizes the discounted utility

$$E\left[\sum_{s=0}^{\infty} \beta^s U(C_{t+s})\right], \tag{118}$$

where β is the discount factor, subject to (116), (117) and the constraint

$$B_t + Y_t = C_t + I_t(1 + r_{t-1})B_{t-1} + T_t, \tag{119}$$

where B_t are one period bond issued by the government. The government collects lump sum taxes, T_t, from the households.

The optimality condition is

$$\frac{E[C_{t+1}]}{C_t} = \beta(1 + r_t), \tag{120}$$

where r_t, the real interest rate, is given by

$$r_t = (1 - \gamma)\frac{E[Y_{t+1}]}{K_t} - \delta. \tag{121}$$

Markets must also clear

$$Y_t = C_t + I_t. \tag{122}$$

(120)-(121) together with (116), (117) , (122), determine the equilibrium.

The stationary model file

```
endogenous
y r l k i dA c D_Z_y D_Z_k D_Z_i D_Z_c D_Z_A

exogenous
u

parameters
std_u lambda gamma g delta beta

model

(c(+1)*D_Z_c(+1))/c = beta*(1+r);
l-1;
y = ((1*l)^gamma)*((k(-1)*D_Z_k^-1)^(1-gamma));
((1-gamma)*(y(+1)*D_Z_y(+1)))/k = r+delta;
k = (1-delta)*(k(-1)*D_Z_k^-1)+i;
y = c+i;
dA = 1/(1*D_Z_A^-1);
1/(1*D_Z_A^-1) = ((g^(1-lambda))*(((1*D_Z_A^-1)/(1*D_Z_A^-1*D_Z_A(-1)^-1))^lambda))*(exp(std_u*u));
gamma*log(D_Z_k)+log(D_Z_y)-gamma*log(D_Z_A)-log(D_Z_k);
log(D_Z_y)-log(D_Z_k);
log(D_Z_k)-log(D_Z_i);
log(D_Z_y)-log(D_Z_c);

reporting
A=exp(cumsum(log(dA) - log(steady_state(dA))));
C=c*A;
I=i*A;
K=k*A;
Y=y*A;
c_log_dev=exp(log(c) - log(steady_state(c)));
i_log_dev=exp(log(i) - log(steady_state(i)));
k_log_dev=exp(log(k) - log(steady_state(k)));
r_dev=r - steady_state(r);
y_log_dev=exp(log(y) - log(steady_state(y)));
```

C The details of the New Keynesian model

Final good sector

The final goods sector combines intermediate goods, $Q_t(n)$ into a final good A_t sold at a price P_t. The production function is:

$$Q_t = \left[\int_0^1 Q_t(n)^{1-\frac{1}{\theta_t^H}} dn\right]^{\frac{\theta_t^H}{\theta_t^H-1}}. \tag{123}$$

where θ_t^H is the elasticity of substitution between different intermediate goods, and it assumed to follow an AR process. The optimal combination of $Q_t(n)$ is found by cost minimization

$$\min_{\{Q_t(n)\}} \int_0^1 P_t^Q(n) Q_t(n) dn,$$

subject to (123). The resulting optimality condition is:

$$Q_t(n) = \left(\frac{P_t^Q(n)}{P_t^Q} \right)^{-\theta_t^H} Q_t, \tag{124}$$

where

$$P_t^Q = \left[\int_0^1 P_t^Q(n)^{1-\theta_t^H} dn \right]^{\frac{1}{1-\theta_t^H}}. \tag{125}$$

Final good firms maximize the profit function

$$PA_t - P_t^Q Q_t, \tag{126}$$

subject to

$$A_t = Q_t. \tag{127}$$

The optimality condition is

$$P_t = P_t^Q. \tag{128}$$

Intermediate goods sector

Intermediate goods firms use effective capital and labor to produce a good which is sold under monopolistic competition to the final goods sector. The intermediate firm n has the production function

$$Q_t(n) = (Z_t z_t^L L_t(n))^{1-\alpha} \overline{K}_t(n)^{\alpha}, \tag{129}$$

where $\alpha \in [0,1]$ is the capital share, $L_t(n)$ and $\overline{K}_t(n)$ denote hours and effective capital of firm n in period t. There are two shocks to productivity: Z_t, a permanent labor-augmenting technology process, growing at the gross rate π_t^z, and z_t^L, a temporary shock to productivity (or labor utilization). Total labor input to firm n aggregate labor inputs from all households j:

$$L_t(n) = \left[\int_0^1 L_t(n,j)^{1-\frac{1}{\psi_t}} dj \right]^{\frac{\psi_t}{\psi_t - 1}}, \tag{130}$$

where ψ_t is the elasticity of substitution between differentiated labor. Let W_t be the wage rate and R_t^K the rental rate. Minimizing total factor outlays:

$$W_t L_t(n) + R_t^K \overline{K}_t(n),$$

subject to 129 leads to the following first order conditions

$$L_t = (1-\alpha)\frac{MC_t}{W_t} Q_t(n), \tag{131}$$

$$\overline{K}_t = \alpha \frac{MC_t}{R_t^K} Q_t(n), \tag{132}$$

where MC_t is marginal cost. In symmetric equilibrium all n firms make the same decision, so $L_t = L_t(n)$, $\overline{K}_t = \overline{K}_t(n)$ and $Q_t = Q_t(n)$. Firm n will minimize labor costs subject to the optimal level of labor input:

$$\min_{l_{I,t}(n,j)} \int_0^1 L_t(n,j) W_t(j) dj, \tag{133}$$

subject to

$$\left[\int_0^1 L_t(n,j)^{1-\frac{1}{\psi_t}} dj \right]^{\frac{\psi_t}{\psi_t - 1}} = L_t(n). \tag{134}$$

Thus the conditional labor demand functions facing household j is

$$L_t(j) = \left(\frac{W_t(j)}{W_t} \right)^{-\psi_t} L_t. \tag{135}$$

Firms in the intermediate sector sell their goods monopolistically. Each firm n charges the $P_t^Q(n)$. Profits paid out as dividends to households) are:

$$\Pi_t(n) = P_t^Q(n) Q_t(n) - W_t L_t(n) - R_t^K(n) \overline{K}_t(n). \tag{136}$$

The costs of adjusting prices are

$$\gamma_{PQ,t}(n) \equiv \frac{\phi^{PQ}}{2} \left[\frac{P_t^Q(n) / P_{t-1}^Q(n)}{P_{t-1}^Q / P_{t-2}^Q} - 1 \right]^2, \tag{137}$$

where $P_t^Q = \left[\int_0^1 P_t^Q(n)^{1-\theta_t^H} dn \right]^{\frac{1}{1-\theta_t^H}}$. The costs of changing prices is governed by the parameter ϕ^{PQ}. These costs are assumed to be intangible. The intermediate firm n faces the demand function: $Q_t(n) = \left(\frac{P_t^Q(n)}{P_t^Q} \right)^{-\theta_t^H} Q_t$ Optimal price setting for firm n requires maximizing [4]:

[4] To make the expression easier to work with, the costs of adjusting prices are linear in $P_t^Q Q_t$ and not $P_t^Q(n) Q_t(n)$

$$\Pi_s = \sum_{t=s}^{\infty} \Delta_{s,t} \left\{ \begin{array}{c} P_t^Q(n) \left(\frac{P_t^Q(n)}{P_t^Q}\right)^{-\theta_t^H} Q_t - MC_t \left[\left(\frac{P_t^Q(n)}{P_t^Q}\right)^{-\theta_t^H} Q_t \right] \\ -\gamma_{PQ,t}(n) P_t^Q Q_t \end{array} \right\}.$$

In symmetric equilibrium, all firms will behave the same, and thus:

$$\begin{aligned} Q_t - \theta_t^H Q_t + MC_t \theta_t^H \frac{Q_t}{P_t^Q} - \phi^{PQ} \left[\frac{P_t^Q/P_{t-1}^Q}{P_{t-1}^Q/P_{t-2}^Q} - 1 \right] \frac{P_t^Q/P_{t-1}^Q}{P_{t-1}^Q/P_{t-2}^Q} Q_t \\ + E_t \left\{ \Delta \phi^{PQ} \left[\frac{P_{t+1}^Q/P_t^Q}{P_t^Q/P_{t-1}^Q} - 1 \right] \frac{(P_{t+1}^Q/P_t^Q)^2}{P_t^Q/P_{t-1}^Q} Q_{t+1} \right\} = 0. \end{aligned} \tag{138}$$

Investment producer

Perfectly competitive firms purchase Y_t^I units from the final good producers, and produce investment goods I_t according to the production function

$$I_t = \Upsilon_t Y_t^I. \tag{139}$$

Υ_t represents investment-specific technological (IST) progress, which is specified later, and I_t is the investment good sold to capital producers. The objective of investment producers is to maximize the profit function

$$P_t^I I_t - P_t Y_t^I, \tag{140}$$

subject to (139). The optimal condition is:

$$\Upsilon_t P_t^I = P_t. \tag{141}$$

Capital producer

Capital goods, K_t^{new}, are produced by a sector which purchases investment goods I_t and transform them into capital, sold to households at price P_t^K. The production function is:

$$K_t^{new} = z_{I,t} \left(1 - S\left(\frac{I_t}{I_{t-1}}\right)\right) I_t. \tag{142}$$

S captures the presence of adjustment costs in investment. $z_{I,t}$ is the marginal efficiency of investment shock process. The objective of capital producers is to maximize the expected discounted future profits:

$$E_s \sum_{t=s}^{\infty} \Delta_{s,t} \left[P_t^K K_t^{new} - P_t^I I_t \right], \tag{143}$$

subject to (139). Optimality implies:

$$E\left[\Delta P_{t+1}^{K} z_{I,t+1} S_2^{'}\left(\frac{I_{t+1}}{I_t}\right) I_{t+1}\right] + P_t^K z_{I,t}\left[1 - S\left(\frac{I_t}{I_{t-1}}\right) - S_1^{'}\left(\frac{I_t}{I_{t-1}}\right) I_t\right] - P_t^I = 0, \tag{144}$$

where we assume that

$$S\left(\frac{I_t}{I_{t-1}}\right) = \frac{\phi_{I1}}{2}\left[\frac{I_t}{I_{t-1}} - \left(\pi_{ss}^{\Upsilon}\right)^{\frac{1}{1-\alpha}} \pi_{ss}^{z}\right]^2, \tag{145}$$

Thus

$$S_1^{'}\left(\frac{I_t}{I_{t-1}}\right) = \phi_{I1}\left[\frac{I_t}{I_{t-1}} - \left(\pi_{ss}^{\Upsilon}\right)^{\frac{1}{1-\alpha}} \pi_{ss}^{z}\right]\frac{1}{I_{t-1}}, \tag{146}$$

$$S_2^{'}\left(\frac{I_t}{I_{t-1}}\right) = -\phi_{I1}\left[\frac{I_t}{I_{t-1}} - \left(\pi_{ss}^{\Upsilon}\right)^{\frac{1}{1-\alpha}} \pi_{ss}^{z}\right]\frac{I_t}{I_{t-1}^2}, \tag{147}$$

Households

Each household supplies a differentiated labor input to intermediate firms and set wages under the assumption of monopolistic competition. Households obtain utility from consumption and leisure. Preferences are additively separable. Lifetime expected utility of household j at time s is

$$U_s\left(j\right) = E_s \sum_{t=s}^{\infty} \beta^{t-s} \rho_t \left[u\left(C_t\left(j\right)\right) - v(L_t\left(j\right))\right], \tag{148}$$

where β is the discount factor, ρ_t is a discount factor shock, C_t denotes consumption, and L_t labor. The in-period utility function is

$$u\left(C_t\left(j\right)\right) = z_t^u\left(1 - \frac{b^c}{\left(\pi_{ss}^{\Upsilon}\right)^{\frac{\alpha}{1-\alpha}} \pi_{ss}^{z}}\right) \ln\left[\frac{C_t\left(j\right) - b^c C_{t-1}}{\left(1 - \frac{b^c}{\left(\pi_{ss}^{\Upsilon}\right)^{\frac{\alpha}{1-\alpha}} \pi_{ss}^{z}}\right)}\right], \tag{149}$$

$$\nu\left(L_t\left(j\right)\right) = \frac{1 - b^l}{1 + \zeta}\left[\frac{L_t\left(j\right) - b^l L_{t-1}}{1 - b^l}\right]^{1+\zeta}, \tag{150}$$

b^c governs habit persistence, π_{ss}^{z} denotes the steady-state labor augmenting technology growth rate and π_{ss}^{Υ} the steady-state investment-specific technology growth rate. The degree of disutility of supplying labor is captured by $\zeta > 0$, the inverse Frisch elasticity. As (149) indicates, we assume a log in-period utility function for consumption. This insure the existence of a balanced growth path. The household's budget constraint is

$$
\begin{aligned}
&P_tC_t(j) + P_t^K K_t^{new}(j) + P_tB_t(j) + P_t\gamma(U_t)K_{t-1} \\
&\quad + W_t(j)L_t(j)\gamma_t(j) + T_t(j) \\
&\leq R_{t-1}P_{t-1}B_{t-1}(j) + W_t(j)L_t(j) + R_t^K U_t K_{t-1} \\
&\quad + Q_t(j) + \Pi_t(j) + DIV_t(j),
\end{aligned} \tag{151}
$$

where P_t is the price level of final goods, R_t is the gross interest rate, $B_t(j)$ is real household's borrowing, $W_t(j)$ is the nominal wage rate set by household j, $\gamma_t(j)$ is intangible wage adjustment costs (defined in (156)), $L_t(j)$ is the total hours worked, and $DIV_t(j)$, $\Pi_t(j)$ and $T_t(j)$ are is the net cash flow from household's j portfolio of state contingent securities, profits (in nominal terms) paid out and taxes paid, respectively. Households rent out utilized capital U_tK_{t-1} at the rate R_t^K, where U_t is the utilization rate and K_{t-1} the capital stock at the end of last period. We assume intangibility of the utilization cost. $K_t^{new}(j)$ is new capital bought from the capital producers at price P_t^K and $\gamma(U_t)K_{t-1}$ is a cost specified and:

$$
\gamma(U_t) = \frac{R_{ss}^K}{P_{ss}\phi_u}\left[e^{\phi_u(U_t-1)} - 1\right], \tag{152}
$$

where ϕ_u governs the cost of adjusting utilization. Time t utilized capital iis

$$
\overline{K}_t = U_tK_{t-1}. \tag{153}
$$

The physical capital accumulation equation is

$$
K_t = (1-\delta)K_{t-1} + K_t^{new}. \tag{154}
$$

Household j faces the following labor demand curve:

$$
L_t(j) = \left(\frac{W_t(j)}{W_t}\right)^{-\psi_t} L_t, \tag{155}
$$

where W_t is the wage rate. We further assume that there is sluggish wage adjustment due to resource costs measured in terms of the total wage bill. We deviate from JPT, and assume that wage adjustments costs are

$$
\gamma_t(j) \equiv \frac{\phi^W}{2}\left[\frac{W_t(j)/W_{t-1}(j)}{W_{t-1}/W_{t-2}} - 1\right]^2. \tag{156}
$$

Thus, costs are related to changes in wage inflation relative to the past observed rate for households. $\phi^W > 0$ determines how costly it is to change the wage inflation rate. Maximizing (148) subject to the budget constraint leads to the following optimal conditions in symmetric equilibrium

$$E_t\left[\beta\frac{\rho_{t+1}u'\left(C_{t+1}\right)}{\rho_t u'\left(C_t\right)}\frac{P_t}{P_{t+1}}\right]R_t=1 \tag{157}$$

$$\begin{aligned}\frac{v'\left(L_t\right)}{u'\left(C_t\right)}\psi_t\frac{P_t}{W_t}&=\left[\left(\psi_t-1\right)\left(1-\gamma_t\right)+\phi^W\left(\frac{W_t/W_{t-1}}{W_{t-1}/W_{t-2}}-1\right)\frac{W_t/W_{t-1}}{W_{t-1}/W_{t-2}}\right]\\&-E_t\left[\beta\frac{\rho_{t+1}u'\left(C_{t+1}\right)}{\rho_t u'\left(C_t\right)}\frac{P_t}{P_{t+1}}\frac{L_{t+1}}{L_t}\phi^W\left(\frac{W_{t+1}/W_t}{W_t/W_{t-1}}-1\right)\frac{\left(W_{t+1}/W_t\right)^2}{W_t/W_{t-1}}\right],\end{aligned} \tag{158}$$

$$\rho_t u'\left(C_t\right)\frac{P_t^K}{P_t}=E_t\left[\beta\rho_{t+1}u'\left(C_{t+1}\right)\left(\frac{P_{t+1}^K}{P_{t+1}}\left(1-\delta\right)+\frac{R_{K,t+1}}{P_{t+1}}U_{t+1}-\gamma\left(U_{t+1}\right)\right)\right], \tag{159}$$

$$\frac{R_{K,t}}{P_t}=\gamma'\left(U_t\right)\frac{R_{K,ss}}{P_{ss}}e^{\phi_u\left(U_t-1\right)}. \tag{160}$$

Final good market clearing and definitions

$$\widetilde{A}_t=\widetilde{C}_t\ +\widetilde{Y}_t^I. \tag{161}$$

Inflation

$$\pi_t\equiv\frac{P_t}{P_{t-1}}. \tag{162}$$

Intermediate goods inflation

$$\pi_t^Q=\frac{P_t^Q}{P_{t-1}^Q}. \tag{163}$$

Investment price inflation

$$\pi_t^I=\frac{P_t^I}{P_{t-1}^I}. \tag{164}$$

Capital price inflation

$$\pi_t^K=\frac{P_t^K}{P_{t-1}^K}. \tag{165}$$

Wage inflation

$$\pi_t^W=\frac{W_t}{W_{t-1}}. \tag{166}$$

Output

$$Y_{NAT,t}=A_t. \tag{167}$$

Output growth

$$\Delta Y_{NAT,t} = \frac{A_t}{A_{t-1}}. \tag{168}$$

The Monetary policy rule is

$$R_t = (R_{t-1})^{\omega_R} \left(R_{ss} \left(\frac{\pi_t}{\pi_{ss}} \right)^{\omega_P} \left(\frac{\widetilde{Y}_{NAT,t}}{\widetilde{Y}_{NAT,ss}} \right)^{\omega_Y} \left(\triangle \widetilde{Y}_{NAT,t} \right)^{\omega_{\triangle Y}} \right)^{1-\omega_R} e^{z_{R,t}}, \tag{169}$$

where ω_R ω_P, ω_Y and $\omega_{\triangle Y}$ are the weights on past nominal rate, inflation, output gap and output growth; and $z_{R,t}$ is a monetary policy shock.

The shock processes

Labor augmenting technology shock:

$$\log \left(\frac{Z_t}{Z_{t-1}} \right) = \log(\pi_t^z) = (1 - \lambda_{\pi^z}) \log(\pi_{ss}^z) + \lambda_{\pi^z} \log(\pi_{t-1}^z) + \epsilon_{\pi^z,t} \sigma_{\pi^z}. \tag{170}$$

Investment-specific technological shock:

$$\log \left(\frac{\Upsilon_t}{\Upsilon_{t-1}} \right) = \log(\pi_t^\Upsilon) = (1 - \lambda_{\pi^\Upsilon}) \log(\pi_{ss}^\Upsilon) + \lambda_{\pi^\Upsilon} \log(\pi_{t-1}^\Upsilon) + \epsilon_{\pi^\Upsilon,t} \sigma_{\pi^\Upsilon}. \tag{171}$$

Intermediate good sector productivity shock:

$$\log(z_t^L) = (1 - \lambda_{z^L}) \log(z_{ss}^L) + \lambda_{z^L} \log(z_{t-1}^L) + \epsilon_{z^L,t} \sigma_{z^L}. \tag{172}$$

Labor market competition shock:

$$\log(\psi_t) = (1 - \lambda_\psi) \log(\psi_{ss}) + \lambda_\psi \log(\psi_{t-1}) + \epsilon_{\psi,t} \sigma_\psi. \tag{173}$$

Intermediate good market competition shock:

$$\log(\theta_t^H) = (1 - \lambda_{\theta^H}) \log(\theta_{ss}^H) + \lambda_{\theta^H} \log(\theta_{t-1}^H) + \epsilon_{\theta^H,t} \sigma_{\theta^H}. \tag{174}$$

Marginal efficiency of investment shock:

$$\log(z_{I,t}) = (1 - \lambda_I) \log(z_{I,ss}) + \lambda_I \log(z_{I,t-1}) + \epsilon_{I,t} \sigma_I. \tag{175}$$

Monetary policy shock:

$$\log(z_{R,t}) = (1 - \lambda_R) \log(z_{R,ss}) + \lambda_R \log(z_{R,t-1}) + \epsilon_{R,t} \sigma_R. \tag{176}$$

Consumption preference shock:

$$\log(z_t^u) = (1 - \lambda_{z^u}) \log(z_{ss}^u) + \lambda_{z^u} \log(z_{t-1}^u) + \epsilon_{z^u,t} \sigma_{z^u}. \tag{177}$$

Discount factor shock:

$$\log(\rho_t) = (1 - \lambda_\rho) \log(\rho_{ss}) + \lambda_\rho \log(\rho_{t-1}) + \epsilon_{\rho,t} \sigma_\rho. \tag{178}$$

where λ_j regulates persistence and σ_j the standard deviation, $j = 1 \ldots, 9$

D The manually made stationary file

```
endogenous
A_NW % Final goods production
C_NW % Consumption
DPQ_C_NW % Consumption growth
DPQ_I_NW % Investment growth
DPQ_K_NW % Capital growth
DPQ_P_NW % Inflation
DPQ_PQ_NW % Intermidate goods inflation
DPQ_REAL_PI_NW % Real investment inflation
DPQ_REAL_W_NW % Real wage inflation
DPQ_W_NW % Wage inflation
DPQ_Y_NW % Output growth
DSA_NW % Stochastic discount factor
DUT_NW % Investment-specific technological progress
DZT_NW % Permanent labor-augmenting technology process
GAMMAPRIME_U_NW % Marginal cost of utilizing the capital
GAMMA_U_NW % Cost of utilize the capital
GAMMA_W_NW % Wage adjustment cost
I_NW % Investment
K_NW % Capital
KBAR_NW % Utilized capital
KNEW_NW % Capital goods produced each period
L_NW % Hours worked
MC_NW % Marginal cost
MRS_NW % Marginal rate of substitution
NAT_Y_NW % Output
PSI_NW % Competition in the labor market shock process
Q_NW % Demand for intermediate goods
REAL_PI_NW % Real price of investment
REAL_PK_NW % Real price of capital
REAL_PQ_NW % Real intermediate goods price
REAL_W_NW % Real wage rate
RHO_NW % Discount factor shock process
RK_NW % Rental rate of capital
RN3M_NW % Money market interest rate
S_NW % Investment adjustment cost function
S_PRIME1_NW % Derivative of the investment adjustment
% cost functcion wrt 1st input times I_NW
S_PRIME2_NW % Derivative of the investment adjustment
%cost function wrt 2nd input times I_NW
T_NW % Intermediate goods production
THETAH_NW % Price markup shock process
U_NW % Utilization rate
UPRIME_NW % Derivative of the utility function of
% households wrt consumption
VPRIME_NW % Derivative of the utility function of
% households wrt labor
Y_I_NW % Input to investment production
Z_I_NW % Marginal efficiency of investment shock process
Z_L_NW % Temporary labor augmenting technology shock process
```

```
Z_RN3M_NW % Monetary policy shock process
Z_U_NW % Consumption preference shock process

exogenous
E_DUT_NW  % Investment-specific technological innovation
E_DZT_NW % Permanent labor-augmenting technology innovation
E_I_NW % Marginal efficiency of investment innovation
E_L_NW % Temporary labor augmenting technology innovation
E_PSI_NW % Competition in the labor market innovation
E_RHO_NW % Discount factor innovation
E_RN3M_NW % Monetary policy innovation
E_THETAH_NW % Price markup innovation
E_U_NW % Consumption preference innovation

parameters
ALPHA_NW % Capital share
BC_NW % Habit in consumption
BETA_NW % Discount factor
BL_NW % Habit in hours worked
DELTA_NW % Depreciation rate
DPQ_P_NW_SS % Steady-state inflation
DUT_NW_SS % Steady-state growth rate in
% investment-specific technology
DZT_NW_SS % Steady-state growth rate in
% labor-augmenting technology
LAMBDA_DUT_NW % Shock persistent parameter for the
% investment-specific technology shock
LAMBDA_DZT_NW % Shock persistent parameter for the
% labor-augmenting technology shock
LAMBDA_I_NW % Shock persistent parameter for the
% marginal efficiency of investment shock
LAMBDA_L_NW % Shock persistent parameter for the
% temporary labor augmenting technology shock
LAMBDA_PSI_NW % Shock persistent parameter for the
% competition in the labor market shock
LAMBDA_RHO_NW % Shock persistent parameter for
% the discount factor shock
LAMBDA_RN3M_NW % Shock persistent parameter for
% the monetary policy shock
LAMBDA_U_NW % Shock persistent parameter for the
% price markup shock
LAMBDA_THETAH_NW % Shock persistent parameter for the
% consumption preference shock
OMEGA_Y_NW % Taylor rule coefficient on ouput gap.
OMEGA_DPQ_Y_NW  % Taylor rule coefficient on ouput growth gap.
OMEGA_P_NW % Taylor rule coefficient on inflation gap.
OMEGA_R_NW % Interest rate smoothing in the taylor rule
PHI_PQ_NW % Intermediate goods price adjustment cost parameter
PHI_I1_NW % Investment adjustment cost parameter
PHI_W_NW % Wage adjustment cost parameter
PHI_U_NW % Capital utilization cost parameter
PSI_NW_SS % Steady-state elasticity of substitution between
```

```
% differentiated labor
RHO_NW_SS % Discount factor shock process in steady state
THETAH_NW_SS % Steady-state elasticity of substitution
% between intermidate goods
ZETA_NW % Inverse Frisch elasticity
Z_I_NW_SS % Marginal efficiency of investment shock in
% steady-state
Z_L_NW_SS % Temporary labor augmenting technology shock
% in steady-state
Z_RN3M_NW_SS % Monetary policy shock in steady-state
Z_U_NW_SS % Consumption preference shock in steady-state
std_E_DUT_NW % Standard deviation of the innovation to
% the Z_DUT_NW shock process
std_E_DZT_NW % Standard deviation of the innovation to
% the Z_DZT_NW shock process
std_E_I_NW % Standard deviation of the innovation to
% the Z_I_NW shock process
std_E_L_NW % Standard deviation of the innovation to
% the Z_L_NW shock process
std_E_PSI_NW % Standard deviation of the innovation to
% the PSI_NW shock process
std_E_RHO_NW % Standard deviation of the innovation to
% the RHO_NW shock process
std_E_RN3M_NW % Standard deviation of the innovation to
% the Z_RN3M_NW shock process
std_E_THETAH_NW % Standard deviation of the innovation to
% the THETAH_NW shock process
std_E_U_NW % Standard deviation of the innovation to
% the Z_U_NW shock process

model

% 1) Final goods sector
% Production function (A_NW)
A_NW = Q_NW;

% FOC (Q_NW)
REAL_PQ_NW = 1;

% 2) Intermediate goods
% Intermidate production function (KBAR_NW)
T_NW = (Z_L_NW*L_NW)^(1-ALPHA_NW)*KBAR_NW^ALPHA_NW;

% Optimality condition wrt utilized capital (I.e. demand function) (MC_NW)
KBAR_NW = ALPHA_NW*(MC_NW/RK_NW)*T_NW;

% Optimality condition wrt aggregated labor (I.e. demand function) (L_NW)
L_NW = (1 - ALPHA_NW)*(MC_NW/REAL_W_NW)*T_NW;

% Pricing (REAL_PQ_NW)
Q_NW - THETAH_NW*Q_NW + MC_NW*THETAH_NW*Q_NW/REAL_PQ_NW
     - 100*PHI_PQ_NW*(DPQ_PQ_NW/DPQ_PQ_NW(-1) - 1)*DPQ_PQ_NW/DPQ_PQ_NW(-1)*Q_NW
```

```
     + DSA_NW*100*PHI_PQ_NW*(DPQ_PQ_NW(+1)/DPQ_PQ_NW - 1)
     *DPQ_PQ_NW(+1)^2/DPQ_PQ_NW*Q_NW(+1)
     *DUT_NW(+1)^(ALPHA_NW/(1 - ALPHA_NW))*DZT_NW(+1) = 0;

% 3) Investment producer
% Investment production function (Y_I_NW)
I_NW = Y_I_NW;

% First order condition of investment decision (REAL_PI_NW)
REAL_PI_NW = 1;

% 4) Capital producer
% Capital production function (KNEW_NW)
KNEW_NW = Z_I_NW*(1 - S_NW)*I_NW;

% Investment adjustment cost function (S_NW)
S_NW = (PHI_I1_NW/2)*(I_NW/I_NW(-1)*DUT_NW^(1/(1 - ALPHA_NW))*DZT_NW
     - steady_state(DUT_NW)^(1/(1 - ALPHA_NW))*steady_state(DZT_NW))^2;

% Derivative of the investment adjustment cost function wrt 1st input
% (Multiplied by I_NW!) (S_PRIME1_NW)
S_PRIME1_NW = PHI_I1_NW*(I_NW/I_NW(-1)*DUT_NW^(1/(1 - ALPHA_NW))*DZT_NW
            - steady_state(DUT_NW)^(1/(1 - ALPHA_NW))*steady_state(DZT_NW))
            *I_NW/I_NW(-1)*DUT_NW^(1/(1 - ALPHA_NW))*DZT_NW;

% Derivative of the investment adjustment cost function wrt 2nd input
% (Multiplied by I_NW!) (S_PRIME2_NW)
S_PRIME2_NW = -PHI_I1_NW*(I_NW/I_NW(-1)*DUT_NW^(1/(1 - ALPHA_NW))*DZT_NW
            - (steady_state(DUT_NW))^(1/(1 - ALPHA_NW))*steady_state(DZT_NW))
            *(I_NW/I_NW(-1)*DUT_NW^(1/(1 - ALPHA_NW))*DZT_NW)^2;

% Optimal capital investment (I_NW)
0 = (DSA_NW*REAL_PK_NW(+1)/DUT_NW(+1))*Z_I_NW(+1)*S_PRIME2_NW(+1)
  + REAL_PK_NW*Z_I_NW*(1 - S_NW - S_PRIME1_NW) - REAL_PI_NW;

% 5) Households
% Derivative of the utility function of households wrt consumption
% (UPRIME_NW)
UPRIME_NW   = Z_U_NW*((C_NW
            - C_NW(-1)*(BC_NW/(DUT_NW^(ALPHA_NW/(1 - ALPHA_NW))*DZT_NW)))
            /(1 - BC_NW/(steady_state(DUT_NW)^(ALPHA_NW/(1 - ALPHA_NW))
            *steady_state(DZT_NW))))^(-1) ;

% Derivative of the utility function of households wrt labor (VPRIME_NW)
VPRIME_NW   = ((L_NW - BL_NW*L_NW(-1))/(1-BL_NW))^ZETA_NW ;

% Stochastic discount factor (DSA_NW)
DSA_NW = BETA_NW*RHO_NW(+1)*UPRIME_NW(+1)/(DPQ_P_NW(+1)
       *DUT_NW(+1)^(ALPHA_NW/(1 - ALPHA_NW))*DZT_NW(+1)*RHO_NW*UPRIME_NW);

% FOC wrt B (Consumption euler equation) (RN3M_NW)
DSA_NW*RN3M_NW = 1;
```

```
% FOC wrt W (REAL_W_NW)
REAL_W_NW = PSI_NW*MRS_NW/((PSI_NW-1)*(1-GAMMA_W_NW)
          + 1000*PHI_W_NW*DPQ_W_NW/DPQ_W_NW(-1)*(DPQ_W_NW/DPQ_W_NW(-1) - 1)
          - DSA_NW*DPQ_W_NW(+1)*L_NW(+1)/L_NW*1000*PHI_W_NW*DPQ_W_NW(+1)
          /DPQ_W_NW*(DPQ_W_NW(+1)/DPQ_W_NW - 1));

% Wage adjusment cost (GAMMA_W_NW)
GAMMA_W_NW = 1000*PHI_W_NW/2*(DPQ_W_NW/DPQ_W_NW(-1) - 1)^2 ;

% Marginal rate of substitution of consumption for leisure (MRS_NW)
MRS_NW = VPRIME_NW/UPRIME_NW;

% FOC wrt K (REAL_PK_NW)
REAL_PK_NW = DSA_NW*DPQ_P_NW(+1)*( (REAL_PK_NW(+1)/DUT_NW(+1))*(1 - DELTA_NW)
           + RK_NW(+1)*U_NW(+1)/DUT_NW(+1) - GAMMA_U_NW(+1)/DUT_NW(+1));

% FOC wrt U (RK_NW)
RK_NW = GAMMAPRIME_U_NW;

% Cost of utilize the capital (GAMMA_U_NW)
GAMMA_U_NW = steady_state(RK_NW)/PHI_U_NW*(exp(PHI_U_NW*(U_NW - 1)) - 1) ;

% Marginal cost of utilizing the capital (GAMMAPRIME_U_NW)
GAMMAPRIME_U_NW  = steady_state(RK_NW)*exp(PHI_U_NW*(U_NW - 1)) ;

% Capital accumulation (K_NW)
K_NW = (1-DELTA_NW)*K_NW(-1)/(DUT_NW^(1/(1 - ALPHA_NW))*DZT_NW) + KNEW_NW;

% Capital utilization (U_NW)
KBAR_NW = U_NW*K_NW(-1)/(DUT_NW^(1/(1 - ALPHA_NW))*DZT_NW);

% 6) Market clearing  and  defintions
% Final good market (C_NW)
A_NW = C_NW + Y_I_NW;

% Intermediate good market (T_NW)
T_NW = Q_NW;

% Definition of output
NAT_Y_NW = A_NW;

% Consumption growth (DPQ_C_NW)
DPQ_C_NW = (C_NW/C_NW(-1))*DZT_NW*DUT_NW^(ALPHA_NW/(1 - ALPHA_NW));

% Investment growth (DPQ_I_NW)
DPQ_I_NW = (I_NW/I_NW(-1))*DZT_NW*DUT_NW^(1/(1 - ALPHA_NW));

% Capital growth (DPQ_K_NW)
DPQ_K_NW = (K_NW/K_NW(-1))*DZT_NW*DUT_NW^(1/(1 - ALPHA_NW));

% Output growth (DPQ_A_NW)
```

```
DPQ_Y_NW = (NAT_Y_NW/NAT_Y_NW(-1))*DZT_NW*DUT_NW^(ALPHA_NW/(1 - ALPHA_NW));

% Intermidate good inflation (DPQ_PQ_NW)
DPQ_PQ_NW = DPQ_P_NW*REAL_PQ_NW/REAL_PQ_NW(-1);

% Real investment inflation (DPQ_REAL_PI_NW)
DPQ_REAL_PI_NW = (REAL_PI_NW/REAL_PI_NW(-1))/DUT_NW;

% Real wage inflation (DPQ_REAL_W_NW)
DPQ_REAL_W_NW = (REAL_W_NW/REAL_W_NW(-1))*DZT_NW*DUT_NW^(ALPHA_NW/(1 - ALPHA_NW));

% Wage inflation (DPQ_W_NW)
DPQ_W_NW = REAL_W_NW/REAL_W_NW(-1)*DPQ_P_NW*DZT_NW*DUT_NW^(ALPHA_NW/(1 - ALPHA_NW));

% 7) Monetary policy rule (DPQ_P_NW)
RN3M_NW = RN3M_NW(-1)^OMEGA_R_NW*(steady_state(RN3M_NW)
        *(DPQ_P_NW/steady_state(DPQ_P_NW))^OMEGA_P_NW
        *(NAT_Y_NW/steady_state(NAT_Y_NW))^OMEGA_Y_NW
        *(DPQ_Y_NW/steady_state(DPQ_Y_NW))^OMEGA_DPQ_Y_NW)
        ^(1-OMEGA_R_NW)*(Z_RN3M_NW/steady_state(Z_RN3M_NW));
[static] DPQ_P_NW = DPQ_P_NW_SS;

% 8) Shock processes
% Permanent labor-augmenting technology process (DZT_NW)
log(DZT_NW) = (1-LAMBDA_DZT_NW)*log(DZT_NW_SS)
            + LAMBDA_DZT_NW*log(DZT_NW(-1)) + E_DZT_NW*std_E_DZT_NW;

% Investment-specific technological progress (DUT_NW)
log(DUT_NW) = (1-LAMBDA_DUT_NW)*log(DUT_NW_SS)
            + LAMBDA_DUT_NW*log(DUT_NW(-1)) + E_DUT_NW*std_E_DUT_NW;

% Competition in the labor market shock process (PSI_NW)
log(PSI_NW) = (1-LAMBDA_PSI_NW)*log(PSI_NW_SS)
            + LAMBDA_PSI_NW*log(PSI_NW(-1)) + E_PSI_NW*std_E_PSI_NW ;

% Discount factor shock process (RHO_NW)
log(RHO_NW) = (1-LAMBDA_RHO_NW)*log(RHO_NW_SS)
            + LAMBDA_RHO_NW*log(RHO_NW(-1)) + E_RHO_NW*std_E_RHO_NW ;

% Price markup shock process (THETAH_NW)
log(THETAH_NW) = (1-LAMBDA_THETAH_NW)*log(THETAH_NW_SS)
               + LAMBDA_THETAH_NW*log(THETAH_NW(-1))
               + E_THETAH_NW*std_E_THETAH_NW ;

% Marginal efficiency of investment shock process (Z_I_NW)
log(Z_I_NW) = (1-LAMBDA_I_NW)*log(Z_I_NW_SS)
            + LAMBDA_I_NW*log(Z_I_NW(-1)) + E_I_NW*std_E_I_NW ;

% Temporary labor augmenting technology shock process (Z_L_NW)
log(Z_L_NW) = (1-LAMBDA_L_NW)*log(Z_L_NW_SS)
            + LAMBDA_L_NW*log(Z_L_NW(-1)) + E_L_NW*std_E_L_NW ;
```

```
% Consumption preference shock process (Z_U_NW)
log(Z_U_NW) = (1-LAMBDA_U_NW)*log(Z_U_NW_SS)
            + LAMBDA_U_NW*log(Z_U_NW(-1)) + E_U_NW*std_E_U_NW ;

% Monetary policy shock process (Z_RN3M_NW)
log(Z_RN3M_NW) = (1-LAMBDA_RN3M_NW)*log(Z_RN3M_NW_SS)
               + LAMBDA_RN3M_NW*log(Z_RN3M_NW(-1)) + E_RN3M_NW*std_E_RN3M_NW;

% Construct reported variables that can be asked for in IRFs, but
% are not part of the model. Here you can use all MATLAB functions
% that act on a double vector, and returns an output with the same
% size as the input.
reporting

    C_NW_LEVEL = exp(cumsum(log(DPQ_C_NW) - log(steady_state(DPQ_C_NW))));
    I_NW_LEVEL = exp(cumsum(log(DPQ_I_NW) - log(steady_state(DPQ_I_NW))));
    K_NW_LEVEL = exp(cumsum(log(DPQ_K_NW) - log(steady_state(DPQ_K_NW))));
    Y_NW_LEVEL = exp(cumsum(log(DPQ_Y_NW) - log(steady_state(DPQ_Y_NW))));
```

To parse and solve this model you can use the following instructions:

```
% Read the stationary model
modelS = nb_dsge('nb_file','jpt.nb');

% Give it a name
modelS = set(modelS,'name','Stationarize JPT manually');

% Set the parameters
param                   = struct();
param.ALPHA_NW          = 0.167;
param.BC_NW             = 0.859;
param.BETA_NW           = 100/(0.134+100);
param.BL_NW             = 0;
param.DELTA_NW          = 0.025;
param.DPQ_P_NW_SS       = (0.702 + 100)/100;
param.DUT_NW_SS         = 1 + (0.597/100);
param.DZT_NW_SS         = 1 + (0.303 - (param.ALPHA_NW/...
                          (1 - param.ALPHA_NW))*0.597)/100;
param.LAMBDA_DUT_NW     = 0.156;
param.LAMBDA_DZT_NW     = 0.286;
param.LAMBDA_I_NW       = 0.772;
param.LAMBDA_L_NW       = 0;
param.LAMBDA_PSI_NW     = 0.967;
param.LAMBDA_RHO_NW     = 0.590;
param.LAMBDA_RN3M_NW    = 0;
param.LAMBDA_U_NW       = 0;
param.LAMBDA_THETAH_NW  = 0.971;
param.OMEGA_DPQ_Y_NW    = 0.208;
param.OMEGA_P_NW        = 1.709;
param.OMEGA_R_NW        = 0.858;
param.OMEGA_Y_NW        = 0.051;
param.PHI_PQ_NW         = 0.2;
param.PHI_I1_NW         = 2.657;
```

```
param.PHI_W_NW          = 1.0080;
param.PHI_U_NW          = 5.434;
param.PSI_NW_SS         = 1.135/(1.135 - 1);
param.RHO_NW_SS         = 1;
param.THETAH_NW_SS      = 1.171/(1.171 - 1);
param.ZETA_NW           = 4.444;
param.Z_I_NW_SS         = 1;
param.Z_L_NW_SS         = 1;
param.Z_U_NW_SS         = 1;
param.Z_RN3M_NW_SS      = 1;
param.std_E_DUT_NW      = 0.630;
param.std_E_DZT_NW      = 0.933;
param.std_E_I_NW        = 5.103;
param.std_E_L_NW        = 0;
param.std_E_PSI_NW      = 0.310;
param.std_E_RHO_NW      = 0.036;
param.std_E_RN3M_NW     = 0.210;
param.std_E_THETAH_NW   = 0.219;
param.std_E_U_NW        = 0;
modelNS                 = assignParameters(modelNS,param);

% Solve steady state numerically
ssInit = struct(...
    'GAMMA_W_NW',0,...
    'PSI_NW',param.PSI_NW_SS,...
    'S_NW',0,...
    'S_PRIME1_NW',0,...
    'S_PRIME2_NW',0,...
    'GAMMA_U_NW',0,...
    'THETAH_NW',param.THETAH_NW_SS);

modelS = checkSteadyState(modelS,...
    'solver',               'fsolve',...
    'steady_state_default', @ones,...
    'steady_state_init',    ssInit,...
    'steady_state_solve',   true);
ss = getSteadyState(modelS)

% Solve stationary model
modelS = solve(modelS);
```

E The non-stationary model file

```
endogenous
A_NW % Final goods production
C_NW % Consumption
DPQ_C_NW % Consumption growth
DPQ_I_NW % Investment growth
DPQ_K_NW % Capital growth
DPQ_P_NW % Inflation
DPQ_PQ_NW % Intermidate goods inflation
DPQ_REAL_PI_NW % Real investment inflation
DPQ_REAL_W_NW % Real wage inflation
DPQ_W_NW % Wage inflation
DPQ_Y_NW % Output growth
DSA_NW % Stochastic discount factor
GAMMAPRIME_U_NW % Marginal cost of utilizing the capital
GAMMA_U_NW % Cost of utilize the capital
GAMMA_W_NW % Wage adjustment cost
I_NW % Investment
K_NW % Capital
KBAR_NW % Utilized capital
KNEW_NW % Capital goods produced each period
L_NW % Hours worked
MC_NW % Marginal cost
MRS_NW % Marginal rate of substitution
NAT_Y_NW % Output
PSI_NW % Competition in the labor market shock process
Q_NW % Demand for intermediate goods
REAL_PI_NW % Real price of investment
REAL_PK_NW % Real price of capital
REAL_PQ_NW % Real intermediate goods price
REAL_W_NW % Real wage rate
RHO_NW % Discount factor shock process
RK_NW % Rental rate of capital
RN3M_NW % Money market interest rate
S_NW % Investment adjustment cost function
S_PRIME1_NW % Derivative of the investment
% adjustment cost function with respect to first input times I_NW
S_PRIME2_NW % Derivative of the investment
% adjustment cost function with respect to second input times I_NW
T_NW % Intermediate goods production
THETAH_NW % Price markup shock process
U_NW % Utilization rate
UPRIME_NW % Derivative of the utility function
% of households wrt consumption
VPRIME_NW % Derivative of the utility function
% of households wrt labor
Y_I_NW % Input to investment production
Z_I_NW % Marginal efficiency of investment shock process
Z_L_NW % Temporary labor augmenting technology shock process
Z_RN3M_NW % Monetary policy shock process
```

```
Z_U_NW % Consumption preference shock process

exogenous
E_DUT_NW  % Investment-specific technological innovation
E_DZT_NW % Permanent labor-augmenting technology innovation
E_I_NW % Marginal efficiency of investment innovation
E_L_NW % Temporary labor augmenting technology innovation
E_PSI_NW % Competition in the labor market innovation
E_RHO_NW % Discount factor innovation
E_RN3M_NW % Monetary policy innovation
E_THETAH_NW % Price markup innovation
E_U_NW % Consumption preference innovation

parameters
ALPHA_NW % Capital share
BC_NW % Habit in consumption
BETA_NW % Discount factor
BL_NW % Habit in hours worked
DELTA_NW % Depreciation rate
DPQ_P_NW_SS % Steady-state inflation
DUT_NW_SS % Steady-state growth rate in investment-
% specific technology
DZT_NW_SS % Steady-state growth rate in labor-
% augmenting technology
LAMBDA_DUT_NW % Shock persistent parameter
% for the investment-specific technology shock
LAMBDA_DZT_NW % Shock persistent parameter
% for the labor-augmenting technology shock
LAMBDA_I_NW % Shock persistent parameter
% for the marginal efficiency of investment shock
LAMBDA_L_NW % Shock persistent parameter
% for the temporary labor augmenting technology shock
LAMBDA_PSI_NW % Shock persistent parameter
% for the competition in the labor market shock
LAMBDA_RHO_NW % Shock persistent parameter
% for the discount factor shock
LAMBDA_RN3M_NW % Shock persistent parameter
% for the monetary policy shock
LAMBDA_U_NW % Shock persistent parameter
% for the price markup shock
LAMBDA_THETAH_NW % Shock persistent parameter
% for the consumption preference shock
OMEGA_Y_NW % Taylor rule coefficient on ouput gap.
OMEGA_DPQ_Y_NW  % Taylor rule coefficient on ouput growth gap.
OMEGA_P_NW % Taylor rule coefficient on inflation gap.
OMEGA_R_NW % Interest rate smoothing in the taylor rule
PHI_PQ_NW % Intermediate goods price adjustment cost parameter
PHI_I1_NW % Investment adjustment cost parameter
PHI_W_NW % Wage adjustment cost parameter
PHI_U_NW % Capital utilization cost parameter
PSI_NW_SS % Steady-state elasticity of substitution
% between differentiated labor
```

```
RHO_NW_SS % Discount factor shock process in steady state
THETAH_NW_SS % Steady-state elasticity of
% substitution between intermidate goods
ZETA_NW % Inverse Frisch elasticity
Z_I_NW_SS % Marginal efficiency of investment
% shock in steady-state
Z_L_NW_SS % Temporary labor augmenting technology
% shock in steady-state
Z_RN3M_NW_SS % Monetary policy shock in
% steady-state
Z_U_NW_SS % Consumption preference shock in
% steady-state
std_E_DUT_NW % Standard deviation of the
% innovation to the Z_DUT_NW shock process
std_E_DZT_NW % Standard deviation of the
% innovation to the Z_DZT_NW shock process
std_E_I_NW % Standard deviation of the
% innovation to the Z_I_NW shock process
std_E_L_NW % Standard deviation of the
% innovation to the Z_L_NW shock process
std_E_PSI_NW % Standard deviation of the
% innovation to the PSI_NW shock process
std_E_RHO_NW % Standard deviation of the
% innovation to the RHO_NW shock process
std_E_RN3M_NW % Standard deviation of the
% innovation to the Z_RN3M_NW shock process
std_E_THETAH_NW % Standard deviation of the
% innovation to the THETAH_NW shock process
std_E_U_NW % Standard deviation of the
% innovation to the Z_U_NW shock process

unitrootvars
Z
UPSILON


model
% 1) Final goods sector
% Production function (A_NW)
A_NW = Q_NW;

% FOC (Q_NW)
REAL_PQ_NW = 1;

% 2) Intermediate goods sector
% Intermidate production function (KBAR_NW)
T_NW = (Z*Z_L_NW*L_NW)^(1-ALPHA_NW)*KBAR_NW^ALPHA_NW;

% Optimality condition wrt utilized capital (I.e. demand function) (MC_NW)
KBAR_NW = ALPHA_NW*(MC_NW/RK_NW)*T_NW;

% Optimality condition wrt aggregated labor (I.e. demand function) (L_NW)
```

```
L_NW = (1 - ALPHA_NW)*(MC_NW/REAL_W_NW)*T_NW;

% Pricing (REAL_PQ_NW)
Q_NW - THETAH_NW*Q_NW + MC_NW*THETAH_NW*Q_NW/REAL_PQ_NW
     - 100*PHI_PQ_NW*(DPQ_PQ_NW/DPQ_PQ_NW(-1) - 1)*DPQ_PQ_NW/DPQ_PQ_NW(-1)*Q_NW
     + DSA_NW*100*PHI_PQ_NW*(DPQ_PQ_NW(+1)/DPQ_PQ_NW - 1)
     *DPQ_PQ_NW(+1)^2/DPQ_PQ_NW*Q_NW(+1) = 0;

% 3) Investment producer
% Investment production function (Y_I_NW)
I_NW = UPSILON*Y_I_NW;

% First order condition for investment  (REAL_PI_NW)
UPSILON*REAL_PI_NW = 1;

% 4) Capital producer
% Capital production function (KNEW_NW)
KNEW_NW = Z_I_NW*(1 - S_NW)*I_NW;

% Investment adjustment cost function (S_NW)
S_NW = (PHI_I1_NW/2)*(I_NW/I_NW(-1) - bgp(I_NW))^2;

% Derivative of the investment adjustment cost function with respect to first input
% (Multiplied by I_NW!) (S_PRIME1_NW)
S_PRIME1_NW = PHI_I1_NW*(I_NW/I_NW(-1) - bgp(I_NW))*I_NW/I_NW(-1);

% Derivative of the investment adjustment cost function with respect to second input
% (Multiplied by I_NW!) (S_PRIME2_NW)
S_PRIME2_NW = -PHI_I1_NW*(I_NW/I_NW(-1) - bgp(I_NW))*(I_NW/I_NW(-1))^2;

% Optimal capital investment (I_NW)
0 = DSA_NW*REAL_PK_NW(+1)*Z_I_NW(+1)*S_PRIME2_NW(+1)
  + REAL_PK_NW*Z_I_NW*(1 - S_NW - S_PRIME1_NW) - REAL_PI_NW;

% 5) Households
 % Marignal  utility of consumption  (UPRIME_NW)
UPRIME_NW   = Z_U_NW*((C_NW - C_NW(-1)*BC_NW)/(1 - BC_NW/bgp(C_NW)))^(-1) ;

% Marginal  utility  of  labor labor (VPRIME_NW)
VPRIME_NW   = ((L_NW - BL_NW*L_NW(-1))/(1-BL_NW))^ZETA_NW ;

% Stochastic discount factor (DSA_NW)
DSA_NW = BETA_NW*RHO_NW(+1)*UPRIME_NW(+1)/(DPQ_P_NW(+1)*RHO_NW*UPRIME_NW);

% Consumption euler equation (RN3M_NW)
DSA_NW*RN3M_NW = 1;

% Optimal wage  setting  (REAL_W_NW)
REAL_W_NW = PSI_NW*MRS_NW/((PSI_NW-1)*(1-GAMMA_W_NW)
          + 1000*PHI_W_NW*DPQ_W_NW/DPQ_W_NW(-1)*(DPQ_W_NW/DPQ_W_NW(-1) - 1)
          - DSA_NW*DPQ_W_NW(+1)*L_NW(+1)/L_NW*1000*PHI_W_NW*DPQ_W_NW(+1)
          /DPQ_W_NW*(DPQ_W_NW(+1)/DPQ_W_NW - 1));
```

```
% Wage adjusment cost (GAMMA_W_NW)
GAMMA_W_NW = 1000*PHI_W_NW/2*(DPQ_W_NW/DPQ_W_NW(-1) - 1)^2 ;

% Marginal rate of substitution between consumption and leisure (MRS_NW)
MRS_NW = VPRIME_NW/UPRIME_NW;

%  6) Entrepreneurs
 % optimality with respect  to  capital  (REAL_PK_NW)
REAL_PK_NW = DSA_NW*DPQ_P_NW(+1)*( REAL_PK_NW(+1)*(1 - DELTA_NW) +
RK_NW(+1)*U_NW(+1) - GAMMA_U_NW(+1));

% optimality with respect to  utilization (RK_NW)
RK_NW = GAMMAPRIME_U_NW;

% Cost of utilizing the capital (GAMMA_U_NW)
GAMMA_U_NW = steady_state(RK_NW)/PHI_U_NW*(exp(PHI_U_NW*(U_NW - 1)) - 1) ;

% Marginal cost of utilizing capital (GAMMAPRIME_U_NW)
GAMMAPRIME_U_NW  = steady_state(RK_NW)*exp(PHI_U_NW*(U_NW - 1)) ;

% Capital accumulation (K_NW)
K_NW = (1-DELTA_NW)*K_NW(-1) + KNEW_NW;

% Capital utilization (U_NW)
KBAR_NW = U_NW*K_NW(-1);

% 7) Market clearing
 % Final good market clearing (C_NW)
A_NW = C_NW + Y_I_NW;

% Intermediate good market clearing  (T_NW)
T_NW = Q_NW;

%  8) Definitions
% Definition of natural output
NAT_Y_NW = A_NW;

% Consumption growth (DPQ_C_NW)
DPQ_C_NW = C_NW/C_NW(-1);

% Investment growth (DPQ_I_NW)
DPQ_I_NW = I_NW/I_NW(-1);

% Capital growth (DPQ_K_NW)
DPQ_K_NW = K_NW/K_NW(-1);

% Definition of natural output growth (DPQ_A_NW)
DPQ_Y_NW = NAT_Y_NW/NAT_Y_NW(-1);

% Definition of intermidate good inflation (DPQ_PQ_NW)
DPQ_PQ_NW = DPQ_P_NW*REAL_PQ_NW/REAL_PQ_NW(-1);
```

```
% Real investment inflation (DPQ_REAL_PI_NW)
DPQ_REAL_PI_NW = REAL_PI_NW/REAL_PI_NW(-1);

% Real wage inflation (DPQ_REAL_W_NW)
DPQ_REAL_W_NW = REAL_W_NW/REAL_W_NW(-1);

% Definition  of wage inflation (DPQ_W_NW)
DPQ_W_NW = REAL_W_NW/REAL_W_NW(-1)*DPQ_P_NW;

% 9)  Taylor rule (DPQ_P_NW)
RN3M_NW = RN3M_NW(-1)^OMEGA_R_NW*(steady_state(RN3M_NW)
        *(DPQ_P_NW/steady_state(DPQ_P_NW))^OMEGA_P_NW
        *(NAT_Y_NW/steady_state(NAT_Y_NW))^OMEGA_Y_NW
        *(DPQ_Y_NW/steady_state(DPQ_Y_NW))^OMEGA_DPQ_Y_NW)
        ^(1-OMEGA_R_NW)*(Z_RN3M_NW/steady_state(Z_RN3M_NW));
[static] DPQ_P_NW = DPQ_P_NW_SS;

% 10) Shock processes
% Labor market competition shock  (PSI_NW)
log(PSI_NW) = (1-LAMBDA_PSI_NW)*log(PSI_NW_SS) +
LAMBDA_PSI_NW*log(PSI_NW(-1)) + E_PSI_NW*std_E_PSI_NW ;

% Discount factor shock  (RHO_NW)
log(RHO_NW) = (1-LAMBDA_RHO_NW)*log(RHO_NW_SS) +
LAMBDA_RHO_NW*log(RHO_NW(-1)) + E_RHO_NW*std_E_RHO_NW ;

% Price markup shock  (THETAH_NW)
log(THETAH_NW) = (1-LAMBDA_THETAH_NW)*log(THETAH_NW_SS)
               + LAMBDA_THETAH_NW*log(THETAH_NW(-1))
               + E_THETAH_NW*std_E_THETAH_NW ;

% Marginal efficiency of investment shock  (Z_I_NW)
log(Z_I_NW) = (1-LAMBDA_I_NW)*log(Z_I_NW_SS)
            + LAMBDA_I_NW*log(Z_I_NW(-1)) + E_I_NW*std_E_I_NW ;

% Temporary labor augmenting technology shock  (Z_L_NW)
log(Z_L_NW) = (1-LAMBDA_L_NW)*log(Z_L_NW_SS)
            + LAMBDA_L_NW*log(Z_L_NW(-1)) + E_L_NW*std_E_L_NW ;

% Consumption preference shock  (Z_U_NW)
log(Z_U_NW) = (1-LAMBDA_U_NW)*log(Z_U_NW_SS)
            + LAMBDA_U_NW*log(Z_U_NW(-1)) + E_U_NW*std_E_U_NW ;

% Monetary policy shock  (Z_RN3M_NW)
log(Z_RN3M_NW)     = (1-LAMBDA_RN3M_NW)*log(Z_RN3M_NW_SS) +
LAMBDA_RN3M_NW*log(Z_RN3M_NW(-1)) + E_RN3M_NW*std_E_RN3M_NW;

% Unit root processes
Z/Z(-1)             = DZT_NW_SS^(1-LAMBDA_DZT_NW)*
(Z(-1)/Z(-2))^LAMBDA_DZT_NW*exp(std_E_DZT_NW*E_DZT_NW);
UPSILON/UPSILON(-1) = DUT_NW_SS^(1-LAMBDA_DUT_NW)*
```

```
(UPSILON(-1)/UPSILON(-2))^LAMBDA_DUT_NW*exp(std_E_DUT_NW*E_DUT_NW);

% Construct reported variables that can be asked for in IRFs, but
% are not part of the model. Here you can use all MATLAB functions
% that act on a double vector, and returns an output with the same
% size as the input.
reporting

    C_NW_LEVEL = exp(cumsum(log(DPQ_C_NW) - log(steady_state(DPQ_C_NW))));
    I_NW_LEVEL = exp(cumsum(log(DPQ_I_NW) - log(steady_state(DPQ_I_NW))));
    K_NW_LEVEL = exp(cumsum(log(DPQ_K_NW) - log(steady_state(DPQ_K_NW))));
    Y_NW_LEVEL = exp(cumsum(log(DPQ_Y_NW) - log(steady_state(DPQ_Y_NW))));
```

The set of NB toolbox commands to transform and solve the model:

```
% Read the non-stationary model
modelNS = nb_dsge('nb_file','jpt_non_stationary.nb');

% Give it a name
modelNS = set(modelNS,'name','Stationarize JPT automatically');

% Set the parameters
param                    = struct();
param.ALPHA_NW           = 0.167;
param.BC_NW              = 0.859;
param.BETA_NW            = 100/(0.134+100);
param.BL_NW              = 0;
param.DELTA_NW           = 0.025;
param.DPQ_P_NW_SS        = (0.702 + 100)/100;
param.DUT_NW_SS          = 1 + (0.597/100);
param.DZT_NW_SS          = 1 + (0.303 - (param.ALPHA_NW/...
                           (1 - param.ALPHA_NW))*0.597)/100;
param.LAMBDA_DUT_NW      = 0.156;
param.LAMBDA_DZT_NW      = 0.286;
param.LAMBDA_I_NW        = 0.772;
param.LAMBDA_L_NW        = 0;
param.LAMBDA_PSI_NW      = 0.967;
param.LAMBDA_RHO_NW      = 0.590;
param.LAMBDA_RN3M_NW     = 0;
param.LAMBDA_U_NW        = 0;
param.LAMBDA_THETAH_NW   = 0.971;
param.OMEGA_DPQ_Y_NW     = 0.208;
param.OMEGA_P_NW         = 1.709;
param.OMEGA_R_NW         = 0.858;
param.OMEGA_Y_NW         = 0.051;
param.PHI_PQ_NW          = 0.2;
param.PHI_I1_NW          = 2.657;
param.PHI_W_NW           = 1.0080;
param.PHI_U_NW           = 5.434;
param.PSI_NW_SS          = 1.135/(1.135 - 1);
param.RHO_NW_SS          = 1;
param.THETAH_NW_SS       = 1.171/(1.171 - 1);
param.ZETA_NW            = 4.444;
param.Z_I_NW_SS          = 1;
```

```
param.Z_L_NW_SS         = 1;
param.Z_U_NW_SS         = 1;
param.Z_RN3M_NW_SS      = 1;
param.std_E_DUT_NW      = 0.630;
param.std_E_DZT_NW      = 0.933;
param.std_E_I_NW        = 5.103;
param.std_E_L_NW        = 0;
param.std_E_PSI_NW      = 0.310;
param.std_E_RHO_NW      = 0.036;
param.std_E_RN3M_NW     = 0.210;
param.std_E_THETAH_NW   = 0.219;
param.std_E_U_NW        = 0;
modelNS                 = assignParameters(modelNS,param);

% Solve for the balanced growth path (steps 1-3 of the algorithm)
modelNS  = solveBalancedGrowthPath(modelNS);

% Stationarize the model (step 4 of the algorithm)
modelNS = stationarize(modelNS)


% Solve stationary steady state (numerically)
modelNS = checkSteadyState(modelNS,...
    'solver',               'fsolve',...
    'steady_state_solve',   true,...
    'steady_state_default', @ones);

% Return the steady state solution as a cell matrix
ss = getSteadyState(modelNS)

% Return the solution to the balanced growth path as a cell matrix
bgp = getBalancedGrowthPath(modelNS)

% Obtain the state space representation of  the  solution
% y(t) = A y(t-1) + B eps(t)
modelNS = solve(modelNS);


% Compute and plot  impulse responses
% The modelS object represents the model that we have stationarized manually.
[~,~,plotter] = irf([modelNS,modelS],'periods',  60,'plotSS',   true,'shocks',   ...
    {'E_DUT_NW'},...
   'variables',{'C_NW','DPQ_P_NW','DPQ_W_NW','I_NW','K_NW',...
                'L_NW','NAT_Y_NW','RN3M_NW'},...
  'settings',{'legBox','off','legFontSize',18,'subPlotSize',[3,3],...
                'figureTitle',false,'lookUpMatrix','lookUpMatrixJPT',...
               'legends',{'Stationarize JPT automatically',...
                          'Stationarize JPT manually',...
                          '',...
                          'Steady state'}});
nb_graphInfoStructGUI(plotter)
```

F The automatically made stationary file

endogenous

```
Z_U_NW Z_RN3M_NW Z_L_NW Z_I_NW Y_I_NW VPRIME_NW U_NW UPRIME_NW T_NW
THETAH_NW S_PRIME2_NW S_PRIME1_NW S_NW RN3M_NW RK_NW RHO_NW REAL_W_NW
REAL_PQ_NW REAL_PK_NW REAL_PI_NW Q_NW PSI_NW NAT_Y_NW MRS_NW MC_NW
L_NW K_NW KNEW_NW KBAR_NW I_NW GAMMA_W_NW GAMMA_U_NW GAMMAPRIME_U_NW
D_Z_Z D_Z_Y_I_NW D_Z_UPSILON D_Z_UPRIME_NW D_Z_T_NW D_Z_RK_NW D_Z_REAL_W_NW
D_Z_REAL_PK_NW D_Z_REAL_PI_NW D_Z_Q_NW D_Z_NAT_Y_NW D_Z_MRS_NW D_Z_K_NW
D_Z_KNEW_NW D_Z_KBAR_NW D_Z_I_NW D_Z_GAMMA_U_NW D_Z_GAMMAPRIME_U_NW
D_Z_C_NW D_Z_A_NW DSA_NW DPQ_Y_NW DPQ_W_NW DPQ_REAL_W_NW DPQ_REAL_PI_NW
DPQ_P_NW DPQ_PQ_NW DPQ_K_NW DPQ_I_NW DPQ_C_NW C_NW A_NW

exogenous
E_DUT_NW E_DZT_NW E_I_NW E_L_NW E_PSI_NW E_RHO_NW E_RN3M_NW E_THETAH_NW
E_U_NW

parameters
std_E_U_NW std_E_THETAH_NW std_E_RN3M_NW std_E_RHO_NW std_E_PSI_NW
std_E_L_NW std_E_I_NW std_E_DZT_NW std_E_DUT_NW Z_U_NW_SS Z_RN3M_NW_SS
Z_L_NW_SS Z_I_NW_SS ZETA_NW THETAH_NW_SS RHO_NW_SS PSI_NW_SS PHI_W_NW
PHI_U_NW PHI_PQ_NW PHI_I1_NW OMEGA_Y_NW OMEGA_R_NW OMEGA_P_NW OMEGA_DPQ_Y_NW
LAMBDA_U_NW LAMBDA_THETAH_NW LAMBDA_RN3M_NW LAMBDA_RHO_NW LAMBDA_PSI_NW
LAMBDA_L_NW LAMBDA_I_NW LAMBDA_DZT_NW LAMBDA_DUT_NW DZT_NW_SS DUT_NW_SS
DPQ_P_NW_SS DELTA_NW BL_NW BETA_NW BC_NW ALPHA_NW

model
A_NW-Q_NW;
REAL_PQ_NW-1;
T_NW = (((1*Z_L_NW)*L_NW)^(1-ALPHA_NW))*(KBAR_NW^ALPHA_NW);
KBAR_NW = (ALPHA_NW*(MC_NW/RK_NW))*T_NW;
L_NW = ((1-ALPHA_NW)*(MC_NW/REAL_W_NW))*T_NW;
Q_NW-(THETAH_NW*Q_NW)+((MC_NW*THETAH_NW)*Q_NW)/REAL_PQ_NW-
(((((100*PHI_PQ_NW)*(DPQ_PQ_NW/DPQ_PQ_NW(-1)-1))*DPQ_PQ_NW)/DPQ_PQ_NW(-1))*Q_NW)+
(((((DSA_NW*100)*PHI_PQ_NW)*(DPQ_PQ_NW(+1)/DPQ_PQ_NW-1))
*(DPQ_PQ_NW(+1)^2))/DPQ_PQ_NW)*(Q_NW(+1)*D_Z_Q_NW(+1))-0;
I_NW = 1*Y_I_NW;
1*REAL_PI_NW-1;
KNEW_NW = (Z_I_NW*(1-S_NW))*I_NW;
S_NW = (PHI_I1_NW/2)*((I_NW/(I_NW(-1)*D_Z_I_NW^-1)-
steady_state(D_Z_I_NW))^2);
S_PRIME1_NW = ((PHI_I1_NW*(I_NW/(I_NW(-1)*D_Z_I_NW^-1)-
steady_state(D_Z_I_NW)))*I_NW)/(I_NW(-1)*D_Z_I_NW^-1);
S_PRIME2_NW = ((-PHI_I1_NW)*(I_NW/(I_NW(-1)*D_Z_I_NW^-1)-
steady_state(D_Z_I_NW)))*((I_NW/(I_NW(-1)*D_Z_I_NW^-1))^2);
0 = ((DSA_NW*(REAL_PK_NW(+1)*D_Z_REAL_PK_NW(+1)))*Z_I_NW(+1))*
S_PRIME2_NW(+1)+(REAL_PK_NW*Z_I_NW)*(1-S_NW-S_PRIME1_NW)-REAL_PI_NW;
UPRIME_NW = Z_U_NW*(((C_NW-((C_NW(-1)*D_Z_C_NW^-1)*BC_NW))/
(1-(BC_NW/steady_state(D_Z_C_NW))))^(-1));
VPRIME_NW = ((L_NW-(BL_NW*L_NW(-1)))/(1-BL_NW))^ZETA_NW;
DSA_NW = ((BETA_NW*RHO_NW(+1))*(UPRIME_NW(+1)*D_Z_UPRIME_NW(+1)))/
((DPQ_P_NW(+1)*RHO_NW)*UPRIME_NW);
DSA_NW*RN3M_NW-1;
REAL_W_NW = (PSI_NW*MRS_NW)/((PSI_NW-1)*(1-GAMMA_W_NW)+
(((1000*PHI_W_NW)*DPQ_W_NW)/DPQ_W_NW(-1))*(DPQ_W_NW/DPQ_W_NW(-1)-1)-
```

```
(((((((((DSA_NW*DPQ_W_NW(+1))*L_NW(+1))/L_NW)*1000)*PHI_W_NW)*DPQ_W_NW(+1))/
DPQ_W_NW)*(DPQ_W_NW(+1)/DPQ_W_NW-1)));
GAMMA_W_NW = ((1000*PHI_W_NW)/2)*((DPQ_W_NW/DPQ_W_NW(-1)-1)^2);
MRS_NW = VPRIME_NW/UPRIME_NW;
REAL_PK_NW = (DSA_NW*DPQ_P_NW(+1))*((REAL_PK_NW(+1)*D_Z_REAL_PK_NW(+1))*(1-DELTA_NW)+
(RK_NW(+1)*D_Z_RK_NW(+1))*U_NW(+1)-(GAMMA_U_NW(+1)*D_Z_GAMMA_U_NW(+1)));
RK_NW-GAMMAPRIME_U_NW;
GAMMA_U_NW = (steady_state(RK_NW)/PHI_U_NW)*(exp(PHI_U_NW*(U_NW-1))-1);
GAMMAPRIME_U_NW = steady_state(RK_NW)*(exp(PHI_U_NW*(U_NW-1)));
K_NW = (1-DELTA_NW)*(K_NW(-1)*D_Z_K_NW^-1)+KNEW_NW;
KBAR_NW = U_NW*(K_NW(-1)*D_Z_K_NW^-1);
A_NW = C_NW+Y_I_NW;
T_NW-Q_NW;
NAT_Y_NW-A_NW;
DPQ_C_NW = C_NW/(C_NW(-1)*D_Z_C_NW^-1);
DPQ_I_NW = I_NW/(I_NW(-1)*D_Z_I_NW^-1);
DPQ_Y_NW = NAT_Y_NW/(NAT_Y_NW(-1)*D_Z_NAT_Y_NW^-1);
DPQ_PQ_NW = (DPQ_P_NW*REAL_PQ_NW)/REAL_PQ_NW(-1);
DPQ_REAL_PI_NW = REAL_PI_NW/(REAL_PI_NW(-1)*D_Z_REAL_PI_NW^-1);
DPQ_REAL_W_NW = REAL_W_NW/(REAL_W_NW(-1)*D_Z_REAL_W_NW^-1);
DPQ_W_NW = (REAL_W_NW/(REAL_W_NW(-1)*D_Z_REAL_W_NW^-1))*DPQ_P_NW;
RN3M_NW = ((RN3M_NW(-1)^OMEGA_R_NW)*((((steady_state(RN3M_NW)*
((DPQ_P_NW/steady_state(DPQ_P_NW))^OMEGA_P_NW))*
((NAT_Y_NW/steady_state(NAT_Y_NW))^OMEGA_Y_NW))*
((DPQ_Y_NW/steady_state(DPQ_Y_NW))^OMEGA_DPQ_Y_NW))^(1-OMEGA_R_NW)))*
(Z_RN3M_NW/steady_state(Z_RN3M_NW));
[static] DPQ_P_NW = DPQ_P_NW_SS;
log(PSI_NW) = (1-LAMBDA_PSI_NW)*log(PSI_NW_SS)+LAMBDA_PSI_NW*log(PSI_NW(-1))+
E_PSI_NW*std_E_PSI_NW;
log(RHO_NW) = (1-LAMBDA_RHO_NW)*log(RHO_NW_SS)+LAMBDA_RHO_NW*log(RHO_NW(-1))+
E_RHO_NW*std_E_RHO_NW;
log(THETAH_NW) = (1-LAMBDA_THETAH_NW)*log(THETAH_NW_SS)+
LAMBDA_THETAH_NW*log(THETAH_NW(-1))+ E_THETAH_NW*std_E_THETAH_NW;
log(Z_I_NW) = (1-LAMBDA_I_NW)*log(Z_I_NW_SS)+LAMBDA_I_NW*log(Z_I_NW(-1))+
E_I_NW*std_E_I_NW;
log(Z_L_NW) = (1-LAMBDA_L_NW)*log(Z_L_NW_SS)+LAMBDA_L_NW*log(Z_L_NW(-1))+
E_L_NW*std_E_L_NW;
log(Z_U_NW) = (1-LAMBDA_U_NW)*log(Z_U_NW_SS)+LAMBDA_U_NW*log(Z_U_NW(-1))+
E_U_NW*std_E_U_NW;
log(Z_RN3M_NW) = (1-LAMBDA_RN3M_NW)*log(Z_RN3M_NW_SS)+
LAMBDA_RN3M_NW*log(Z_RN3M_NW(-1))+
E_RN3M_NW*std_E_RN3M_NW;
```

The unit root processes are specified as:

```
1/(1*D_Z_Z^-1) = ((DZT_NW_SS^(1-LAMBDA_DZT_NW))*
(((1*D_Z_Z^-1)/(1*D_Z_Z^-1*D_Z_Z(-1)^-1))^LAMBDA_DZT_NW))*
(exp(std_E_DZT_NW*E_DZT_NW));
1/(1*D_Z_UPSILON^-1) = ((DUT_NW_SS^(1-LAMBDA_DUT_NW))*
(((1*D_Z_UPSILON^-1)/(1*D_Z_UPSILON^-1*D_Z_UPSILON(-1)^-1))^LAMBDA_DUT_NW))*
(exp(std_E_DUT_NW*E_DUT_NW));
```

The identified G^i functions are:

```
log(D_Z_A_NW)-log(D_Z_Q_NW);
ALPHA_NW*log(D_Z_Z)+log(D_Z_T_NW)-ALPHA_NW*log(D_Z_KBAR_NW)-log(D_Z_Z);
log(D_Z_KBAR_NW)+log(D_Z_RK_NW)-log(D_Z_T_NW);
log(D_Z_REAL_W_NW)-log(D_Z_T_NW);
log(D_Z_I_NW)-log(D_Z_UPSILON)-log(D_Z_Y_I_NW);
log(D_Z_REAL_PI_NW)+log(D_Z_UPSILON);
log(D_Z_KNEW_NW)-log(D_Z_I_NW);
log(D_Z_REAL_PK_NW)-log(D_Z_REAL_PI_NW);
log(D_Z_C_NW)+log(D_Z_UPRIME_NW);
log(D_Z_REAL_W_NW)-log(D_Z_MRS_NW);
log(D_Z_MRS_NW)+log(D_Z_UPRIME_NW);
log(D_Z_REAL_PK_NW)-log(D_Z_RK_NW);
log(D_Z_REAL_PK_NW)-log(D_Z_GAMMA_U_NW);
log(D_Z_RK_NW)-log(D_Z_GAMMAPRIME_U_NW);
log(D_Z_K_NW)-log(D_Z_KNEW_NW);
log(D_Z_KBAR_NW)-log(D_Z_K_NW);
log(D_Z_A_NW)-log(D_Z_C_NW);
log(D_Z_NAT_Y_NW)-log(D_Z_A_NW);
```

Finally the reporting:

```
reporting
    C_NW_LEVEL = exp(cumsum(log(DPQ_C_NW) - log(steady_state(DPQ_C_NW))));
    I_NW_LEVEL = exp(cumsum(log(DPQ_I_NW) - log(steady_state(DPQ_I_NW))));
    K_NW_LEVEL = exp(cumsum(log(DPQ_K_NW) - log(steady_state(DPQ_K_NW))));
    Y_NW_LEVEL = exp(cumsum(log(DPQ_Y_NW) - log(steady_state(DPQ_Y_NW))));
```

G JPT model with unit root in preferences

The steps are the same as in the main paper. First, we write the equations of the (non-stationary) model to a file and save it with name jpt_non_stationary_pref.nb. The non-stationary model file is in this case

```
endogenous
A_NW % Final goods production
C_NW % Consumption
DPQ_C_NW % Consumption growth
DPQ_I_NW % Investment growth
DPQ_K_NW % Investment growth
DPQ_P_NW % Inflation
DPQ_PQ_NW % Intermidate goods inflation
DPQ_REAL_PI_NW % Real investment inflation
DPQ_REAL_W_NW % Real wage inflation
DPQ_W_NW % Wage inflation
DPQ_Y_NW % Output growth
DSA_NW % Stochastic discount factor
GAMMAPRIME_U_NW % Marginal cost of utilizing the capital
GAMMA_U_NW % Cost of utilize the capital
GAMMA_W_NW % Wage adjustment cost
I_NW % Investment
K_NW % Capital
KBAR_NW % Utilized capital
```

```
KNEW_NW % Capital goods produced each period
L_NW % Hours worked
MC_NW % Marginal cost
MRS_NW % Marginal rate of substitution
NAT_Y_NW % Output
PSI_NW % Competition in the labor market shock process
Q_NW % Demand for intermediate goods
REAL_PI_NW % Real price of investment
REAL_PK_NW % Real price of capital
REAL_PQ_NW % Real intermediate goods price
REAL_W_NW % Real wage rate
RHO_NW % Discount factor shock process
RK_NW % Rental rate of capital
RN3M_NW % Money market interest rate
S_NW % Investment adjustment cost function
S_PRIME1_NW % Derivative of the investment adjustment cost
% function wrt 1st input times I_NW
S_PRIME2_NW % Derivative of the investment adjustment cost
% function wrt 2nd input times I_NW
T_NW % Intermediate goods production
THETAH_NW % Price markup shock process
U_NW % Utilization rate
UPRIME_NW % Derivative of the utility function of households wrt consumption
VPRIME_NW % Derivative of the utility function of households wrt labor
Y_I_NW % Input to investment production
Z_I_NW % Marginal efficiency of investment shock process
Z_L_NW % Temporary labor augmenting technology shock process
Z_RN3M_NW % Monetary policy shock process
Z_U_NW % Consumption preference shock process

exogenous
E_DXT_NW  % Preference innovation
E_DZT_NW % Permanent labor-augmenting technology innovation
E_I_NW % Marginal efficiency of investment innovation
E_L_NW % Temporary labor augmenting technology innovation
E_PSI_NW % Competition in the labor market innovation
E_RHO_NW % Discount factor innovation
E_RN3M_NW % Monetary policy innovation
E_THETAH_NW % Price markup innovation
E_U_NW % Consumption preference innovation

parameters
ALPHA_NW % Capital share
BC_NW % Habit in consumption
BETA_NW % Discount factor
BL_NW % Habit in hours worked
DELTA_NW % Depreciation rate
DPQ_P_NW_SS % Steady-state inflation
DXT_NW_SS % Steady-state growth rate in preference
DZT_NW_SS % Steady-state growth rate in labor-augmenting technology
LAMBDA_DXT_NW % Shock persistent parameter for the preference shock
LAMBDA_DZT_NW % Shock persistent parameter for the labor-augmenting
```

```
% technology shock
LAMBDA_I_NW % Shock persistent parameter for the marginal efficiency
% of investment shock
LAMBDA_L_NW % Shock persistent parameter for the temporary labor
% augmenting technology shock
LAMBDA_PSI_NW % Shock persistent parameter for the competition
% in the labor market shock
LAMBDA_RHO_NW % Shock persistent parameter for the discount
% factor shock
LAMBDA_RN3M_NW % Shock persistent parameter for the monetary
% policy shock
LAMBDA_U_NW % Shock persistent parameter for the price
% markup shock
LAMBDA_THETAH_NW % Shock persistent parameter for the consumption
% preference shock
OMEGA_Y_NW % Taylor rule coefficient on ouput gap.
OMEGA_DPQ_Y_NW  % Taylor rule coefficient on ouput growth gap.
OMEGA_P_NW % Taylor rule coefficient on inflation gap.
OMEGA_R_NW % Interest rate smoothing in the taylor rule
PHI_PQ_NW % Intermediate goods price adjustment cost parameter
PHI_I1_NW % Investment adjustment cost parameter
PHI_W_NW % Wage adjustment cost parameter
PHI_U_NW % Capital utilization cost parameter
PSI_NW_SS % Steady-state elasticity of substitution between
% differentiated labor
RHO_NW_SS % Discount factor shock process in steady state
THETAH_NW_SS % Steady-state elasticity of substitution between
% intermidate goods
ZETA_NW % Inverse Frisch elasticity
Z_I_NW_SS % Marginal efficiency of investment shock in
% steady state
Z_L_NW_SS % Temporary labor augmenting technology shock in
% steady state
Z_RN3M_NW_SS % Monetary policy shock in steady state
Z_U_NW_SS % Consumption preference shock in steady state
std_E_DXT_NW % Standard deviation of the innovation to the
% Z_DXT_NW shock process
std_E_DZT_NW % Standard deviation of the innovation to the
% Z_DZT_NW shock process
std_E_I_NW % Standard deviation of the innovation to the
% Z_I_NW shock process
std_E_L_NW % Standard deviation of the innovation to the
% Z_L_NW shock process
std_E_PSI_NW % Standard deviation of the innovation to the
% PSI_NW shock process
std_E_RHO_NW % Standard deviation of the innovation to the
% RHO_NW shock process
std_E_RN3M_NW % Standard deviation of the innovation to the
% Z_RN3M_NW shock process
std_E_THETAH_NW % Standard deviation of the innovation to the
% THETAH_NW shock process
std_E_U_NW % Standard deviation of the innovation to the
```

```
% Z_U_NW shock process

unitrootvars
Z
X

model
% 1) Final goods sector
% Production function (A_NW)
A_NW = Q_NW;

% FOC (Q_NW)
REAL_PQ_NW = 1;

% 2) Intermediate goods sector
% Intermidate production function (KBAR_NW)
T_NW = (Z*Z_L_NW*L_NW)^(1-ALPHA_NW)*KBAR_NW^ALPHA_NW;

% Optimality condition wrt utilized capital (I.e. demand function) (MC_NW)
KBAR_NW = ALPHA_NW*(MC_NW/RK_NW)*T_NW;

% Optimality condition wrt aggregated labor (I.e. demand function) (L_NW)
L_NW = (1 - ALPHA_NW)*(MC_NW/REAL_W_NW)*T_NW;

% Pricing (REAL_PQ_NW)
Q_NW - THETAH_NW*Q_NW + MC_NW*THETAH_NW*Q_NW/REAL_PQ_NW
     - 100*PHI_PQ_NW*(DPQ_PQ_NW/DPQ_PQ_NW(-1) - 1)*DPQ_PQ_NW/DPQ_PQ_NW(-1)*Q_NW
     + DSA_NW*100*PHI_PQ_NW*(DPQ_PQ_NW(+1)/DPQ_PQ_NW - 1)
     *DPQ_PQ_NW(+1)^2/DPQ_PQ_NW*Q_NW(+1) = 0;

% 3) Investment producer
% Investment production function (Y_I_NW)
I_NW = Y_I_NW;

% First order condition for investment  (REAL_PI_NW)
REAL_PI_NW = 1;

% 4) Capital producer
% Capital production function (KNEW_NW)
KNEW_NW = Z_I_NW*(1 - S_NW)*I_NW;

% Investment adjustment cost function (S_NW)
S_NW = (PHI_I1_NW/2)*(I_NW/I_NW(-1) - bgp(I_NW))^2;

% Derivative of the investment adjustment cost function with respect to
% first input (Multiplied by I_NW!) (S_PRIME1_NW)
S_PRIME1_NW = PHI_I1_NW*(I_NW/I_NW(-1) - bgp(I_NW))*I_NW/I_NW(-1);

% Derivative of the investment adjustment cost function with respect to
% second input (Multiplied by I_NW!) (S_PRIME2_NW)
S_PRIME2_NW = -PHI_I1_NW*(I_NW/I_NW(-1) - bgp(I_NW))*(I_NW/I_NW(-1))^2;
```

```
% Optimal capital investment (I_NW)
0 = DSA_NW*REAL_PK_NW(+1)*Z_I_NW(+1)*S_PRIME2_NW(+1)
  + REAL_PK_NW*Z_I_NW*(1 - S_NW - S_PRIME1_NW) - REAL_PI_NW;

% 5) Households
 % Marignal  utility of consumption  (UPRIME_NW)
UPRIME_NW   = X*Z_U_NW*((C_NW - C_NW(-1)*BC_NW)/(1 - BC_NW/bgp(C_NW)))^(-1) ;

% Marginal  utility  of  labor labor (VPRIME_NW)
VPRIME_NW   = ((L_NW - BL_NW*L_NW(-1))/(1-BL_NW))^ZETA_NW ;

% Stochastic discount factor (DSA_NW)
DSA_NW = BETA_NW*RHO_NW(+1)*UPRIME_NW(+1)/(DPQ_P_NW(+1)*RHO_NW*UPRIME_NW);

% Consumption euler equation (RN3M_NW)
DSA_NW*RN3M_NW = 1;

% Optimal wage  setting  (REAL_W_NW)
REAL_W_NW = PSI_NW*MRS_NW/((PSI_NW-1)*(1-GAMMA_W_NW)
          + 1000*PHI_W_NW*DPQ_W_NW/DPQ_W_NW(-1)*(DPQ_W_NW/DPQ_W_NW(-1) - 1)
          - DSA_NW*DPQ_W_NW(+1)*L_NW(+1)/L_NW*1000*PHI_W_NW*DPQ_W_NW(+1)
          /DPQ_W_NW*(DPQ_W_NW(+1)/DPQ_W_NW - 1));

% Wage adjusment cost (GAMMA_W_NW)
GAMMA_W_NW = 1000*PHI_W_NW/2*(DPQ_W_NW/DPQ_W_NW(-1) - 1)^2 ;

% Marginal rate of substitution between consumption and leisure (MRS_NW)
MRS_NW = VPRIME_NW/UPRIME_NW;

%  6) Entrepreneurs
 % optimality with respect  to  capital  (REAL_PK_NW)
REAL_PK_NW = DSA_NW*DPQ_P_NW(+1)*( REAL_PK_NW(+1)*(1 - DELTA_NW) +
RK_NW(+1)*U_NW(+1) - GAMMA_U_NW(+1));

% optimality with respect to  utilization (RK_NW)
RK_NW = GAMMAPRIME_U_NW;

% Cost of utilizing the capital (GAMMA_U_NW)
GAMMA_U_NW = steady_state(RK_NW)/PHI_U_NW*(exp(PHI_U_NW*(U_NW - 1)) - 1) ;

% Marginal cost of utilizing capital (GAMMAPRIME_U_NW)
GAMMAPRIME_U_NW  = steady_state(RK_NW)*exp(PHI_U_NW*(U_NW - 1)) ;

% Capital accumulation (K_NW)
K_NW = (1-DELTA_NW)*K_NW(-1) + KNEW_NW;

% Capital utilization (U_NW)
KBAR_NW = U_NW*K_NW(-1);

% 7) Market clearing
 % Final good market clearing (C_NW)
A_NW = C_NW + Y_I_NW;
```

```
% Intermediate good market clearing  (T_NW)
T_NW = Q_NW;

%  8) Definitions
% Definition of natural output
NAT_Y_NW = A_NW;

% Consumption growth (DPQ_C_NW)
DPQ_C_NW = C_NW/C_NW(-1);

% Investment growth (DPQ_I_NW)
DPQ_I_NW = I_NW/I_NW(-1);

% Capital growth (DPQ_K_NW)
DPQ_K_NW = K_NW/K_NW(-1);

% Definition of natural output growth (DPQ_A_NW)
DPQ_Y_NW = NAT_Y_NW/NAT_Y_NW(-1);

% Definition of intermidate good inflation (DPQ_PQ_NW)
DPQ_PQ_NW = DPQ_P_NW*REAL_PQ_NW/REAL_PQ_NW(-1);

% Real investment inflation (DPQ_REAL_PI_NW)
DPQ_REAL_PI_NW = REAL_PI_NW/REAL_PI_NW(-1);

% Real wage inflation (DPQ_REAL_W_NW)
DPQ_REAL_W_NW = REAL_W_NW/REAL_W_NW(-1);

% Definition  of wage inflation (DPQ_W_NW)
DPQ_W_NW = REAL_W_NW/REAL_W_NW(-1)*DPQ_P_NW;

% 9)  Taylor rule (DPQ_P_NW)
RN3M_NW = RN3M_NW(-1)^OMEGA_R_NW*(steady_state(RN3M_NW)
        *(DPQ_P_NW/steady_state(DPQ_P_NW))^OMEGA_P_NW
        *(NAT_Y_NW/steady_state(NAT_Y_NW))^OMEGA_Y_NW
        *(DPQ_Y_NW/steady_state(DPQ_Y_NW))^OMEGA_DPQ_Y_NW)
        ^(1-OMEGA_R_NW)*(Z_RN3M_NW/steady_state(Z_RN3M_NW));
[static] DPQ_P_NW = DPQ_P_NW_SS;

% 10) Shock processes
% Labor market competition shock  (PSI_NW)
log(PSI_NW) = (1-LAMBDA_PSI_NW)*log(PSI_NW_SS) +
LAMBDA_PSI_NW*log(PSI_NW(-1)) + E_PSI_NW*std_E_PSI_NW ;

% Discount factor shock  (RHO_NW)
log(RHO_NW) = (1-LAMBDA_RHO_NW)*log(RHO_NW_SS) +
LAMBDA_RHO_NW*log(RHO_NW(-1)) + E_RHO_NW*std_E_RHO_NW ;

% Price markup shock  (THETAH_NW)
log(THETAH_NW) = (1-LAMBDA_THETAH_NW)*log(THETAH_NW_SS)
               + LAMBDA_THETAH_NW*log(THETAH_NW(-1))
```

```
               + E_THETAH_NW*std_E_THETAH_NW ;

% Marginal efficiency of investment shock  (Z_I_NW)
log(Z_I_NW) = (1-LAMBDA_I_NW)*log(Z_I_NW_SS)
            + LAMBDA_I_NW*log(Z_I_NW(-1)) + E_I_NW*std_E_I_NW ;

% Temporary labor augmenting technology shock  (Z_L_NW)
log(Z_L_NW) = (1-LAMBDA_L_NW)*log(Z_L_NW_SS)
            + LAMBDA_L_NW*log(Z_L_NW(-1)) + E_L_NW*std_E_L_NW ;

% Consumption preference shock  (Z_U_NW)
log(Z_U_NW) = (1-LAMBDA_U_NW)*log(Z_U_NW_SS)
            + LAMBDA_U_NW*log(Z_U_NW(-1)) + E_U_NW*std_E_U_NW ;

% Monetary policy shock  (Z_RN3M_NW)
log(Z_RN3M_NW)     = (1-LAMBDA_RN3M_NW)*log(Z_RN3M_NW_SS) +
LAMBDA_RN3M_NW*log(Z_RN3M_NW(-1)) + E_RN3M_NW*std_E_RN3M_NW;

% Unit root processes
Z/Z(-1) = DZT_NW_SS^(1-LAMBDA_DZT_NW)*(Z(-1)/Z(-2))^LAMBDA_DZT_NW*exp(std_E_DZT_NW*E_DZT_NW);
X/X(-1) = DXT_NW_SS^(1-LAMBDA_DXT_NW)*(X(-1)/X(-2))^LAMBDA_DXT_NW*exp(std_E_DXT_NW*E_DXT_NW);

% Construct reported variables that can be asked for in IRFs, but
% are not part of the model. Here you can use all MATLAB functions
% that act on a double vector, and returns an output with the same
% size as the input.
reporting

    C_NW_LEVEL = exp(cumsum(log(DPQ_C_NW) - log(steady_state(DPQ_C_NW))));
    I_NW_LEVEL = exp(cumsum(log(DPQ_I_NW) - log(steady_state(DPQ_I_NW))));
    K_NW_LEVEL = exp(cumsum(log(DPQ_K_NW) - log(steady_state(DPQ_K_NW))));
    Y_NW_LEVEL = exp(cumsum(log(DPQ_Y_NW) - log(steady_state(DPQ_Y_NW))));
```

With this file and the following commands to make the model stationary and compute impulse responses.

```
% Read the non-stationary model
modelNS = nb_dsge('nb_file','jpt_non_stationary_pref.nb');

% Give it a name
modelNS = set(modelNS,'name','Stationarize JPT automatically');

% Set the parameters
param                   = struct();
param.ALPHA_NW          = 0.167;
param.BC_NW             = 0.859;
param.BETA_NW           = 100/(0.134+100);
param.BL_NW             = 0;
param.DELTA_NW          = 0.025;
param.DPQ_P_NW_SS       = (0.702 + 100)/100;
param.DXT_NW_SS         = 1 + (0.597/100);
param.DZT_NW_SS         = 1 + (0.303 - (param.ALPHA_NW/...
                          (1 - param.ALPHA_NW))*0.597)/100;
```

```
param.LAMBDA_DXT_NW     = 0.156;
param.LAMBDA_DZT_NW     = 0.286;
param.LAMBDA_I_NW       = 0.772;
param.LAMBDA_L_NW       = 0;
param.LAMBDA_PSI_NW     = 0.967;
param.LAMBDA_RHO_NW     = 0.590;
param.LAMBDA_RN3M_NW    = 0;
param.LAMBDA_U_NW       = 0;
param.LAMBDA_THETAH_NW  = 0.971;
param.OMEGA_DPQ_Y_NW    = 0.208;
param.OMEGA_P_NW        = 1.709;
param.OMEGA_R_NW        = 0.858;
param.OMEGA_Y_NW        = 0.051;
param.PHI_PQ_NW         = 0.2;
param.PHI_I1_NW         = 2.657;
param.PHI_W_NW          = 1.0080;
param.PHI_U_NW          = 5.434;
param.PSI_NW_SS         = 1.135/(1.135 - 1);
param.RHO_NW_SS         = 1;
param.THETAH_NW_SS      = 1.171/(1.171 - 1);
param.ZETA_NW           = 4.444;
param.Z_I_NW_SS         = 1;
param.Z_L_NW_SS         = 1;
param.Z_U_NW_SS         = 1;
param.Z_RN3M_NW_SS      = 1;
param.std_E_DXT_NW      = 0.630;
param.std_E_DZT_NW      = 0.933;
param.std_E_I_NW        = 5.103;
param.std_E_L_NW        = 0;
param.std_E_PSI_NW      = 0.310;
param.std_E_RHO_NW      = 0.036;
param.std_E_RN3M_NW     = 0.210;
param.std_E_THETAH_NW   = 0.219;
param.std_E_U_NW        = 0;
modelNS                 = assignParameters(modelNS,param);

% Solve for the balanced growth path (steps 1-3 of the algorithm)
modelNS  = solveBalancedGrowthPath(modelNS);

% Stationarize the model (step 4 of the algorithm)
modelNS = stationarize(modelNS)


% Solve stationary steady state (numerically)
modelNS = checkSteadyState(modelNS,...
    'solver',               'fsolve',...
    'steady_state_solve',   true,...
    'steady_state_default', @ones);

% Return the steady state solution as a cell matrix
ss = getSteadyState(modelNS)

% Return the solution to the balanced growth path as a cell matrix
bgp = getBalancedGrowthPath(modelNS)

% Write the stationary model to file (optional)
```

```
writeModel2File(modelNS,'stationarized_pref.nb')

% Obtain the state space representation of  the  solution
% y(t) = A y(t-1) + B eps(t)
modelNS = solve(modelNS);

% compute and  plot  impulse responses
[~,~,plotter] = irf([modelNS,modelS],'periods',  60,'plotSS',   true,'shocks',   ...
    {'E_DXT_NW'},...
   'variables',{'C_NW','DPQ_P_NW','DPQ_W_NW','I_NW','K_NW',...
                'L_NW','NAT_Y_NW','RN3M_NW'},...
  'settings',{'legBox','off','legFontSize',18,'subPlotSize',[3,3],...
                'figureTitle',false,'lookUpMatrix','lookUpMatrixJPT',...
               'legends',{'Stationarize JPT automatically',...
                          'Stationarize JPT manually',...
                          '',...
                          'Steady state'}});
nb_graphInfoStructGUI(plotter)
```

H Generating results from the JPT model

In the main paper we have calculated the steady state, balanced growth path and moments. In this section we present the code on how you can calculate these tables using NB toolbox.

```
%% Parametrization

param                    = struct();
param.ALPHA_NW           = 0.167;
param.BC_NW              = 0.859;
param.BETA_NW            = 100/(0.134+100);
param.BL_NW              = 0;
param.DELTA_NW           = 0.025;
param.DPQ_P_NW_SS        = (0.702 + 100)/100;
param.DUT_NW_SS          = 1 + (0.597/100);
param.DZT_NW_SS          = 1 + (0.303 - (param.ALPHA_NW/...
                           (1 - param.ALPHA_NW))*0.597)/100;
param.LAMBDA_DUT_NW      = 0.156;
param.LAMBDA_DZT_NW      = 0.286;
param.LAMBDA_I_NW        = 0.772;
param.LAMBDA_L_NW        = 0;
param.LAMBDA_PSI_NW      = 0.967;
param.LAMBDA_RHO_NW      = 0.590;
param.LAMBDA_RN3M_NW     = 0;
param.LAMBDA_U_NW        = 0;
param.LAMBDA_THETAH_NW   = 0.971;
param.OMEGA_DPQ_Y_NW     = 0.208;
param.OMEGA_P_NW         = 1.709;
param.OMEGA_R_NW         = 0.858;
param.OMEGA_Y_NW         = 0.051;
param.PHI_PQ_NW          = 0.2;
param.PHI_I1_NW          = 2.657;
param.PHI_W_NW           = 1.0080;
```

```
param.PHI_U_NW          = 5.434;
param.PSI_NW_SS         = 1.135/(1.135 - 1);
param.RHO_NW_SS         = 1;
param.THETAH_NW_SS      = 1.171/(1.171 - 1);
param.ZETA_NW           = 4.444;
param.Z_I_NW_SS         = 1;
param.Z_L_NW_SS         = 1;
param.Z_U_NW_SS         = 1;
param.Z_RN3M_NW_SS      = 1;
param.std_E_DUT_NW      = 0.630;
param.std_E_DZT_NW      = 0.933;
param.std_E_I_NW        = 5.103;
param.std_E_L_NW        = 0;
param.std_E_PSI_NW      = 0.310;
param.std_E_RHO_NW      = 0.036;
param.std_E_RN3M_NW     = 0.210;
param.std_E_THETAH_NW   = 0.219;
param.std_E_U_NW        = 0;

%% Read the non-stationary model

modelNS = nb_dsge('nb_file','jpt_non_stationary.nb');
modelNS = set(modelNS,'name','JPT with investment specific technology');

%% Assign parameters

modelNS = assignParameters(modelNS,param);

%% Solve for the balanced growth path

modelNS  = solveBalancedGrowthPath(modelNS);

%% Stationarize the non-stationary model

modelNS = stationarize(modelNS);

%% Solve steady state numerically

ssInit = struct(...
    'GAMMA_W_NW',0,...
    'PSI_NW',param.PSI_NW_SS,...
    'S_NW',0,...
    'S_PRIME1_NW',0,...
    'S_PRIME2_NW',0,...
    'GAMMA_U_NW',0,...
    'THETAH_NW',param.THETAH_NW_SS);

modelNS = checkSteadyState(modelNS,...
    'solver',               'fsolve',...
    'steady_state_solve',   true,...
    'steady_state_init',    ssInit,...
    'steady_state_default', @ones);

%% Solve stationary model

modelNS = solve(modelNS);
```

```
%% Parametrization

paramP                    = struct();
paramP.ALPHA_NW           = 0.167;
paramP.BC_NW              = 0.859;
paramP.BETA_NW            = 100/(0.134+100);
paramP.BL_NW              = 0;
paramP.DELTA_NW           = 0.025;
paramP.DPQ_P_NW_SS        = (0.702 + 100)/100;
paramP.DXT_NW_SS          = 1 + (0.597/100);
paramP.DZT_NW_SS          = 1 + (0.303 - (param.ALPHA_NW/...
                            (1 - param.ALPHA_NW))*0.597)/100;
paramP.LAMBDA_DXT_NW      = 0.156;
paramP.LAMBDA_DZT_NW      = 0.286;
paramP.LAMBDA_I_NW        = 0.772;
paramP.LAMBDA_L_NW        = 0;
paramP.LAMBDA_PSI_NW      = 0.967;
paramP.LAMBDA_RHO_NW      = 0.590;
paramP.LAMBDA_RN3M_NW     = 0;
paramP.LAMBDA_U_NW        = 0;
paramP.LAMBDA_THETAH_NW   = 0.971;
paramP.OMEGA_DPQ_Y_NW     = 0.208;
paramP.OMEGA_P_NW         = 1.709;
paramP.OMEGA_R_NW         = 0.858;
paramP.OMEGA_Y_NW         = 0.051;
paramP.PHI_PQ_NW          = 0.2;
paramP.PHI_I1_NW          = 2.657;
paramP.PHI_W_NW           = 1.0080;
paramP.PHI_U_NW           = 5.434;
paramP.PSI_NW_SS          = 1.135/(1.135 - 1);
paramP.RHO_NW_SS          = 1;
paramP.THETAH_NW_SS       = 1.171/(1.171 - 1);
paramP.ZETA_NW            = 4.444;
paramP.Z_I_NW_SS          = 1;
paramP.Z_L_NW_SS          = 1;
paramP.Z_U_NW_SS          = 1;
paramP.Z_RN3M_NW_SS       = 1;
paramP.std_E_DXT_NW       = 0.630;
paramP.std_E_DZT_NW       = 0.933;
paramP.std_E_I_NW         = 5.103;
paramP.std_E_L_NW         = 0;
paramP.std_E_PSI_NW       = 0.310;
paramP.std_E_RHO_NW       = 0.036;
paramP.std_E_RN3M_NW      = 0.210;
paramP.std_E_THETAH_NW    = 0.219;
paramP.std_E_U_NW         = 0;

%% Read the non-stationary model

modelNSPref = nb_dsge('nb_file','jpt_non_stationary_pref.nb');
modelNSPref = set(modelNSPref,'name','JPT with unit root in preferences');

%% Assign parameters

modelNSPref = assignParameters(modelNSPref,paramP);
```

```
%% Solve for the balanced growth path

modelNSPref = solveBalancedGrowthPath(modelNSPref);

%% Stationarize the non-stationary model

modelNSPref = stationarize(modelNSPref);

%% Solve steady state numerically

ssInit = struct(...
    'GAMMA_W_NW',0,...
    'PSI_NW',paramPref.PSI_NW_SS,...
    'S_NW',0,...
    'S_PRIME1_NW',0,...
    'S_PRIME2_NW',0,...
    'GAMMA_U_NW',0,...
    'THETAH_NW',paramPref.THETAH_NW_SS);

modelNSPref = checkSteadyState(modelNSPref,...
    'solver',               'fsolve',...
    'steady_state_solve',   true,...
    'steady_state_init',    ssInit,...
    'steady_state_default', @ones);

%% Solve stationary model

modelNSPref = solve(modelNSPref);

%% Compare balanced growth paths

ssGrowth    = getBalancedGrowthPath([modelNS,modelNSPref],...
    {'C_NW','I_NW','K_NW','L_NW','NAT_Y_NW','REAL_W_NW',...
     'REAL_PI_NW'},'headers')

%% Compare steady state results

ss = getSteadyState([modelNS,modelNSPref],...
    {'C_NW','DPQ_P_NW','DPQ_W_NW','I_NW','K_NW',...
     'L_NW','NAT_Y_NW','RN3M_NW'},...
     'headers')

%% Theoretical moments

vars = {'C_NW','DPQ_P_NW','DPQ_W_NW','I_NW','K_NW',...
        'L_NW','NAT_Y_NW','RN3M_NW'};

[~,CNS] = theoreticalMoments(modelNS,'vars',vars,...
            'type','covariance');
VNS     = diag(CNS)';
VNS     = rename(VNS,'variable','diag',getName(modelNS));

[~,CNSPref] = theoreticalMoments(modelNSPref,'vars',vars,...
                'type','covariance');
VNSPref     = diag(CNSPref)';
```

```
VNSPref     = rename(VNSPref,'variable','diag',getName(modelNSPref));
```

I The automatically made stationary JPT model file with unit root in preferences

```
endogenous
Z_U_NW Z_RN3M_NW Z_L_NW Z_I_NW Y_I_NW VPRIME_NW U_NW UPRIME_NW T_NW
THETAH_NW S_PRIME2_NW S_PRIME1_NW S_NW RN3M_NW RK_NW RHO_NW REAL_W_NW
REAL_PQ_NW REAL_PK_NW REAL_PI_NW Q_NW PSI_NW NAT_Y_NW MRS_NW MC_NW
L_NW K_NW KNEW_NW KBAR_NW I_NW GAMMA_W_NW GAMMA_U_NW GAMMAPRIME_U_NW
D_Z_Z D_Z_Y_I_NW D_Z_X D_Z_VPRIME_NW D_Z_UPRIME_NW D_Z_T_NW D_Z_REAL_W_NW
D_Z_Q_NW D_Z_NAT_Y_NW D_Z_MRS_NW D_Z_L_NW D_Z_K_NW D_Z_KNEW_NW
D_Z_KBAR_NW D_Z_I_NW D_Z_C_NW D_Z_A_NW DSA_NW DPQ_Y_NW DPQ_W_NW
DPQ_REAL_W_NW DPQ_REAL_PI_NW DPQ_P_NW DPQ_PQ_NW DPQ_I_NW DPQ_C_NW C_NW A_NW

exogenous
E_DXT_NW E_DZT_NW E_I_NW E_L_NW E_PSI_NW E_RHO_NW E_RN3M_NW E_THETAH_NW
E_U_NW

parameters
std_E_U_NW std_E_THETAH_NW std_E_RN3M_NW std_E_RHO_NW std_E_PSI_NW
std_E_L_NW std_E_I_NW std_E_DZT_NW std_E_DXT_NW Z_U_NW_SS Z_RN3M_NW_SS
Z_L_NW_SS Z_I_NW_SS ZETA_NW THETAH_NW_SS RHO_NW_SS PSI_NW_SS PHI_W_NW
PHI_U_NW PHI_PQ_NW PHI_I1_NW OMEGA_Y_NW OMEGA_R_NW OMEGA_P_NW
OMEGA_DPQ_Y_NW LAMBDA_U_NW LAMBDA_THETAH_NW LAMBDA_RN3M_NW
LAMBDA_RHO_NW LAMBDA_PSI_NW LAMBDA_L_NW LAMBDA_I_NW LAMBDA_DZT_NW
LAMBDA_DXT_NW DZT_NW_SS DXT_NW_SS DPQ_P_NW_SS DELTA_NW BL_NW
BETA_NW BC_NW ALPHA_NW

model

A_NW-Q_NW;
REAL_PQ_NW-1;
T_NW = (((1*Z_L_NW)*L_NW)^(1-ALPHA_NW))*(KBAR_NW^ALPHA_NW);
KBAR_NW = (ALPHA_NW*(MC_NW/RK_NW))*T_NW;
L_NW = ((1-ALPHA_NW)*(MC_NW/REAL_W_NW))*T_NW;
Q_NW-(THETAH_NW*Q_NW)+((MC_NW*THETAH_NW)*Q_NW)/REAL_PQ_NW-
(((((100*PHI_PQ_NW)*(DPQ_PQ_NW/DPQ_PQ_NW(-1)-1))*DPQ_PQ_NW)/DPQ_PQ_NW(-1))*Q_NW)+
(((((DSA_NW*100)*PHI_PQ_NW)*(DPQ_PQ_NW(+1)/DPQ_PQ_NW-1))*
(DPQ_PQ_NW(+1)^2))/DPQ_PQ_NW)*(Q_NW(+1)*D_Z_Q_NW(+1))-0;
I_NW-Y_I_NW;
REAL_PI_NW-1;
KNEW_NW = (Z_I_NW*(1-S_NW))*I_NW;
S_NW = (PHI_I1_NW/2)*((I_NW/(I_NW(-1)*D_Z_I_NW^-1)-
steady_state(D_Z_I_NW))^2);
S_PRIME1_NW = ((PHI_I1_NW*(I_NW/(I_NW(-1)*D_Z_I_NW^-1)-
steady_state(D_Z_I_NW)))*I_NW)/(I_NW(-1)*D_Z_I_NW^-1);
S_PRIME2_NW = ((-PHI_I1_NW)*(I_NW/(I_NW(-1)*D_Z_I_NW^-1)-
steady_state(D_Z_I_NW)))*((I_NW/(I_NW(-1)*D_Z_I_NW^-1))^2);
0 = ((DSA_NW*REAL_PK_NW(+1))*Z_I_NW(+1))*S_PRIME2_NW(+1)+
```

```
(REAL_PK_NW*Z_I_NW)*(1-S_NW-S_PRIME1_NW)-REAL_PI_NW;
UPRIME_NW = (1*Z_U_NW)*(((C_NW-((C_NW(-1)*D_Z_C_NW^-1)*BC_NW))/
(1-(BC_NW/steady_state(D_Z_C_NW))))^(-1));
VPRIME_NW = ((L_NW-(BL_NW*(L_NW(-1)*D_Z_L_NW^-1)))/(1-BL_NW))^ZETA_NW;
DSA_NW = ((BETA_NW*RHO_NW(+1))*(UPRIME_NW(+1)*D_Z_UPRIME_NW(+1)))/
((DPQ_P_NW(+1)*RHO_NW)*UPRIME_NW);
DSA_NW*RN3M_NW-1;
REAL_W_NW = (PSI_NW*MRS_NW)/((PSI_NW-1)*(1-GAMMA_W_NW)+
(((1000*PHI_W_NW)*DPQ_W_NW)/DPQ_W_NW(-1))*(DPQ_W_NW/DPQ_W_NW(-1)-1)-
(((((((DSA_NW*DPQ_W_NW(+1))*(L_NW(+1)*D_Z_L_NW(+1)))/L_NW)*1000)*PHI_W_NW)*
DPQ_W_NW(+1))/DPQ_W_NW)*(DPQ_W_NW(+1)/DPQ_W_NW-1)));
GAMMA_W_NW = ((1000*PHI_W_NW)/2)*((DPQ_W_NW/DPQ_W_NW(-1)-1)^2);
MRS_NW = VPRIME_NW/UPRIME_NW;
REAL_PK_NW = (DSA_NW*DPQ_P_NW(+1))*(REAL_PK_NW(+1)*(1-DELTA_NW)+
RK_NW(+1)*U_NW(+1)-GAMMA_U_NW(+1));
RK_NW-GAMMAPRIME_U_NW;
GAMMA_U_NW = (steady_state(RK_NW)/PHI_U_NW)*(exp(PHI_U_NW*(U_NW-1))-1);
GAMMAPRIME_U_NW = steady_state(RK_NW)*(exp(PHI_U_NW*(U_NW-1)));
K_NW = (1-DELTA_NW)*(K_NW(-1)*D_Z_K_NW^-1)+KNEW_NW;
KBAR_NW = U_NW*(K_NW(-1)*D_Z_K_NW^-1);
A_NW = C_NW+Y_I_NW;
T_NW-Q_NW;
NAT_Y_NW-A_NW;
DPQ_C_NW = C_NW/(C_NW(-1)*D_Z_C_NW^-1);
DPQ_I_NW = I_NW/(I_NW(-1)*D_Z_I_NW^-1);
DPQ_Y_NW = NAT_Y_NW/(NAT_Y_NW(-1)*D_Z_NAT_Y_NW^-1);
DPQ_PQ_NW = (DPQ_P_NW*REAL_PQ_NW)/REAL_PQ_NW(-1);
DPQ_REAL_PI_NW = REAL_PI_NW/REAL_PI_NW(-1);
DPQ_REAL_W_NW = REAL_W_NW/(REAL_W_NW(-1)*D_Z_REAL_W_NW^-1);
DPQ_W_NW = (REAL_W_NW/(REAL_W_NW(-1)*D_Z_REAL_W_NW^-1))*DPQ_P_NW;
RN3M_NW = ((RN3M_NW(-1)^OMEGA_R_NW)*((((steady_state(RN3M_NW)*
((DPQ_P_NW/steady_state(DPQ_P_NW))^OMEGA_P_NW))*
((NAT_Y_NW/steady_state(NAT_Y_NW))^OMEGA_Y_NW))*
((DPQ_Y_NW/steady_state(DPQ_Y_NW))^OMEGA_DPQ_Y_NW))^(1-OMEGA_R_NW)))*
(Z_RN3M_NW/steady_state(Z_RN3M_NW));
[static] DPQ_P_NW = DPQ_P_NW_SS;
log(PSI_NW) = (1-LAMBDA_PSI_NW)*log(PSI_NW_SS)+
LAMBDA_PSI_NW*log(PSI_NW(-1))+E_PSI_NW*std_E_PSI_NW;
log(RHO_NW) = (1-LAMBDA_RHO_NW)*log(RHO_NW_SS)+
LAMBDA_RHO_NW*log(RHO_NW(-1))+E_RHO_NW*std_E_RHO_NW;
log(THETAH_NW) = (1-LAMBDA_THETAH_NW)*log(THETAH_NW_SS)+
LAMBDA_THETAH_NW*log(THETAH_NW(-1))+E_THETAH_NW*std_E_THETAH_NW;
log(Z_I_NW) = (1-LAMBDA_I_NW)*log(Z_I_NW_SS)+
LAMBDA_I_NW*log(Z_I_NW(-1))+E_I_NW*std_E_I_NW;
log(Z_L_NW) = (1-LAMBDA_L_NW)*log(Z_L_NW_SS)+
LAMBDA_L_NW*log(Z_L_NW(-1))+E_L_NW*std_E_L_NW;
log(Z_U_NW) = (1-LAMBDA_U_NW)*log(Z_U_NW_SS)+
LAMBDA_U_NW*log(Z_U_NW(-1))+E_U_NW*std_E_U_NW;
log(Z_RN3M_NW) = (1-LAMBDA_RN3M_NW)*log(Z_RN3M_NW_SS)+
LAMBDA_RN3M_NW*log(Z_RN3M_NW(-1))+E_RN3M_NW*std_E_RN3M_NW;
```

The unit root processes specified

```
1/(1*D_Z_Z^-1) = ((DZT_NW_SS^(1-LAMBDA_DZT_NW))*
(((1*D_Z_Z^-1)/(1*D_Z_Z^-1*D_Z_Z(-1)^-1))^LAMBDA_DZT_NW))*
(exp(std_E_DZT_NW*E_DZT_NW));
1/(1*D_Z_X^-1) = ((DXT_NW_SS^(1-LAMBDA_DXT_NW))*
(((1*D_Z_X^-1)/(1*D_Z_X^-1*D_Z_X(-1)^-1))^LAMBDA_DXT_NW))*
(exp(std_E_DXT_NW*E_DXT_NW));
```

The identified G^i functions

```
log(D_Z_A_NW)-log(D_Z_Q_NW);
ALPHA_NW*log(D_Z_L_NW)+ALPHA_NW*log(D_Z_Z)+log(D_Z_T_NW)-
ALPHA_NW*log(D_Z_KBAR_NW)-log(D_Z_L_NW)-log(D_Z_Z);
log(D_Z_KBAR_NW)-log(D_Z_T_NW);
log(D_Z_L_NW)+log(D_Z_REAL_W_NW)-log(D_Z_T_NW);
log(D_Z_I_NW)-log(D_Z_Y_I_NW);
log(D_Z_KNEW_NW)-log(D_Z_I_NW);
log(D_Z_C_NW)+log(D_Z_UPRIME_NW)-log(D_Z_X);
log(D_Z_VPRIME_NW)-ZETA_NW*log(D_Z_L_NW);
log(D_Z_REAL_W_NW)-log(D_Z_MRS_NW);
log(D_Z_MRS_NW)+log(D_Z_UPRIME_NW)-log(D_Z_VPRIME_NW);
log(D_Z_K_NW)-log(D_Z_KNEW_NW);
log(D_Z_KBAR_NW)-log(D_Z_K_NW);
log(D_Z_A_NW)-log(D_Z_C_NW);
log(D_Z_A_NW)-log(D_Z_Y_I_NW);
log(D_Z_NAT_Y_NW)-log(D_Z_A_NW);
```

Finally the reporting:

```
reporting
    C_NW_LEVEL = exp(cumsum(log(DPQ_C_NW) - log(steady_state(DPQ_C_NW))));
    I_NW_LEVEL = exp(cumsum(log(DPQ_I_NW) - log(steady_state(DPQ_I_NW))));
    K_NW_LEVEL = exp(cumsum(log(DPQ_K_NW) - log(steady_state(DPQ_K_NW))));
    Y_NW_LEVEL = exp(cumsum(log(DPQ_Y_NW) - log(steady_state(DPQ_Y_NW))));
```

J The model file to automatically solve the endogenous RBC model

```
endogenous

c dA i k l pa r s srd ups upsprime y


exogenous

e u


unitrootvars

A


parameters

beta delta delta_a g gamma lambda_srd srd_ss std_e std_u theta zeta
```

```
model
    % Optimality condition of the households
    c(+1)/c           = beta*(1 + r);
    l                 = 1;
    y                 = (A(-1)*l)^gamma * k(-1)^(1-gamma);
    (1-gamma)*y(+1)/k = r + delta;
    1 + r             = (pa(+1)/pa)*(1 - delta_a) + (1/pa)*gamma*y(+1)/A;
    k                 = (1-delta)*k(-1) + i;

    % Optimality condition of the capital producers
    log(srd) = (1 - lambda_srd)*log(srd_ss) + lambda_srd*log(srd(-1))
        + std_e*e;
    pa*(A(-1)/A)*upsprime*exp(std_u*u) = 1 - srd;
    ups = g + theta/(1 - 1/zeta)*(s/A)^(1 - 1/zeta);
    upsprime = theta*(s/A)^(-1/zeta);
    A = (1 - delta_a + ups*exp(std_u*u))*A(-1);

    % Market clearing
    y = c + i + s;

    % Other
    dA = A/A(-1);

% Construct reported variables not part of the model.
reporting

    A         = exp(cumsum(log(dA) - log(steady_state(dA))));
    C         = c*A;      I         = i*A;
    K         = k*A;     S         = s*A;      Y         = y*A;
    c_log_dev = exp(log(C) - log(C(1)));
    i_log_dev = exp(log(I) - log(I(1)));
    k_log_dev = exp(log(K) - log(K(1)));
    s_log_dev = exp(log(S) - log(S(1)));
    r_dev     = r - steady_state(r);
```

```
    srd_dev   = srd - steady_state(srd);
    y_log_dev = exp(log(Y) - log(Y(1)));
```

K Comparison with Dynare

The dynare code to solve the RBC model of section 5.1 is the following

```
var dA;
trend_var(growth_factor=dA) A;
var l, r;
var(deflator=A) c, i, k, y;

varexo u;
parameters beta, delta, g, gamma, lambda, std_u;
g      = 1.03;  gamma  = 0.60;  delta  = 0.10;
beta   = 0.97;  lambda = 0;  std_u  = 0.01;

model;
      % Model equations in non-stationary form
    c(+1)/c           = beta*(1 + r); // c
    l                 = 1; // l
    y                 = (A*l)^gamma * k(-1)^(1-gamma); // k
    (1-gamma)*y(+1)/k = r + delta; // r
    k                 = (1-delta)*k(-1) + i; // i
    y                 = c + i; // y
    dA               = g^(1-lambda)*dA(-1)^lambda*exp(std_u*u);
end;

shocks;
var u; stderr 1;
end;

initval;
k = 1;  dA = g;  c = 1;  l = 1; y = 1;  k = 1;  r = 1;  i = 1;
end;
```

```
steady;
check;
stoch_simul;
```

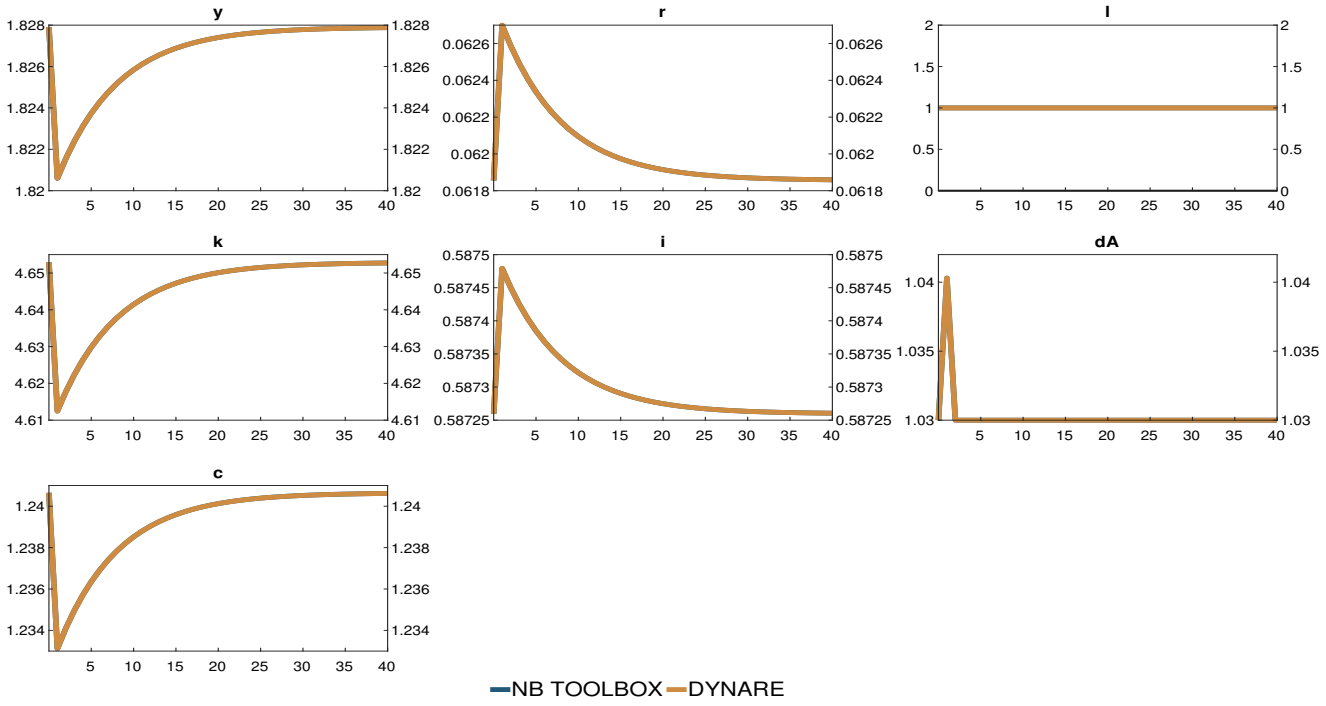

Figure 5: Responses to a technology shock in the RBC model of section 5.1

The responses we obtain are in the figure 5. Superimposed are those obtained solving the model with NB toolbox.