Handout for Extended Model of the Monetary Business Cycle

©OGResearch

This note introduces a reduced-form new Keynesian model that can be calibrated to fit stylized facts in most countries and to prepare an inflation forecast. It is an extended version of a generic Quarterly Projection Model used in several central banks [Beneš et al., 2008], and other institutions such as the IMF [Berg et al., 2006]. The objective of this class of models is to help decide on an appropriate level of the policy interest rate, given the inflation target and the current state of the economy.

The model is solved numerically using the Matlab software and IRIS Toolbox. The solution algorithm and commands are explained in this note. We focus on three practical aspects of the modeling process in the context of inflation targeting: (i) calibration; (ii) testing the model properties and data preparation and filtration; and (iii) inflation forecast, its analysis and interpretation.

Before We Start

Before you can start modeling in the IRIS environment, you need to install IRIS and eventually a few other freeware programs (MiKTeX, Ghostscript, and Ghostgum) on your computer. Detailed instructions for IRIS installation are contained in OGResearch 'Welcome to IRIS' handout and we suggest that you follow them as closely as possible.

Start each session (after you open Matlab) by running the following line from the Matlab command window: `addpath C:/Iris_Tbx_8_20110430; irisstartup` replacing c:/Iris_Tbx_8_20110430 with your own main IRIS folder chosen during installation. Navigate to your working folder to check that it contains the following files (they will be described in more detail later):

1. model.model

2. readmodel.m

3. modelproperties.m

4. makedata.m

5. kalmanfilter.m

6. forecast.m

7. alternatives.m

8. data.csv

With the exception of `model.model` and `data.csv`, all files are Matlab programs (extension `.m`) and work in the Matlab environment only. The programs get started in the following way. Type the name of a program, for instance `makedata`, in the command window and push [enter]. Once the screen shows >> the program has been run successfully. You can check that all programs are functioning by executing them in the following sequence: `modelproperties`, `makedata`, `kalmanfilter`, `forecast`, `alternatives`. From Windows explorer, in the appropriate folder, double click on files that are generated to report simulation results: `Report_Shocks.pdf`, `Stylized_facts.pdf`, `Filtration.pdf`, `Forecast.pdf` and `Compare.pdf`. Click [OK] and you should see charts and tables of various model outputs. The output is generated in Acrobat (.pdf) format and can be opened using either Acrobat Reader or GSview.

The Model

The model we use for our exercises is a reduced-form, New Keynesian model, sometimes also called the 'gap' model. Similar versions of it are being utilized in many central banks all around the world. The model consists of four basic behavioral equations and several identities.[1] The behavioral equations are:

- Aggregate Demand (IS curve)

---

[1] The model code is contained in the `model.model` file. It is a text file that can be open, in any text editor. Be aware that the Microsoft Word is not a text editor though. You may use the Matlab editor, if necessary.

- Aggregate Supply Block (Phillips curves)

- Uncovered Interest Rate Parity

- Monetary Policy Rule

Aggregate Demand

The aggregate spending relationship corresponds to the open economy version of the traditional IS curve and takes the form:

$$\hat{y}_t = a_1\hat{y}_{t-1} + a_2 mci_t + a_3\hat{y}_t^* + \epsilon_t^y \tag{1}$$

where $\hat{y}_t$ is the deviation of the log of output from its noninflationary level, i.e., the output gap; mci_t is the real monetary condition index that is defined as a weighted average of deviations of the long-term real interest rate, r_t, from its neutral (noninflationary) level, and deviation of the real exchange rate, z_t, from its trend level, $mci_t = a_4\hat{r}_t + (1 - a_4)(-\hat{z}_t)$; $\hat{y}_t^*$ is the foreign output gap and ϵ_t^y is an aggregate demand shock. The aggregate demand shocks are governed by a Normal distribution and have no serial correlation. The coefficients a_1, a_2, and a_3 capture the persistence of output; the impact of monetary conditions on real economic activity; and the impact of foreign environment, respectively.

Aggregate Supply Block

The 'Phillips curve block' consists of six equations:

$$
\begin{align}
\pi_t^{core} &= b_1\pi_{t-1}^{core} + (1 - b_1)E_t\pi_{t+1} + b_2\hat{\varphi}_t + \epsilon_t \tag{2}\\
\hat{\varphi}_t &= b_3\hat{y}_t + (1 - b_3)\hat{z}_t \tag{3}\\
\pi_t^{food} &= b_{21}\pi_{t-1}^{food} + (1 - b_{21})E_t\pi_{t+1} + b_{22}\hat{\psi}_t + \eta_t \tag{4}\\
\hat{\psi}_t &= b_{23}\hat{z}_t^{food} + (1 - b_{23})\hat{y}_t \tag{5}\\
\pi_t^{oil} &= b_{31}\pi_{t-1}^{oil} + (1 - b_{31} - b_{32})E_t\pi_{t+1} + b_{32}(\Delta p_t^{oil} + \Delta s_t^{USD} - \Delta\bar{z}_t) + \nu_t \tag{6}\\
\pi_t &= w^{oil}\pi_t^{oil} + w^{food}\pi_t^{food} + (1 - w^{oil} - w^{food})\pi_t^{core} + \varepsilon_t \tag{7}
\end{align}
$$

Equation (2) represents the standard forward-looking Phillips curve for core inflation (π_t^{core}), where the core inflation, defined as headline CPI excluded for food and oil components, depends on the overall inflation expectations ($E_t\pi_{t+1}$), past core inflation values (π_{t-1}^{core}) and on the current value of the real marginal costs ($\hat{\varphi}_t$). The real marginal costs (see equation (3)) is related to domestic producers (approximated by the output gap, $\hat{y}_t$) as well as to importers (approximated by the real exchange rate gap, $\hat{z}_t$). It is important to model expectations properly, capturing the behavior of agents, some of which may be forward looking, using model consistent (rational) expectations, $E_t\pi_{t+1}$, while others are backward looking. Economic agents who are assumed to be fully forward looking comprise $(1 - b_1)$ of the population and b_1 agents follow the rule of thumb of past inflation. Another way of thinking about this parameter is the persistence of inflation - the more persistent inflation, the higher is b_1. The coefficient b_2 captures the influence of the gap in the real marginal costs on inflation (the slope of the Phillips curve) and combination of b_2 and b_3 measures the sacrifice ratio, i.e., how much output will be lost in order to bring inflation down by 1 percentage point.

Equation (4) captures for inflation of food prices (π_t^{food}). While it closely resembles a forward-looking Phillips curve, it is defined for food prices part of the CPI basket only. The food inflation depends on the overall inflation expectations ($E_t\pi_{t+1}$), lagged food inflation values (π_{t-1}^{food}) and on the current value of the real marginal costs ($\hat{\psi}_t$) of food good retailers. It is the definition of real marginal costs what makes the equation for food prices different from the one for core inflation. The real marginal cost for the food goods retailers is calculated as combination of the output gap (low weight) and the gap in the relative food price $\hat{z}_t^{food}$ (see equation (5)). The gap in the relative food price is calculated using the world food price index in US dollars, nominal exchange rate against the US dollar and the domestic food price index; $z_t^{food} = wp_t^{food} + s_t^{USD} - cpi_t^{food}$. In other words, the domestic prices of food goods are driven by the de-trended relative world and domestic price of food goods adjusted for the nominal exchange rate. This relation between the domestic food prices and the relative price of domestic and world food prices as defined above is usually strongly supported in the data. Similarly to core inflation Phillips curve, coefficient b_{21} captures backward verus forward-lookigness of food retailers, coefficient b_{22} the influence of real marginal costs on food prices

and b_{23} the impact of world food prices versus domestic business cycle (output gap) on food prices. Naturally, the passthrough from output gap to food prices is often rather limited, i.e. the value of b_{23} is close to one.

Evolution of domestic oil prices is represented by equation (6). In comparison to core and food inflation, here the pass-through from world oil prices to domestic prices is done in a simple way where the domestic oil inflation depends on its past value and changes in the world oil prices (Δp_t^{oil}) as well as changes in the nominal exchange rate against the US dollar (Δs_t^{USD}). Again, coefficient b_{31} captures persistency in oil prices evolution, while coefficient b_{32} the impact of world oil prices on domestic oil price. Together, both coefficient determine the speed of passthrough from world do domestic oil prices. The influence of expectations is clearly limited with sum of b_{31} and b_{32} being usually close to one.

Finally equation (7) sums core, food and oil inflations to determine the headline inflation (π_t) based on the weights taken from the CPI basket (w^{food}, w^{oil}).

Uncovered Interest Rate Parity (UIP)

We capture the relationship with the rest of the world using alternative versions of the uncovered interest rate parity condition. In its pure forward-looking version, the UIP relates the behavior of domestic and foreign interest rates, and the nominal exchange rate:

$$s_t = s_{t+1} + (i_t^* - i_t + prem_t)/4 + \epsilon_t^s \tag{8}$$

where s_t is the nominal exchange rate; s_{t+1}^e is the model consistent expectation of the nominal exchange rate in period $t+1$; i_t is the domestic nominal interest rate (annualized); i_t^* is the foreign nominal interest rate (annualized); $prem_t$ is the risk premium; and ϵ_t^s is the exchange rate shock. With the canonical UIP, the model has difficulty in matching the observed behavior of the economy (looking, say, at impulse responses), because the exchange rate has little persistence [Beneš et al., 2008]. The pure forward-looking element in Equation (8) forces the current exchange rate level to adjust immediately to the sum of all future interest rate differentials implied by the model behavior. This is at odds with the observed exchange rates that tend to move more slowly.

There are a few alternative ways of increasing the persistence of the exchange rate. One way is to modify the pure UIP by limiting its forward-looking nature. This approach substitutes the model-consistent exchange rate expectations in Equation (8) by a combination of backward-looking and model-consistent (forward-looking) expectations ($E_t s_{t+1}$). This can also be interpreted as assuming backward-looking expectations for a portion of the agents in the economy and forward-looking expectations for the remaining. The approach below is more general than that of [Berg et al., 2006], allowing for a nonzero rate of growth of the exchange rate in the long-run. Equation for exchange rate expectations (s_{t+1}^E) takes the form:

$$s_{t+1}^E = e E_t s_{t+1} + (1 - e)(s_{t-1} + \frac{2}{4}\Delta\bar{s}_t) \tag{9}$$

where

$$\Delta\bar{s}_t = \bar{\pi}_t - \bar{\pi}_t^* + \Delta\bar{z}_t \tag{10}$$

In this setting, $\bar{\pi}_t$ is domestic inflation target; $\bar{\pi}_t^*$ is foreign inflation target; and $\Delta\bar{z}_t$ is the trend (long-run) change in the real exchange rate. The coefficient e determines the degree of forward-looking behavior in the financial market or the portion of the agents that are assumed to be forward-looking.

The second element in Equation (9), ($s_{t-1} + \frac{2}{4}\Delta\bar{s}_t$), is the backward-looking exchange rate expectation, which projects the exchange rate in period $t + 1$ as an extrapolation of the past exchange rate using the trend rate of growth of the real exchange rate and the average inflation differential approximated by the difference in inflation targets. While such expectations are not model-consistent in the short-run, they are consistent in the long-run, in line with the finding that the Purchasing Power Parity holds at longer horizons only. The term $\Delta\bar{s}_t$ is the change in the exchange rate consistent with long-term economic fundamentals represented by the inflation targets and the real exchange rate trend. By construction, $\Delta\bar{s}_t = \Delta s_t$ in the long-run, so the long-run properties of the model are intact.

The above extension of the UIP is designed to make the nominal exchange rate more persistent, however, it does not reduce the overall volatility of the exchange rate. The fluctuations in the nominal exchange rate are more persistent, but its standard deviation is not necessarily smaller. In the model code we call this 'modified UIP with a backward-looking element (persistent exchange rate)'

$$s_t = (1 - e_1)s_{t+1} + e_1(s_{t-1} + \frac{2}{4}(\bar{\pi}_t - \bar{\pi}_t^* + \Delta\bar{z}_t)) + (i_t^* - i_t + prem_t)/4 + \epsilon_t^s \tag{11}$$

where setting e_1 equal to zero collapses this modified UIP into the original UIP relationship of Equation (8).

Alternatively, the UIP can be modified to account for the central banks that use the FOREX interventions actively (and the exchange rate channel) to meet the inflation target.[2] However, to model the exchange rate consistently, the change in the exchange rate target must be defined as a 'sum' of the inflation differential and trend appreciation as in Equation (10). Moreover, to avoid unrealistic exchange rate behavior in the first simulated period, the last observed level of the exchange rate needs to be taken as the target at time $t-1$. The targeted exchange rate is

$$s_t^T = s_{t-1}^T + \Delta\bar{s}_t \tag{12}$$
$$s_{t-1}^T = s_{t-1} \tag{13}$$

where s_t^T is the exchange rate target at time t. Thus, first, Equation (10) determines the long-run change in the nominal exchange rate as implied by the Purchasing Power Parity theory and makes the change in the exchange rate target consistent with the chosen inflation target. Second, the target for the level of the nominal exchange rate is determined by its connection to the last observed level of the nominal exchange rate.

The UIP then becomes a 'weighted average' of the exchange rate target and the canonical UIP

$$s_t = e_1 s_t^T + (1 - e_1)(E_t s_{t+1} + (i_t^* - i_t + prem_t)/4) + \epsilon_t^s \tag{14}$$

with the coefficient e_1 measuring the degree of the central bank intervention on the FOREX market. The extended UIP defined as in Equation (14) is known to stabilize the exchange rate in a way that resembles a managed exchange rate regime [Beneš *et al.*, 2008].

The stability of the exchange rate, however, comes at the cost of persistent deviations of inflation from the inflation target whenever a shock hits the economy. Intuitively, the introduction of an exchange rate target is equivalent to switching from an inflation target to a price level target. Even if the central bank publicly defines its target as a rate of growth of the price level, it simply means that the target is a price level growing at some constant rate. Any overshooting of such an inflation target must still be compensated by subsequent undershooting and *vice versa* to jointly satisfy the inflation and exchange rate targets. Models with this version of the UIP exhibit oscillations in inflation that die out gradually.

The model also satisfies the long-run version of the UIP expressed in real terms. This version of the UIP links the trend values for real exchange rate appreciation (due to the Balassa-Samuelson effect) to the trend values of domestic and foreign real interest rates

$$\Delta\bar{z}_{t+1} = \bar{r}_t - \bar{r}_t^* - prem_t \tag{15}$$

The steady-state values of $\Delta\bar{z}_t, \bar{r}_t$ and $\bar{r}_t^*$ are set as parameters in this model, requiring prior assessment of these trend values,[3] while the risk premium is calculated endogenously, assuring the existence of a consistent steady state. Values of those parameters frame the forecast over the medium to long run.

Monetary Policy Rule

The model is closed by a policy reaction function of the monetary authority. The rule is defined typically for a central bank maintaining a floating exchange rate regime, however, it can be modified to reflect the objective of controlling the exchange rate.

For simplicity, we take the three month interest rate to be the instrument of monetary policy, and the authority is assumed to respond to deviations of next period inflation from its target and to the output gap. In other words, we assume that the credit markets flawlessly transmit the changes in the policy rate into the money-market rates. The last period policy stance affects the current policy stance allowing the authority to smooth interest rates by adjusting them gradually to the desired level implied by the deviations of inflation and output from equilibrium. Mathematically,

$$i_t = f_1 i_{t-1} + (1 - f_1)(i_t^n + f_2(E_t \pi_{t+1} - \bar{\pi}_{t+1}) + f_3 \hat{y}_t) + \epsilon_t^i \tag{16}$$

where i_t is the domestic short-term nominal interest rate and ϵ_t^i is a policy shock. The monetary authority is fully forward-looking and uses model consistent inflation expectations, $E_t \pi_{t+1}$.[4] The policy neutral rate, i_t^n, is such that

---

[2]A similar structure of the UIP is used in the forecasting models in Botswana, Belarus, Ukraine, and Serbia [Beneš *et al.*, 2008].

[3]For example, one could assume that a fast-converging transition economy would have an equilibrium real appreciation to the tune of 3-5 percent annually.

[4]Also the foreign monetary authority is assumed to obey a policy rule: $i_t^* = 0.7 i_{t-1}^* + (1 - 0.7)(i_t^n + 2(E_t \pi_{t+1}^* - \bar{\pi}_{t+1}^*) + f_3 \hat{y}_t^*) + \epsilon_t^{i*}$, where the superscript $*$ denotes the foreign country.

would keep the output gap unchanged. We calculate it as the sum of the trend real interest rate and model consistent inflation expectations:

$$i_t^n = \bar{r}_t + E_t \pi_{t+1} \tag{17}$$

Central banks attempting to control the exchange rate using foreign exchange interventions find it difficult to control simultaneously also the money market and hence the short-term nominal interest rate. The domestic money market interest rate i_t becomes partly determined by the UIP equation and the policy rule needs to be modified as follows

$$i_t = g_1(E_t \Delta s_{t+1} + i_t^* + prem_t) + (1 - g_1)(f_1 i_{t-1} + (1 - f_1)(i_t^n + f_2(E_t \pi_{t+1} - \bar{\pi}_{t+1}) + f_3 \hat{y}_t)) + \epsilon_t^i \tag{18}$$

The coefficient $g - 1$ reflects the degree of control the central bank retains over the domestic money market (if $g_1 = 0$, then the bank has retained full control; if $g_1 = 1$ the bank has lost control). Of course, the extended version of the policy rule needs to be dovetailed with the uncovered interest parity equation (Equation (14)) and the e_1 coefficient ('high' values of g_1 require 'high' values of e_1, and vice versa). Setting simultaneously a 'low' value of e_1 and a 'high' value of g_1 would lead to system indeterminacy - the exchange rate is floating ('low e_1'), while the central bank does not control the interest rate ('high' g_1) and, hence, nothing anchors the price level. Intuitively, central banks choose to stabilize inflation through controlling either the exchange rate or interest rate.

For simplicity we work with the three month interest rate and do not use long term interest rates. The model includes further identities and transformations that are reported in the model codes.

Model Solution

The solution of this model involves four steps. The main complication arises due to the need for calculating model consistent expectations as explained below. Without the calculation of model consistent expectations, the problem at hand is quite straightforward to solve in many mathematical software as it is a system of four linear equations with four unknowns. Solution of the full model with expectations also requires solving the same set of four linear equations not just once but potentially many times.

To see the intuition of how the solution algorithm works, imagine that the only expectations operator is in the Phillips curve. See below that on the right hand side of the equation, one period ahead expectation of inflation appears while the left hand side is the contemporaneous inflation which we are trying to solve for. We run into a chicken and eggs problem because to be able to solve for contemporaneous inflation we need to know expected inflation that appears on the rights hand side. Similarly, we cannot calculate expected inflation unless we know contemporaneous inflation as a function of other endogenous and exogenous variables.

$$\pi_t = b_1 \pi_{t-1} + (1 - b_1)E_t \pi_{t+1} + b_2 \hat{\varphi}_t + \epsilon_t^\pi \tag{19}$$

In such circumstances typically the solution requires iterations, in other words, specifying a guess for the contemporaneous inflation and using this guess to calculate the right hand side of the Phillips curve which would imply a contemporaneous inflation that does not need to be the same as the initial guess. At this point the initial guess would be modified depending on how different the inflation implied by the Phillips curve is. Then this new guess would be used to recalculate the right hand side of the Phillips curve which will yield yet another inflation figure. The guess would be updated similarly and this procedure would continue until the initial guess is close enough to the inflation implied by the Phillips curve.

Below are more detailed and specific steps on how the algorithm works:

1. Guess inflation. Let's call it π^{old}. Note that here the superscript "old" does not refer to time, it is the guess used in the first iteration. If there is prior information about how the solution for inflation will look like, this information can be incorporated into the guess. If not, a guess of $\pi^{old} = 0$ should work.

2. Use the Phillips curve to recalculate π. Specifically, calculate $b_1 \pi_{t-1} + (1 - b_1)E_t \pi_{t+1} + b_2 \hat{\varphi}_t + \epsilon_t^\pi$ using π^{old} to evaluate the expected inflation. This calculation will deliver a different inflation: π^{new}. Note that at this point $\pi^{old} \neq \pi^{new}$. There is no reason for these two inflation values to be the same unless the initial inflation guess was "very educated."

3. Update initial inflation guess with the new guess: $\pi^{old} = \pi^{new}$.

4. Repeat steps 2-3 until $\|\pi^{old} - \pi^{new}\| \leq \xi$ where ξ is a small number. Intuitively, one would need to continue iterating until the absolute value of the difference between the two values of inflation is small enough. This is called iterating until 'convergence'. The higher the accuracy needed from this numerical solution, the smaller ξ needs to be.

Model Calibration

The baseline version of the model is based on 'reasonable' values for all the parameters, however, the model exhibits comparatively low persistence with these parameters. The calibration process can be split into two parts: 1) the calibration of the trend variables, determining the steady state levels of the model, 2) the calibration of individual equations parameters, pinning down the business cycle properties of the model.

Calibration of Long-Term Parameters

The model converges to the long-term trend values of the real exchange rate, domestic and foreign real interest rates, and risk premium and we impose these values via parameters. For simplicity, also the targets for domestic and foreign inflation are set as parameters. The choice of the inflation target predetermines the inflation forecast - the forecasted inflation converges to the target. The model structure set in Equations (1) to (18) determines important long run relationships. The nominal interest rate converges to the neutral rate that in turn equals to the sum of the trend real interest rate and the inflation target. (In the steady-state, inflation expectations equal the inflation target). The steady-state change in the nominal exchange rate becomes the sum of steady-state inflation differential vis--vis the world and steady-state real exchange rate appreciation. The risk premium is calculated endogenously given the parameterization of domestic and foreign real interest rates and the change in the real exchange rate. This approach not only keeps the model simple but also ensures that the trend values in Equations (10) and (15) are on a consistent path.

The model requires setting six long-run parameters:

1. Inflation target

2. Foreign long-run inflation (or foreign inflation target)

3. Foreign trend real interest rate

4. Equilibrium real exchange rate appreciation/depreciation

5. Potential output growth

We recommend that after changing to the model structure you check the steady state consistency of the model. This is done by first typing `[m,p,mss] = readmodel(false)` and pushing [ENTER] in the Matlab command window, instructing the program to solve the model steady-state and reduced form, while replacing the old model parameters with the new ones. Then type `mss` - the command returns information about the model solution, its parameterization and steady-state. The screen message is self-explanatory and we suggest checking the following:

1. The domestic (including all components) and foreign inflation rates equal their respective inflation targets.

2. The domestic and foreign real interest rates equal their trend parameters.

3. The change in the real exchange rate equals the assumed trend.

4. The risk premium equals the difference between domestic and foreign real interest rates minus the change in the real exchange rate (equation 15).

5. The change in the nominal exchange rate must equal the inflation differential adjusted for the change in the real exchange rate.

6. All the gaps are zeros.

Violating any of these conditions will result in either inflation missing its target at the end of the forecast period or a non-convergence of the model. The most likely mistakes are:

1. A wrong parameter sign.

2. Violated linear homogeneity in one of the relevant equations. For example, you may have inadvertently set $f_1 \geq 1$ in the Taylor rule. See Table 1 for further references on linear homogeneity conditions.

3. Explosive paths in some equations. For example, you may have inadvertently set $a_1 \geq 1$ in the IS curve and your output gap is 'exploding'. See Table 1.

Calibration: Business Cycle properties

In this section we discuss model calibration to reflect country specific, business cycle conditions by modifying parameters a_1 to g_1 in the file `readmodel`. Tables 1, 2 and 3 summarize the baseline parameterization of the model.[5]

Following [Berg *et al.*, 2006], in setting the parameters values we follow (i) the economic theory; (ii) international experience; and (iii) domestic economy stylized facts, based, for example, on impulse responses of a structural VAR model. Estimated values, if available, may be used only to the extent that they do not violate model assumptions. For example, short-sample regressions for fast growing emerging market economies are known to suggest that inflation declines when output expands above its potential (the estimated b_3 coefficient in the Phillips curve is negative). However, using a negative value for b_3 would violate the model convergence toward its steady state.

Table 1: Baseline Parameters: Aggregate Demand

a_1	**Output gap persistence** varies between 0.1 (extremely low persistence) to 0.95 (high persistence). The nonexplosive path requires: $0 < a_1 < 1$. *Calibration*: Run an OLS regression of a log of output on its lagged value and a trend (linear, Hodrick Prescott, etc.)
a_2	**Pass-through from monetary conditions to real economy.** The value varies between 0.1 (low impact) to 0.5 (high impact); the higher the parameter the more responsive is the output gap to changes in monetary policy and, hence, policy reactions need to be less pronounced. The nonexplosive path requires: $0.1 < a_2 < 0.9$. *Calibration*: We suggest basing this parameter either on an impulse-response function in a structural VAR (if available) or on expert assessment.
a_3	**Impact of foreign demand on the output gap** typically varies from 0.1 to 0.5. *Calibration*: Base the calibration on the export to GDP ratio.
a_4	**The relative weight of the real interest rate and real exchange rate** in real monetary conditions in the IS curve (*mci*). The value varies between 0.3 (open economy) and 0.8 (closed economy). *Calibration*: Expert assessment.

In addition to bivariate regressions, structural VAR models and their impulse responses may be used for parameter calibration of exchange rate or output shocks. Many such VARs have been published on national central bank websites. We would caution, however, against using the impulse responses to the policy shock as the model we present is designed to capture systematic policy behavior. In other words, the monetary authority is assumed to fight against the shocks as opposed to creating them and the policy pass-through estimated in VAR model is likely to be underestimated.

To summarize, the calibration exercise includes the following steps. First, start with the parameter values established in the theory. Second, look for a reasonable estimate of the IS curve, Phillips curve, or policy rule. If there is no such estimate available, run a simple bivariate regression yourself to get a sense of the persistence in the economy. Third, look for a structural VAR model already estimated for the economy you are calibrating, focusing especially on the exchange rate pass-through.

Some Useful Calibration Hints

1. *To make the policy response more (less) sluggish*, increase (decrease) the persistence in the policy rule (f_1). Be aware, however, that by increasing (decreasing) persistence in the policy rule you *ceteris paribus* increase (decrease) the impact of monetary policy on the real economy (via both the real interest and exchange rates).

---

[5]The model uses a few additional parameters that we suggest keeping unchanged (t_1, h_0, h_1 and h_2). The parameter h_1 determines the speed of convergence of domestic variable trend values with respect to past estimated values and to their steady-state values and is set to 0.7. For example, the trend real interest rate reflects past trend rates and the steady-state real interest rate: $\bar{r}_t = h_1 \bar{r}_{t-1} + (1-h_1)\bar{r}_{ss}$. The parameter h_2 determines the persistence of the foreign output gap: $\hat{y}_t^* = h_2 \hat{y}_{t-1}^* + \epsilon_t^{y*}$, and so on. See the file `model.model` for the code and `readmodel.m` for calibrations.

Table 2: Baseline Parameters: Aggregate Supply Block

b_1	**Inflation persistence** determines the share of forward looking versus backward looking agents on the goods markets. The value varies between 0.4 (low persistence) and 0.9 (high persistence), the higher is the share of forward looking agents (the lower is the parameter). The less persistent the model becomes, the less pronounced policy reactions are required for a given disinflation goal. The linear homogeneity condition: $0 < b_1 < 1$, otherwise monetary policy cannot bring inflation toward the target with zero gaps. *Calibration*: Run an OLS regression of the rate of inflation (quarter-on-quarter) on its lagged value and complement with expert assessment.
b_2	**The impact of real marginal costs on inflation (policy pass-through)**. The value typically varies between 0.1 (low impact and high sacrifice ratio) to 0.5 (strong impact and low sacrifice ratio). The higher the parameter the less costly is disinflation. *Calibration*: Run an OLS regression of the rate of inflation on output gap and complement with expert assessment. Alternatively, data permitting, you can compare the output gap and decline in inflation during a clearly defined disinflation period (Ball, 1993). The latter technique is unlikely to produce meaningful results for emerging/transition economies.
b_3	**The relative weight of output gap and real exchange rate gap in firms real marginal costs**. $(1 - b_3)$ is the share of imported goods in the consumption good basket. The value typically varies between 0.9 (relatively closed economy) and 0.6 (open economy). *Calibration*: Base the calibration on the share of imported goods in the CPI basket. While such a number is rarely published by the statistical office, some central banks calculated such breakdowns.
b_{21}	**Food prices persistence** determines the forward versus backward looking behavior of food retailers. The value varies between 0.2 (low persistence) and 0.7 (high persistence). Generally it applies that the food prices are less persistent than is the core inflation. The linear homogeneity condition applies and b_{21} must be set between zero and one. *Calibration*: Run an OLS regression of the rate of food inflation (quarter-on-quarter) on its lagged value and complement with expert assessment.
b_{22}	**The impact of real marginal costs on food prices.** The value varies between 0.1 to 0.5. The speed of the passthough from world food prices to domestic prices depends on both b_{21} and b_{22}. *Calibration*: Vary the coefficient, run historical forecasts and check the fit.
b_{23}	**The relative weight of relative food prices output gap in food retailers real marginal costs.** [b_{23} is the weight on the impact of relative food prices (world food prices)]. The value is varies between 0.7 (relatively high food market protection) and 1 (no food market protection). *Calibration*: Base the calibration on correlation between the domestic and world food prices.
b_{31}	**Oil prices persistence** determines the forward versus backward looking behavior of oil prices. The value varies between 0.2 (low persistence) and 0.7 (high persistence). Generally it applies that the oil prices are quite volatile and follow the world oil prices closely. *Calibration*: Run an OLS regression of the rate of oil inflation (quarter-on-quarter) on its lagged value and complement with expert assessment.
b_{32}	**The impact of world oil prices on domestic oil prices.** The value varies between 0.3 to 0.5. The speed of the passthough from world oil prices to domestic prices depends on both b_{31} and b_{32}. Linear homogeineity condition applies, sum of b_{31} and b_{32} mustnt exceed one. *Calibration*: Vary the coefficient, run historical forecasts and check the fit.

2. *To make the whole economy more persistent*, increase (decrease) the persistence parameter in the Phillips curve (b_1). As a result, a more (less) pronounced policy reaction will be needed for a given disinflation goal.

3. *To force a faster (slower) convergence toward the trend ('steady state') values*, increase (decrease) the policy pass-through either through the real monetary condition parameter in the IS curve (a_2) or the real marginal cost parameter in the Phillips curve (b_2). As a result, monetary policy becomes more (less) powerful.

4. *To make the exchange rate path more (less) persistent or stable*, increase (decrease) the persistence (interventions) parameter (e_1) in the modified UIP equations.

Table 3: Baseline Parameters: UIP, Policy Rule and the Rest

e_1	**Exchange rate persistence or central bank presence on the FOREX market**. The value varies between zero (forward looking FOREX market or no central bank interventions) to 0.9 (either heavily backward looking FOREX agents or a central bank heavily intervening on the FOREX market). *Caveat*: Interventions stabilize the exchange rate but cannot violate the steady-state consistency of the model, e.g. the exchange rate will never appreciate as long as there is on average a positive inflation differential $\pi \geq \pi^*$ and no trend real exchange rate appreciation. The nonexplosive path requires: $0 < e_1 < 1$. *Calibration*: Expert assessment.
f_1	**Policy rate persistence in the Taylor rule**. The value varies between zero (no persistence in policy setting) to 0.8 ('wait-and-see' policy). *Caveat*: a value above 0.7 may be too high when the model is calibrated as backward looking. The linear homogeneity condition imposes $0 < f_1 < 1$, otherwise the model does not converge. *Calibration*: Run an OLS regression of the policy rate on its lagged value and complement with expert assessment.
f_2	**Weight put by the policy maker on deviations of inflation from the target in the policy rule**. Ranges typically vary from 0.3 to 1. The Taylor principle requires that $f_2 > 0$, otherwise monetary policy does not stabilize the economy. *Calibration*: Expert assessment.
f_3	**Weight put by the policy maker on output gap in the policy rule**. The value of f_3 has to be positive, $f_3 > 0$, otherwise the model does not converge. Typically varies from 0.3 to 1. *Calibration*: Expert assessment.
g_1	**Central banks control of the domestic money market and its short-term nominal interest rate**. The value varies between 0 (full control of the short-term rates) and 1 (no control of the short-term rates). The latter case ($g1 = 1$) implies that the central bank uses nominal exchange rate to stabilize inflation by intervening in the FOREX market. *Caveat*: high values of this coefficient in combination with low values of the e_1 parameter invalidate the model solution. *Calibration*: Expert assessment.
t_1	**Speed of exchange rate adjustment from actual data to the exchange rate target**. This coefficient is used only if an exchange-rate-target adjusted UIP is used. Plausible values are: $0 < t_1 < 0.5$.
h_0	**Persistence parameter in the auxiliary risk premium equation**. *Calibration*: Expert assessment.
h_1	**Persistence parameters in various auxiliary equations (trend appreciation, equilibrium real interest rate, potential growth, and so on)**. The values of these coefficients matter mainly if you decide to shock the underlying variables, for example, to model a temporary departure in trend appreciation. *Calibration*: Expert assessment.
h_2	**Persistence parameter in the AR(1) process for foreign output gap**. *Calibration*: As in the domestic IS curve ($a1$).

5. *To make the transmission mechanism really sluggish*, you may introduce a lag for monetary conditions in the IS curve (a_2) or a lag for the real marginal cost in the Phillips curve (b_2) instead of the current values present in the canonical version.

You should exercise caution in increasing persistence in the economy. There exists a point, not easily defined in advance, when the economic agents may become too persistent for the central bank to be able to stabilize the economy. For example, if you make both the Phillips curve and the UIP fully backward looking, you should leave at least the policy rule to be forward looking, otherwise the model may become unstable.

How to Check the Model Properties

After the model has been calibrated, you may want to check visually the behavior of the model economy. You will do it by running the program `modelproperties` in the Matlab command window and then inspecting six pages of printouts in a file `Report_shocks.pdf`:

1. Aggregate demand shock

2. Inflationary shock (core inflation)

3. World food price shock

4. World oil price shock

5. Exchange rate shock

6. Monetary policy shock

Note that if you open the `Report_shocks.pdf` file using the GSview (right click on the file in Windows explorer, choose 'Open with .. and choose GSview), you can (i) change the model calibration by `edit` `readmodel`; (ii) save it; (iii) run `modelproperties` without having to close and open the `Report_shocks.pdf` file every time you run the `modelproperties` program. While open in GSview the `Report_shocks.pdf` file changes with the change in model calibration - a useful property when fine tuning the model. If you decide to change the model structure, for instance by including the lagged output gap in the Phillips curve, change the code in the `model.model` file as follows:

```
%% Phillips Curve (Aggregate supply)
dot_cpi_x = b1*dot_cpi_x{-1} + (1-b1)*dot_cpi_x{+1} + b2*rmc{-1} + e_dot_cpi_x;
```

Save the file and run the file `modelproperties`.

Forecasting

This section discusses the use of the model for a basic forecasting exercise, the goal of which is to decide on an appropriate level of the policy rate. Rather than providing an unconstrained forecast of inflation, this class of models brings inflation close to the target by construction and the focus is on the path of the policy rate that would be consistent with such an inflation rate. The policy rate path depends on (i) the model structure, (ii) model parameterization, and (iii) initial conditions. We have discussed the model structure and parameterization in the preceding section and discuss below the importance of initial conditions.

Database Preparation

A historical database, which provides initial conditions, is necessary to run the model forward. As a minimum, the database must contain a value for any variable that appears with a lag in the model structure. Thus, for a forecast starting in 2012Q1, we need initial conditions for 2011Q4 for all the variables with the lag of order one. In practice, however, the database is prepared and maintained using a longer time series data for all the relevant variables. For example, some of the variables needed for the initial conditions may be unobserved (output gap) and have to be estimated using the historical series. The longer the data set, the more precise is the estimation of unobserved variables.

The program `makedata` is set to prepare the database for inspection without manual calculations of seasonal adjustment, de-trending or chart preparation. Nevertheless, before running `makedata` in Matlab a basic data set needs to be compiled:

These series - compiled in any program, including Microsoft Excel - need to be saved into the file `data.csv` in the text format (`csv`).[6] For more details see the template file `data.csv`. Four your country series you have to use exactly the same names as in `data.csv`, otherwise the code would not recognize your series.

If any of your series is not seasonally adjusted, extend the series title with a suffix _u (for instance from cpi to cpi_u). The program `makedata` will recognize seasonally unadjusted series and will run the seasonal adjustment (Census X12) automatically, saving the newly seasonally adjusted series without the suffix _u. The program `makedata` calculates the natural logarithm of all series (except interest rates), growth rates for relevant variables, and the real interest rate. It also estimates the trend and gap for real output, real exchange rate and domestic and foreign real interest rate. Here band-pass filter is used for the estimation of the output gap (domestic and foreign) and Hodrick-Prescott filter for the estimation of gap and trend for real exchange rate and domestic and foreign real interest rate.[7]

Visual inspection of the data is a useful check of your prior intuition about the economy in question. Program `makedata` creates panels of domestic and foreign inflation, nominal interest rates, exchange rates and estimated

---

[6]In Excel, select Save As; in the 'Save As' window click on 'Save as type'; scroll down to CSV (MS-DOS); and select this option.

[7]For detailed discussion of data handling and filtering in IRIS and Matlab see the OGResearch manual 'Welcome to IRIS' and 'Statistical Filters in IRIS'.

Table 4: Data Requirements

Series	Format
Domestic economy	
GDP at constant prices	Billions of national currency or an index
CPI	Index (not a rate of growth)
CPI components	Index (not a rate of growth)
Nominal interest rate	Annualized rate, e.g., 5%
Nominal exchange rate	Units of domestic currency per one unit of foreign currency
Foreign economy	
GDP at constant prices	Billions of national currency or an index
CPI	Index (not a rate of growth)
Nominal interest rate	Annualized rate, e.g., 5%
World food prices	Index (in USD, IMF World Economic Outlook, for instance)
World oil prices	Index or average price per barrel (both in USD)
USD/EUR nominal exchange rate	If your nominal exchange rate is against the EUR

trends and gaps. It should be possible to recognize most mistakes or omissions in the `data.csv` file just by inspecting the figures. We recommend checking especially the estimates of trends and gaps, as these will influence the forecast the most. The estimates of trends and gaps should correspond to your priors about the economy.

If the estimated trends or gaps seem biased, you may overwrite them manually (see below).[8] In fact, it suffices to change the last observation (initial condition) only, as only this information matters for the forecast. Note that with the exception of the year-on-year inflation rate the model contains no lags of order higher than one.

For example, the Hodrick-Prescott filter may generate an unreasonable trend for domestic real interest rates, while your priors suggest that this trend rate is in fact equal to, say, 1 percent annually. To manually change the database:

1. Type edit `makedata` in Matlab command window [enter]

2. Go to the section `% Expert change in the database` and type: d.rr_eq(qq(2011,4)) = 1;

3. Save `makedata` in the editor and run `makedata` in the Matlab command window.

Your domestic trend real interest rate is now set to equal one in the fourth quarter of 2011. It is sufficient to change the last observation only as previous observations do not influence the forecast - the forecast depends only on the initial conditions.

The Method of Kalman Filtration

The final initial conditions (final database) may be based on the method of Kalman filtration rather than band-pass and Hodrick-Prescott filters. After running the program `makedata` you may run program `kalmanfilter`. The program estimates all the unobserved variables, gaps and trends, based on the observed variables and the model structure, ensuring that the results are model- and calibration-dependent.

A reduced-form of the model serves as a starting point for the estimation of gaps and trends based on the method of Kalman filtration. The Kalman filter applies a reduced-form of the model extended for measurement equations that map observed variables to the unobserved. Together they represent the state description of model. The form is as follows:

$$y_t = Zx_t + \epsilon_t \tag{20}$$
$$x_t = Tx_{t-1} + \nu_t \tag{21}$$

where x denotes the vector of unobserved state variables, y denotes the vector of observed (measurement) variables, ϵ is a random vector we call process noise, and ν is the measurement noise. Furthermore, we assume a Gaussian distribution of the random vectors and of the x state vectors initial state. Based on the state form of the model and using observed variables, the Kalman filter identifies all unobserved variables that are a part of the model, i.e. gaps,

---

[8]For example, the Hodrick-Prescott filter may be an inappropriate technique given the presence of a structural break in the series or because of an end-period measurement error.

trends, and shocks. For linear systems it represents an optimum estimate in terms of the least squares criterion. Application of the filter itself takes on the recursive algorithm form, wherein the conditional probability density of state variables gets updated based on observed variables. Variables used as observed in estimation for all countries are set out in Table 4. The first step of the algorithm under the Kalman filtration is the prediction step. During the

Table 5: Observed Variables for the Kalman Filtration

CPI (level)	World food prices (q-o-q)
CPI food prices (q-o-q)	World oil prices (q-o-q)
CPI oil prices (q-o-q)	Cross exchange rate (USD/EUR, q-o-q)
GDP (level)	Foreign GDP gap (HP or band-pass filter)
Nominal exchange rate	Foreign CPI (level)
3M nominal interest rate (pa)	Foreign 3M nominal interest rate (pa)

step, equation (21) is used to estimate the predictive probability density of states at time t based on the previous conditional probability density at time $t - 1$. This probability density is however inexistent in the first period of the data sample and is therefore substituted with a random vector with a mean value and the unconditional variance of state variables described by the equation (21). Due to the presence of trends within the model and resulting non-stationarity of certain variables, the unconditional variance does not have a finite value and a diffuse Kalman filter need to be applied.

The filtration step follows after the prediction step, representing an update of the predictive probability density based on the information contained in the observed data. The measurement equation (20) is used for that purpose. Additional information drawn from the observed data enables a refined estimate of the state variables, also including the estimate of shocks. In addition to the above Kalman filter steps, we use also a smoothing step of the filter which, as opposed to the prediction and filtration steps, uses complete information from the observed data.

Results are saved in a graphical form in to file `Filtration.pdf` and as data in to files `kalm_his.csv`. The latter serves as your forecasting database.

The Forecast

Mechanically, preparation of the forecast is the shortest part of the whole process, however, its quality depends on previous work (calibration and data preparation) and ex post analysis. The forecast is produced by running the program `forecast`. This program reads the model (`[m,p,mss] = readmodel(false)`) and historical data (initial conditions) from the database and simulates the model forward. The program produces indicator figures and tables that can be viewed by doubleclicking on the file `Forecast.pdf` (or right click and Open with .. choose GSview). As the data source you may use either the database based on statistical filtration (`history.csv`) or method of Kalman filtration (`kalm_his.csv`). You switch between those two databases by changing the name of the respective file in command `h = dbload(...)`. Reminder: You need to run `makedata` and `kalmanfilter` any time you change your data before you run the program `forecast`.

The Programming Code of `forecast`

The following section briefly summarizes the code of the program `forecast`. The program starts with reading the model and data into the objects `m` and `h` respectively and sets the forecast range. You can change the range depending on your current data sample. The key command for running the forecast is simulate with the following syntax:

```
s = simulate(m,h,fcastrange);
```

The simulation results are saved into the database object `s` and the command `dbextend` adds `h` to `s`, overlaying the historical data with the current projections. The whole database is saved into the file `fcastdata.csv` (by the command `dbsave`) and the figure and tables are saved into the `Forecast.pdf`.

How to Pre-Set Future Values

You may want to set one or more future variables to an exact number, either because you do not believe that the models AR(1) process forecasts a particular variable well (for example, some of the foreign variables) or because some near term variables are better forecast by time-series models (for example, inflation one period ahead). You save the dataset `data.csv` with these future values and tell the program `forecast` to use these values using the command `plan` with the following syntax:

```
simplan = plan(m,fcastrange);
```

The command plan creates an object `simplan` that initially contains an empty plan for model `m` over the forecasting

range. In the next step you specify which variable are to be taken from `data.csv` using commands exogenize and endogenize. Assuming we want to pre-set the external rate of inflation and interest rates, the syntax is as follows:

```
simplan = exogenize(simplan,'dot_x_cpi','x_rn',fcastrange);
```

The command exogenize specifies which variables are going to be taken as exogenous variables and decides over which horizon this will hold (in this case over the whole forecast horizon; if you want to pre-set these variables only for the next two quarters, replace `fcastrange` with `qq(2011,2):qq(2011,3)`, and so on). Pre-setting a variable causes, however, a discrepancy between the left hand side of the respective equations (the pre-set values) and their right-hand sides (the data generating processes). Both sides of the equation are equalized through the residual term, which now becomes an endogenous variable, see endogenize:

```
simplan = endogenize(simplan,'e_dot_x_cpi','e_x_rn',fcastrange);
```

Intuitively, you can pre-set only those variables that have a residual term in their equations specified in model.model.

To summarize, if you decide to pre-set the two variables above, in the program `forecast` put a percentage sign % in front of `s = simulate(m,h,fcastrange)` and remove the percentage signs from the following four lines:

```
simplan = plan(m,fcastrange);
simplan = exogenize(simplan,'dot_x_cpi','x_rn',fcastrange);
simplan = endogenize(simplan,'e_dot_x_cpi','e_x_rn',fcastrange);
s = simulate(m,h,fcastrange,'plan',simplan);
```

Interpreting the Forecast

The simulations of your model need to be carefully interpreted and this section suggests some steps toward this objective. All relevant results are contained in file `Forecast.pdf` in the form of a figure and five tables.

Note that the inflation forecast converges to the targeted inflation, possibly with some initial fluctuation. This is a normal property of the so-called unconditional forecast where the monetary authority does all it can to keep the inflation rate close to the targeted one within the transmission period (usually 6 8 quarters). The most interesting variable is thus the trajectory of nominal interest rate set by the central bank rather than the forecasted inflation rate. The simulated path of the policy interest rate provides a policy reaction given the past data and the model structure.

In addition to inflation and the nominal interest rate, the figure shows the forecast of the nominal exchange rate, output gap, and monetary conditions. The last chart - the real interest and exchange rate gaps - plots both the direct and indirect exchange rate channels and the interest channel. Based on this chart, one can observe how tight the monetary stance must be (the real interest and exchange rate gaps) in order to meet the inflation target and what would be the cost of this stance in terms of real economic activity (the output gap).

The forecast printout also contains five tables: Main Indicators, Inflation Decomposition, Demand and Supply, Policy Decomposition and Foreign Variables. The first table is self-explanatory and all variables correspond directly to the variables discussed earlier. The second table provides a breakdown of the contributing factors to your inflation forecast based on the relevant equations. For example, the core inflation is a function of the past value, inflation expectations real marginal costs including breakdown for domestic and external factors. Similarly, breakdown of food and oil prices is provided below. Next page provides decomposition of the output gap and information about the supply side assumptions. Decomposition of monetary policy and information about foreign variables follow on pages five and six.

The Model Code

For the model code see the file `model.model` and you may open it in the Matlabs editor typing the command `edit model.model` in the command window. The model code has four basic parts: endogenous variables, equation residuals, parameters, and equations:

!transition_variables The list of all endogenous (transition) variables.

!transition_shocks The list of all residuals defined in individual equations.

!parameters The list of parameters. Note that this list must be identical to the parameters defined within the `readmodel` program - any parameter change in `model.model` must be reflected in `readmodel`, and vice versa.

!transition_equations The list of equations defining the endogenous (transition) variables. The number of transition variables must equal the number of transition equations.

The program `readmodel` (as a Matlab file it has an extension `.m`) takes the code written in the `model.model`, assigns the parameters values, and solves the model for its reduced form. To modify model parameters and steady-state values you need to open the `readmodel` in the Matlab editor typing the command `edit readmodel` in the Matlab command window or using the File menu. The IRIS commands used for handling the model are as follows:

The first command is model:

```
m = model('model.mod','linear',true,'assign',p);
```

and it reads the text file `model.model` containing the model code and assigns the model parameters and trend values preset in the database `p` to the model. In addition it transforms the model for the matrix algebra. Transformed model is written in the object `m`. The option 'linear' is logical function. If your model is linear you assign the value true. This makes the transformations and calculations easier and saves time.

Second, the model is solved for its reduced form solve:

```
m = solve(m);
```

For the reduced-form solutiom the Blanchard-Kahn algorithm is used. The object `m` contains now the reduced form solution.

Finally, steady-state of the model is calculated using the command `sstate`. Exact quotation looks as follows:

```
m = sstate(m);
```

The command `sstate` takes the reduced-form solution of the model written in object `m` and calculates its steady-state. This is done via iterations using the Matlabs optimization toolbox. The steady-state values are written in object `mss` that may be viewed by typing the `mss` [enter] in the Matlab command window. It is useful to check the steady-state values any time you change the model structure (see section Calibration for further details).

Using the command `save` the object `m` is saved in the database file `model.mat`. This is used later for shock analysis and forecasting. You may run the `readmodel` simply by typing: `readmodel` [enter] in the Matlab command window. The evidence that the `readmodel` works properly is either the appearance of file `model.mat` in your current directory (if it was not there before) and you may see the list of steady-state values in the Matlab command window after typing `mss` in the Matlab command window once the `readmodel` has been run. You do not need run the `readmodel`, however, as the other programs call the `readmodel` as a function jointly with the model and its parameterization and steady-state as the outcome.

References

[Beneš et al., 2008] Beneš, Jaromír, Hurník, Jaromír and David, Vávra, 2008, 'Exchange Rate Management and Inflation Targeting: Modeling the Exchange Rate in Reduced-Form New Keynesian Models', *Czech Journal of Economics and Finance*, Vol. 3-4/2008.

[Berg et al., 2006] Berg, Andrew, Karam, Philippe and Douglas, Laxton, 2006, 'Practical Model-Based Monetary Policy Analysis: A How-To Guide', IMF Working Paper 06/81 (Washington: International Monetary Fund).