Solving and estimating stochastic models with block decomposition

Ferhat MIHOUBI[1]

1st September 2011

JEL codes: C63, C87
Key words: block decomposition, stochastic simulation, DSGE model, Algorithm

1.- Introduction

Dynamic Stochastic General Equilibrium (DSGE) models built in central Banks or in public institutions often contain several hundred equations. Their estimation using Bayesian methods is extremely expensive in CPU time. The computation of the posterior distribution mode and the posterior distribution using MCM algorithm generally requires at least several thousand evaluations of the kernel distribution. During the kernel evaluation the four following steps are performed:

- Solve the deterministic steady-state.
- Check the Blanchard and Kahn conditions.
- Compute the rational expectation solution of the model
- Compute the likelihood and the kernel distribution.

Even in the simplest configuration: a rational expectation solution computed using a first order perturbation method and an evaluation of the likelihood using a Kalman filter, these steps may require several hours of computing time.

This paper investigates the ways to reduce the computational time devoted to the simulation and the estimation of large scale DSGE models. Most of DSGE models have a block recursive structure. As an obvious example, AR(1) shocks can be solved independently of the remaining variables of the model. This recursive block structure is also met in models with nominal rigidities where the potential GDP is computed with the same model without nominal rigidities, or in multicountry models composed of a large country and of several small countries with no feedback effects from the small countries to the large one, or in the overlapping generation models without intergenerational altruism.

Several papers have examined the ways to exploit this block structure in order to improve deterministic simulations (Gilli and Pauletto (1998), van't Veer(2006) but few contributions have considered stochastic simulations or likelihood evaluation (Strid and Walentin (2009)).

[1] EPEE Department of Economic Evry-val d'Essonne University:
4 Boulevard F. Mitterrand 91025 Evry, France
and CEPREMAP Dynare team:
142 rue du Chevaleret 75013 Paris, France
ferhat.mihoubi@univ-evry.fr

This paper investigates the way to speed up stochastic simulation and estimation of the DSGE models using the block decomposition. More precisely, the gains of the block decomposition are considered in the four steps involved in the DSGE likelihood evaluation.

The first part of the paper addresses the question of block decomposition of DSGE models. The block decomposition method is carefully described considering the way to reduce the block size. The second part examines the gains involved by the block decomposition in the two first steps of the likelihood evaluation. A particular attention is given to the reduction step during the rational expectation solution. The last part describes the first order approximation of the block decomposed model. The last part described the implementation of the block Kalman filter proposed by Strid and Walentin (2009), in order to make the most of the block structure.

2.- The block decomposition

We consider the following rational expectation model:

$$E_t\left[f\left(y_{t+1}, y_t, y_{t-1}, u_t\right)\right] = 0$$

with f a system formed of n equations, y_t the endogenous variables and u_t an iid exogenous shocks with $E\left(u_t\right) = 0$ and $V\left(u_t\right) = \Omega$. This formulation is rather general since models with more than one lag or lead, could be rewritten into the previous form adding by auxiliary variables and their definition equations.

To describe the block decomposition of the model, we will consider its linearized form:

$$A\hat{y}_{t-1} + B\hat{y}_t + CE_t\left[\hat{y}_{t+1}\right] + HE_t\left[u_t\right] = 0 \quad (1)$$

with $\hat{y}_t$ the endogenous variables expressed in deviation to their steady-state values, $A = \frac{\partial f}{\partial y_{t+1}}$, $B = \frac{\partial f}{\partial y_t}$ and $C = \frac{\partial f}{\partial y_{t-1}}$ the Jacobian matrices with respect to future current and past endogenous variables, and with $H = \frac{\partial f}{\partial u_t}$ the Jacobian matrix with respect to exogenous shocks. All the Jacobian matrices are evaluated at the deterministic steady-state.

As we want to construct a recursive block structure where the variables of one block have no feedback effect on the variables of the previous blocks whatever the periods considered, we have to consider the block decomposition of the Jacobian matrix of the deterministic steady-state model:

$$D^* = A + B + C$$

The model has to be solved block by block, so each endogenous variable has to be unambiguously matched with an equation. This is a well-known problem of matching in a bipartite graph: a graph connecting two independent sets –equation and endogenous variables-. The augmenting path algorithm finds the maximum cardinality matching (ie the maximum number of equations and variables that could be matched) by starting with a first naïve matching and trying to improve this initial matching considering new matching based on unmatched vertices (equations or endogenous variables).

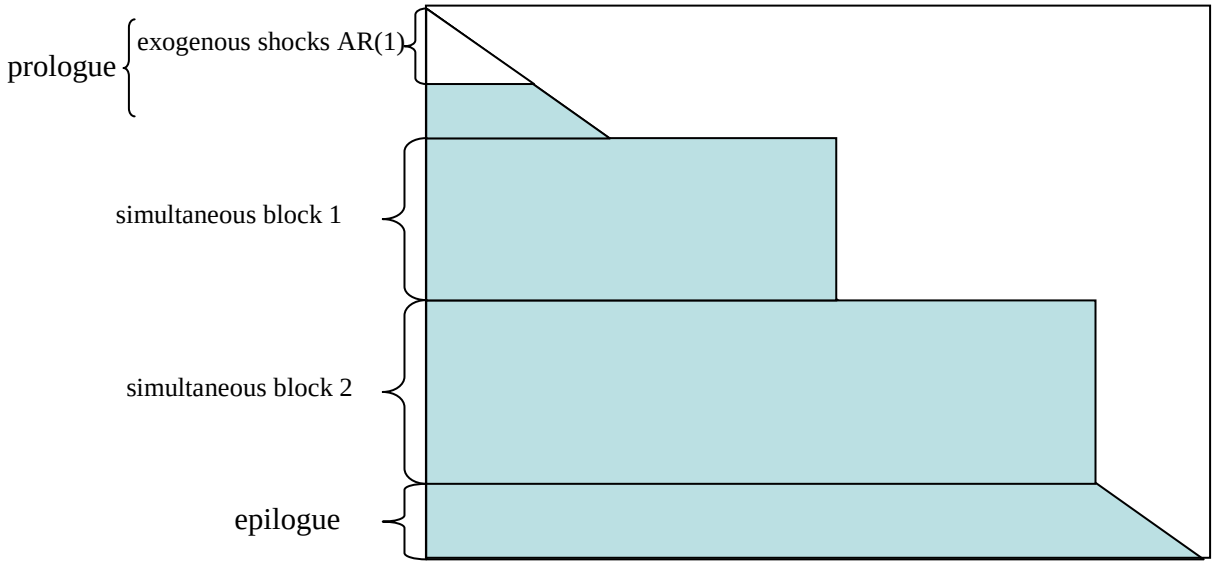
The matching process can produce a singular system, as soon as the variable y_j is matched with equation *i* and the element *i*,*j* normalized[2] Jaocbian matrix ($D_{i,j}^{*,N}$) is close to zero. To prevent this potential singularity, the augmenting path algorithm can be applied starting with a large cutoff applied to the Jaobian matrix (all the elements below the value of the cutoff are set to zero and the related edge in the bipartite graph are discarded) and reducing the cutoff until a perfect matching[3] is found.

Some parts of the medium/large scale models are purely recursive. The prologue is composed of the equations where the endogenous variable depends only on exogenous variables or on endogenous variables of the previous equations of the prologue. Typically, the shocks belong to the prologue. The prologue forms the first block of the reordered model and is low triangular. The epilogue contains all the endogenous variables that are pure output (the endogenous variables that do not appear in any equation except in the following equations of the epilogue). The epilogue has a low triangular shape.

The remaining equations and variables could be split in several simultaneous sub blocks. For example, in a model with nominal rigidities, the computation of the potential GDP requires to add the same model without nominal rigidities. In this case the overall model could be split in a block recursive structure, the first block containing the model without nominal rigidities and the second the model with nominal rigidities. The same remark applied to a two-country model composed of a large and a small country, if there is no feedback effect from the small country to the large one: the first block contains the large country model and the second the small country model.
The smallest simultaneous blocks correspond to the strong component (i.e. a sub graph where all the vertices could be joined from any other) of an oriented graph. This graph is a representation of the model structure where each vertex represents an equation. If the endogenous variable associated to the equation *i* appears in equation *j* an arc from vertex *i* to vertex *j* is added to the graph.
Several algorithms could be used to find the strong components of the graph formed by the remaining equations: Tarjan algorithm or Gabow algorithm (Cormen and ali(2001))
Figure 1 presents the general shape of the incidence matrix[4] related to the reordered *D** matrix once the model has been block-decomposed.

[2] The normalized Jacobian matrix is obtained dividing each element by the sum of the absolute value of all the elements of its row.
[3] A matching where all the vertices are matched.
[4] The incidence matrix is a Boolean matrix describing the model structure. If e variable *j* enter the equation *i* (or if row *i* and column *j* of the Jacobian matrix is non null), the element located at row *i* and column *j* of the incidence matrix is set to 1.

Figure 1: general form of the steady-state incidence matrix of the block decomposed model



3.- The computation of the steady-state and Blanchard and Kahn's conditions on a block decomposed model

The first benefit of the block decomposition lies in the speed-up of the computation of steady-state solution. For the following nonlinear system

$$f\left(\overline{y},\overline{y},\overline{y},0\right)=0$$

the steady-state solution can be computed using a Newton algorithm, where at each iteration the following linear system is solved:

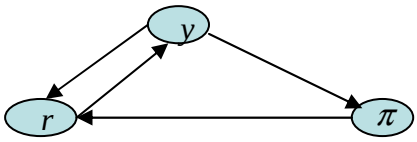
$$D^{*}\overline{y}=b$$

Instead of solving the overall system composed of n equations which has a time complexity of $O(n^3)$, with a block decomposed model, the prologue and the epilogue equations have to be evaluated or solved equation per equation and each simultaneous blocks have to be solved. If we note n^p, n^e and n^{b_i} the size of the prologue, the epilogue and the b simultaneous blocks ($i=1,\ldots,b$), the time complexity order is now at most $\left(n^p+n^e\right)O(1)+\sum_{i=1}^{b}O\left(n_{b_i}^3\right)\leq O(n^3)$.

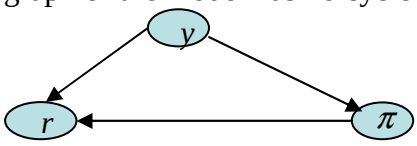
In addition the size of each block could be reduced using the feedback variables. In a simultaneous linear system, once the feedback variables are known, the system becomes recursive. The remaining variables, called recursive variables, are computed recursively and could be considered as temporary variables. The number of variables involved in the system to solve reduces to the number of feedback variables. Consider the following example:

$$\begin{cases} y_t=\alpha_1 r_t+\varepsilon_t^g \\ \pi_t=\varphi y_t+\varepsilon_t^\pi \\ r_t=\mu\pi_t+\tau y_t \end{cases}$$

The graph associated to this model is:



Several feedback sets could be considered: $\{y\},\{r\},\{y,\pi\},\{r,\pi\},\{y,r\}$. But only two correspond to the minimal feedback sets: $\{y\},\{r\}$. Hence r or y are feedback variables. If y is known the system becomes recursive (π first and then r are simply evaluated) and the graph of the model has no cycle any more:



To find one of the minimum feedback sets several heuristic methods are available (Gaurdabassi (1974) and Cheung and Kuh (1974)) and they are based on the simplification of model's graph to get an acyclic graph.

Using feedback variable, the maximum time complexity order of the linear systems to solve is

$$\left(n^p + n^e\right)O(1) + \sum_{i=1}^{b} O\left(\left(n_{b_i}^{fv}\right)^3\right) \le O(n^3)$$

with $n_{b_i}^{fv}\left(\le n_{b_i}\right)$ the number of feedback variables in block *i*.

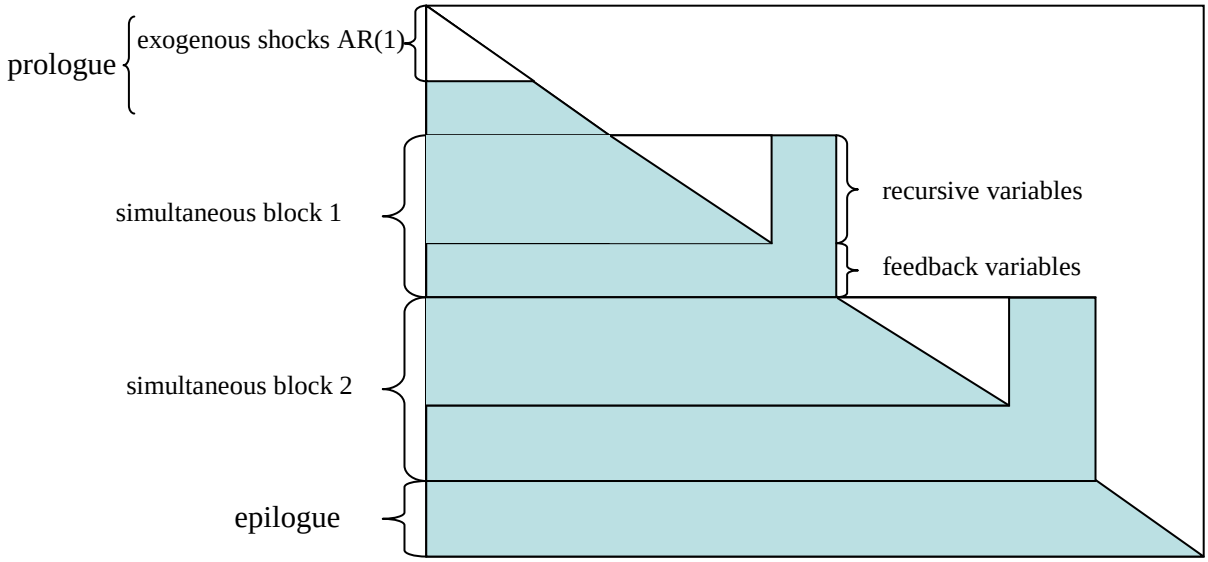
The Jocabian matrix of the reduced simultaneous block *i* expressed only with respect to feedback variables, is:

$$D_{b_i}^{*,fv} = D_{b_i}^{*,fv,fv} + D_{b_i}^{*,fv,r} D_{b_i}^{*,r,fv}$$

with $D_{b_i}^{*,fv,fv}$ the $\left(n_{b_i}^{fv}, n_{b_i}^{fv}\right)$ matrix containing the derivatives of the feedback variables with respect to the feedback variables, $D_{b_i}^{*,fv,r}$ the $\left(n_{b_i}^{fv}, n_{b_i} - n_{b_i}^{fv}\right)$ matrix containing the derivatives of the feedback variables with respect to the recursive variables and $D_{b_i}^{*,r,fv}$ the $\left(n_{b_i} - n_{b_i}^{fv}, n_{b_i}^{fv}\right)$ triangular matrix containing the derivatives of the recursive variable with respect to the feedback variables computed recursively using the triangular sub-matrix.

Figure 2 presents the general form of the incidence matrix after the determination of the feedback set for each simultaneous block. During the steady-state computation, these blocks have to be solved only for the feedback variables, the recursive variables are only a by product.

Figure 2: general form of the steady-state incidence matrix of the block decomposed model with feedback variables



To check the Blanchard and Kahn's conditions, the endogenous variables have to be split in three sets and the Jacobian matrix B of the linear approximation (1) in three components:

$$A^1\hat{y}_{t-1}^1+\left(B^0|B^1|B^2\right)\left[\hat{y}_t^0\hat{y}_t^1\hat{y}_t^2\right]+C^2E_t\left[\hat{y}_{t+1}^2\right]=0 \quad (2)$$

with $\hat{y}_t^0$ the n_s purely static variables, $\hat{y}_t^1$ the n_1 predetermined variables (appearing in contemporary and lagged form) and $\hat{y}_t^2$ the n_2 anticipated variables (appearing in contemporary and leaded form) and B^0, B^1 and B^2 their respective Jacobian Matrices.

The system could be reduced by eliminating the purely static variables $\hat{y}_t^0$. To do it, a QR decomposition of the B^0 matrix is performed:

$$B^0=QR$$

with Q an orthogonal matrix and R an upper triangular matrix.
The reduced system becomes:

$$\tilde{A}^1\hat{y}_{t-1}^1+\left(\tilde{B}^1|\tilde{B}^2\right)\left[\hat{y}_t^1\hat{y}_t^2\right]+\tilde{C}^2E_t\left[\hat{y}_{t+1}^2\right]=0 \quad (3)$$

with $\tilde{A}^1$ the n_1+n_2 last rows of the matrix $Q'A^1$, $\tilde{B}^1$ the n_1+n_2 last rows of the matrix $Q'B^1$, $\tilde{B}^2$ the n_1+n_2 last rows of the matrix $Q'B^2$ and $\tilde{C}^2$ the n_1+n_2 last rows of the matrix $Q'C^2$.

It is worth noting that a part of this reduction step could be performed using the feedback variables computed under the constraint that all the dynamic variables have to belong to the set of feedback variables[5]. In this case, a part of the static variables correspond to recursive variables and the QR decomposition has to be applied to the smaller set of the static variables belonging to the feedback set.

Because some of dynamic variables appear both with leads and lags in the model, the dynamic system could be rewritten as:

[5] This constraint prevents to increase the dynamic dimension of the system/.

$$\begin{bmatrix} \tilde{C}^2 & \tilde{B}^1 \\ 0 & I^1 \end{bmatrix} \begin{bmatrix} E_t\left[\hat{y}_{t+1}^2\right] \\ \hat{y}_t^1 \end{bmatrix} = \begin{bmatrix} -\tilde{B}^2 & -\tilde{A}^1 \\ I^2 & 0 \end{bmatrix} \begin{bmatrix} \hat{y}_t^2 \\ \hat{y}_{t-1}^1 \end{bmatrix}$$

$$\Leftrightarrow F \begin{bmatrix} E_t\left[\hat{y}_{t+1}^2\right] \\ \hat{y}_t^1 \end{bmatrix} = G \begin{bmatrix} \hat{y}_t^2 \\ \hat{y}_{t-1}^1 \end{bmatrix} \qquad (4)$$

with I^1 a selector matrix indicating the position of the mixed variables (variables appearing with both leads and lags in the model) in the matrix $\tilde{B}^1$ and I^2 the same selector matrix for $\tilde{B}^2$.

The generalized eigenvalues of this dynamic system are computed using a Schur decomposition of the pencil (*F*, *G*).

This procedure has to be applied only for the simultaneous blocks. For the other dynamics blocks purely recursive, the eigenvalues are straightforward computed using the normalized diagonal terms of the dynamic Jacobian matrix.

Because the order of complexity of the Schur decomposition is at least $O(n^3)$, the block-decomposition of the model reduces here also the computational cost for a medium/large scale model.

In addition, the block decomposition could also be helpful to locate the sources of instability since it reduces the number of variables/equations involved in the instability to those belonging to the block.

To evaluate the gains coming from the block decomposition, we compute the steady state and the Blanchard and Kahn's conditions for the three large scale models: Eagle (Gomez, Jacquinot and Pisani(2010)), Gimf (Kumhof and ali(2010)) and an overlapping generation model.

Table 1: Computation time of the steady-state and the Blanchard and Kahn's conditions with and without block decomposition[6]

		Without block decomposition	With block decomposition
Eagle	Size of the biggest block[7]	965	560
	Blanchard and Kahn conditions (seconds)	23	7.5
	Steady state (seconds)	0.10	0.08
GIMF	Size of the biggest block	2032	903
	Blanchard and Kahn conditions (seconds)	258	45
	Steady state (seconds)	0.11	0.10
Overlapping generations model	Size of the biggest block	1086	441
	Blanchard and Kahn conditions (seconds)	91	22
	Steady state (seconds)	0.27	0.13

[6] These simulations are performed on a Intel double Core T9300 with 4Go RAM using Dynare 4.2 with "bytecode" option in model command for the model without block decomposition and with options "block" and "bytecode" for the block-decomposed model. The steady-state is computed using a sparse LU decomposition (Dynare option solve_algo=5).

Table 1 reports the computational time reduction involved by the block decomposition. The computation time required to solve the steady-state model, is reduced at least by 10% for the smallest model, and by a factor 3 for the biggest one (GIMF). The CPU time cost to compute the Blanchard and Kahn condition, is also strongly reduced by at least a factor of 3 with Eagle and by a factor 5.5 with GIMF.

4.- **The first order perturbation method applied to a block-decomposed model**

In the traditional case the model to solve is:

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

with u_t an iid exogenous shocks satisfying $E(u_t) = 0$ and $V(u_t) = \Omega$.

We want to compute the rational expectation (RE) solution which has the following form:

$y_t = g(y_{t-1}, u_t)$

In case of a block decomposed model, we have for a block *b*:

$E_t[f_b(y_{t+1}, y_t, y_{t-1}, u_t, x_{t+1}, x_t, x_{t-1})] = 0$ (5)

with x_t the endogenous variables from the previous blocks and y_t the endogenous variables of the current block.

The RE solution of the previous blocks is supposed to be computed and for the endogenous variables determined in the previous and appearing in the block *b*:

$x_t = l_b(x_{t-1}, u_t)$

We want to compute the RE solution of the current block:

$y_t = g_b(y_{t-1}, u_t, x_{t-1})$

To do so, we first compute the following function, where the expected values of y and x are replaced by their RE solutions:

$$F_b(y_{t-1}, u_t, u_{t+1}, x_t, x_{t-1})$$
$$= f_b\left(g_b\left(g_b(y_{t-1}, u_t, x_{t-1}), u_{t+1}, l_b(x_{t-1}, u_t)\right), g_b(y_{t-1}, u_t, x_{t-1}), y_{t-1}, u_t, l_b\left(l_b(x_{t-1}, u_t), u_{t+1}\right), l_b(x_{t-1}, u_t), x_{t-1}\right)$$

Thus (5) can be rewritten as:

$E_t[F_b(y_{t-1}, u_t, u_{t-1}, x_t, x_{t-1})] = 0$

We want to linearize the model around a deterministic steady-state defined by:

$f_b(\bar{y}, \bar{y}, \bar{y}, 0, \bar{x}, \bar{x}, \bar{x}) = 0$

$\bar{x} = l_b(\bar{x}, 0)$

$\bar{y} = g_b(\bar{y}, 0, \bar{x})$

The first order expansion of (5) is

$$
\begin{aligned}
E_t\left[F_b\left(\mathrm{y}_{t-1},\mathrm{u}_t,\mathrm{u}_{t+1},\mathrm{x}_t,\mathrm{x}_{t-1}\right)\right] \approx E_t\Big[\, & f_b\left(\bar{\mathrm{y}},\bar{\mathrm{y}},\bar{\mathrm{y}},0,\bar{\mathrm{x}},\bar{\mathrm{x}},\bar{\mathrm{x}}\right) \\
& + A\Big(g_y\left(g_y\hat{y}_{t-1}+g_x\hat{x}_{t-1}+g_u u_t\right)+g_u u_{t+1}+g_x\left(l_x\hat{x}_{t-1}+l_u u_t\right)\Big) \\
& + B\left(g_y\hat{y}_{t-1}+g_x\hat{x}_{t-1}+g_u u_t\right)+C\hat{y}_{t-1}+f_u u_t \;+ f_{x_{t+1}}\left(l_x\left(l_x\hat{x}_{t-1}+l_u u_t\right)+l_u u_{t+1}\right) \\
& + f_{x_t}\left(l_x\hat{x}_{t-1}+l_u u_t\right)+f_{x_{t-1}}l_{sx}\hat{x}_{t-1}\Big] \\
= E_t\Big[\, & f_b\left(\bar{\mathrm{y}},\bar{\mathrm{y}},\bar{\mathrm{y}},0,\bar{\mathrm{x}},\bar{\mathrm{x}},\bar{\mathrm{x}}\right) \\
& + \left(Ag_y g_y + Bg_y + C\right)\hat{y}_{t-1} \\
& + \left(Ag_y g_u + Ag_x l_u + Bg_u + f_u + f_{x_{t+1}}l_x l_u + f_{x_t}l_u\right)u_t \\
& + \left(A\left(g_u\right)+ f_{x_{t+1}}l_u\right)u_{t+1} \\
& + \left(A\left(g_y g_x + g_x l_x\right)+Bg_x + f_{x_{t+1}}l_x l_x + f_{x_t}l_x + f_{x_{t-1}}l_{sx}\right)\hat{x}_{t-1}\Big] \\
= f\left(\bar{\mathrm{y}},\bar{\mathrm{y}},\bar{\mathrm{y}},0,\bar{\mathrm{x}},\bar{\mathrm{x}},\bar{\mathrm{x}}\right) & \\
& + \left(Ag_y g_y + Bg_y + C\right)\hat{y}_{t-1} \\
& + \left[Ag_y g_u + Bg_u + f_u + f_{x_{t+1}}\left(l_x l_u\right)+ f_{x_t}l_u + Ag_x l_u\right]u_t \\
& + \left[A\left(g_y g_x + g_x l_x\right)+Bg_x +\left(f_{x_t} + f_{x_{t+1}}l_x\right)l_x + f_{x_{t-1}}l_{sx}\right]\hat{x}_{t-1}
\end{aligned}
$$

The new terms implied by the block decomposition are reported in red colour in the last equation.

with $A=\dfrac{\partial f_b\left(\bar{\mathrm{y}},\bar{\mathrm{y}},\bar{\mathrm{y}},0,\bar{\mathrm{x}},\bar{\mathrm{x}},\bar{\mathrm{x}}\right)}{\partial y_{t+1}}$, $B=\dfrac{\partial f_b\left(\bar{\mathrm{y}},\bar{\mathrm{y}},\bar{\mathrm{y}},0,\bar{\mathrm{x}},\bar{\mathrm{x}},\bar{\mathrm{x}}\right)}{\partial y_t}$,

$C=\dfrac{\partial f_b\left(\bar{\mathrm{y}},\bar{\mathrm{y}},\bar{\mathrm{y}},0,\bar{\mathrm{x}},\bar{\mathrm{x}},\bar{\mathrm{x}}\right)}{\partial y_{t-1}}$, $f_u=\dfrac{\partial f_b\left(\bar{\mathrm{y}},\bar{\mathrm{y}},\bar{\mathrm{y}},0,\bar{\mathrm{x}},\bar{\mathrm{x}},\bar{\mathrm{x}}\right)}{\partial u_t}$, $f_{x_{t+1}}=\dfrac{\partial f_b\left(\bar{\mathrm{y}},\bar{\mathrm{y}},\bar{\mathrm{y}},0,\bar{\mathrm{x}},\bar{\mathrm{x}},\bar{\mathrm{x}}\right)}{\partial x_{t+1}}$,

$f_{x_t}=\dfrac{\partial f_b\left(\bar{\mathrm{y}},\bar{\mathrm{y}},\bar{\mathrm{y}},0,\bar{\mathrm{x}},\bar{\mathrm{x}},\bar{\mathrm{x}}\right)}{\partial x_t}$, $f_{x_{t-1}}=\dfrac{\partial f_b\left(\bar{\mathrm{y}},\bar{\mathrm{y}},\bar{\mathrm{y}},0,\bar{\mathrm{x}},\bar{\mathrm{x}},\bar{\mathrm{x}}\right)}{\partial x_{t-1}}$, $l_x=\dfrac{\partial l_b\left(\bar{x},0\right)}{\partial x_{t-1}}$, $l_u=\dfrac{\partial l_b\left(\bar{x},0\right)}{\partial u_t}$,

$g_u=\dfrac{\partial g_b\left(\bar{y},0,\bar{x}\right)}{\partial u_t}$, $g_y=\dfrac{\partial g_b\left(\bar{y},0,\bar{x}\right)}{\partial y_{t-1}}$ and $g_x=\dfrac{\partial g_b\left(\bar{y},0,\bar{x}\right)}{\partial x_{t-1}}$

and l_{sx} a selecting matrix with a rows number equal to the number of state variables and a number of columns equal to the number of endogenous variable belonging to the previous blocks and appearing in the current block.

The RE solution imposes the three following conditions:

$$
\begin{cases}
Ag_y g_y + Bg_y + C = 0 & (6) \\
A\left(g_y g_x + g_x l_x\right)+Bg_x +\left(f_{x_t} + f_{x_{t+1}}l_x\right)l_x + f_{x_{t-1}} = 0 & (7) \\
Ag_y g_u + Bg_u + f_u + f_{x_{t+1}}\left(l_x l_u\right)+\left(f_{x_t} + Ag_x\right)l_u = 0 & (8)
\end{cases}
$$

As in the traditional case, g_y is recovered from the equation (6). Using a Schur decomposition of the pencil (F,G) and excluding the explosive trajectories we get the RE solution in g_y (Collard and Juillard(2001)).

Knowing g_y, we get g_x from equation (7) which is a Sylvester equation:

$$\left[Ag_y + B\right]g_x + Ag_x l_x = -\left[\left(f_{x_t} + f_{x_{t+1}} l_x\right)l_x + f_{x_{t-1}}\right]$$

We know that a unique solution exists for this Sylvester equation if $\left[Ag_y + B\right]$ and $-l_x$ have no common eigenvalues. g_x is then the solution of:

$$\left\{I \otimes \left[Ag_y + B\right] + l_x^{'} \otimes A\right\} vec\left(g_x\right) = -vec\left(\left(f_{x_t} + f_{x_{t+1}} l_x\right)l_x + f_{x_{t-1}}\right)$$

This equation could also be numerically solved using the Bartels-Stewart algorithm.

Finally, knowing g_y and g_x, we recover directly g_u from equation (8):

$$g_u = -\left[Ag_y + B\right]^{-1}\left[f_u + f_{x_{t+1}}\left(l_x l_u\right) + \left(f_{x_t} + Ag_x\right)l_u\right]$$

The RE solution of the complete model $z_t = l\left(z_{t-1}, u_t\right)$ is updated by stacking the new solution: $z_t = \begin{bmatrix} z_t \\ y_t \end{bmatrix} = l(z_{t-1}, u_t) = \begin{bmatrix} l_z z_{t-1} + l_z u_t \\ \begin{bmatrix} l_x & g_y \end{bmatrix}\begin{bmatrix} x_{t-1} \\ y_{t-1} \end{bmatrix} + g_u u_t \end{bmatrix}$

For blocks containing only leads or only lags variables, the RE solution is much simpler. So one of the advantages of the block decomposition applied to the computation of RE solution, is to limit the use of generalized Schur decomposition only to the blocks containing both lead and lags variables.

The gain related to the block decomposition is weaker than the complexity reduction for the Blanchard and Kahn conditions, because of the additional specific cost induced by the computation of g_x for a block decomposed model. However, the implementation of the first order approximation on a block decomposed model remains useful as soon as the evaluation of the likelihood using a Kalman filter takes advantage of this block triangular form of the RE solution.

5.- Block Kalman filter

The general idea of Block Kalman Filter (Strid and Walentin 2009) is to take advantage of the block structure of the model to reduce the number of operations involved in the likelihood evaluation with a Kalman filter.

Consider the state-space representation of a DSGE:

$\mathrm{Y}_t = d\left(\theta\right) + ZX_t + v_t$ measurement equation

$X_t = c + T\left(\theta\right)X_{t-1} + R\left(\theta\right)\varepsilon_t$ state equation

with Y_t the measurement vector (Nx1), X_t the state vector (mx1), ε_t the innovation vector (gx1) ($\varepsilon_t \sim N(0,Q)$) , v_t the measurement error (Nx1) ($v_t \sim N(0,H)$) and θ the structural parameters of the model.
In our case $T(\theta) = g_y = l_x$ has a block triangular form and $R(\theta) = g_u$.

The traditional Kalman Filter (without block decomposition) is composed of the following two steps:

- The updating equations:

$$X_{t|t} = E\left[X_t \middle| Y_1,\ldots,Y_t\right] = X_{t|t-1} + K_t F_t^{-1}\left(Y_t - ZX_{t|t-1} - d\right)$$

$$P_{t|t} = P_{t|t-1} - K_t F_t^{-1} K_t^{'}$$

with $K_t = P_{t|t-1}Z'$ and $F_t = V\left[Y_t \middle| Y_1,\ldots,Y_{t-1}\right] = ZK_t + H$

- The prediction equations

$$X_{t+1|t} = TX_{t|t} + c$$

$$P_{t+1|t} = TP_{t|t}T' + RQR'$$

The likelihood is simply: $\log L(\theta) = -\frac{NT}{2}\log(2\pi) - \frac{1}{2}\sum_{t=1}^{T}\log\left(\left|F_t\right|\right) - \frac{1}{2}\sum_{t=1}^{T}\hat{v}_t^{'}F_t^{-1}\hat{v}_t$ with $\hat{v}_t = Y_t - ZX_{t|t-1}$.

The most time consuming step of the Kalman filter lies in the computation of the conditional variance of the prediction error $P_{t+1|t}$. The structure of the transition matrix, which depends on the block structure of the model, could be used to reduce the computational burden.

Instead of splitting by blocks all the equations of the Kalman filter as in Strid and Walentin (2009), we focus on the most time consuming step: the computation of the conditional variance of the prediction error $P_{t+1|t}$. The last term RQR' is time-invariant and is computed once for all at the beginning of the filter. Most of the effort is devoted on the first term: $TP_{t|t}T'$.
This product has two features:
- The transition matrix has a block recursive form as described in figure 1. Only the non-zero terms of the matrix T (the lower block triangular terms) have to be considered in the product. It should be noted that this approach differs from that in which a sparse representation of T is considered. Since the block triangular form is constant throughout the estimation process, the block structure could be hard coded in the matrix multiplication. This feature is exploited in both product ($Tmp = TP_{t|t}$ and $tmp.T'$)
- The overall product produces a symmetric matrix. This feature is taken into account in the second product $tmp.T'$.

Those two features are incorporated in a mex-file where matrix product is implemented as described in figure 3:

Figure 3: computation of $TP_{t|t}T'$ (`first_nz` and `last_nz` are index vectors indicating for each row of the transition matrix respectively the first and the last nonzero element)

```
do i=1 to n;
  do j=1 to n;
     do k=first_nz[i] to last_nz[i]
          tmp[i , j] = tmp[i , j] + T[i , k] * Ptt[k , j];
     endo k;
   endo j;
endo i;
do i=1 to n;
  do j=i to n;
     do k=first_nz[j] to last_nz[j]
          P[i , j] = P[i , j] + tmp[i , k] * T[j , k];
     endo k;
     P[j , i] = P[i , j];
   endo j;
endo i;
```

For the AR(1) shocks located in the first part of the prologue, the index vectors are such that `first_nz[i] = last_nz = i.` In this case, the element i,j of the conditional variance of the prediction error become simply: $\left[P_{t+1|t}\right]_{i,j} = \left[P_{t|t}\right]_{i,j} T_{i,i}T_{j,j}$.

The reduction of the computing burden when using block Kalman filter comes in this case from the diagonal transition matrix in the prologue part of the state-space representation and from the symmetry of the covariance matrix. Strid and Walentin have evaluated the computational time reduction using a block Kalman filter. They show that with large scale models, the CPU time decreases with the number of blocks considered and the computational time is reduced by a factor of 1.5 for a four block Kalman Filter.

To evaluate the CPU time reduction involved by the computation of the RE solution of the block decomposed model and the use of a block Kalman filter, the complete likelihood of the Smets and Wouters (2007) medium scale model has been evaluated one thousand times.

Table 2: one thousand evaluations likelihood of the Smets and Wouters(2007) model

	Without block decomposition	With block decomposition and kalman filter	With block decomposition and block-kalman filter
Size of the biggest block	58	26	26
Number of blocks	1	12	12
CPU-time (seconds)	27	25	20

With a medium scale model, the overall CPU time gain induced by the block decomposition is at most 25%. The two last columns of table 2 indicate that most of the CPU-time reduction is related to the block-Kalman filter (the CPU time is reduced by 20%). This result is in line with Strid and Walentin (2009) with an algorithm gain of 50% for a large scale model.

6-. Conclusion

This paper investigates the reduction of computational time induced by the block decomposition in all the steps of the stochastic simulation and the estimation of the DSGE models. For large scale models, the block decomposition can strongly reduce the time required to compute the steady-state and the Blanchard and Kahn's conditions. The reduction of the computational time seems to be modest when we consider the RE solution and the evaluation of the likelihood using a block-Kalman Filter. However, the gains using block decomposition seem to be much more promising when global methods are used to compute the RE solution. The block decomposition could be view as one solution to curse of dimension problem faced with global method on medium/large scale model.

References

- Cheung, L.K. and Kuh, E.S.(1974): The Bordered Triangular Matrix and Minimum Essential Sets of a Digraph", *IEEE Trans. On Circuits and Systems*, vol CAS-21, 5.
- Collard, F.and Juillard, M. (2001): A Higher-Order Taylor Expansion Approach to Simulation of Stochastic Forward-Looking Models with an Application to a Nonlinear Phillips Curve Model, *Computational Economics*, 17, 125-139.
- Cormen, T., Leiserson, C., Rivest R., and Stein (2001): C.. *Introduction to Algorithms*, Second Edition. MIT Press and McGraw-Hill.
- Gilli, M. and Pauleto, G. (1997): Sparse Direct Methods for Model Simulation, *Journal of Economic Dynamics and Control*, 21, 1093-1111.
- Gomez, S., Jacquinot, P. and Pisani, M. (2010) : The Eagle a Model for Policy Analysis of Macroeconomic Interdependence in the Euro Area, *ECB Working Paper*, 1195.
- Guardabassi, G. (1974): A note on Mininimal Essential Sets", *IEEE Trans. on Circuits and Systems*, vol CT-18, 5.
- Kumhof, M., Laxton, D., Muir, D., and Mursula S. (2010): The Global Integrated Monetary and Fiscal Model (GIMF) – Theoretical Structure, IMF Working Paper, 10/34.
- van't Veer,O. (2006): Solving large scale normalised rational expectation models, *CPB Discussion Paper*, 54.
- Smets F. and Wouters R. (2007): Shocks and Frictions in US Business Cycles: A Bayesian DSGE Approach, *American Economic Review*, vol. 97(3), 586-606.
- Strid, I. and Walentin, K. (2009): Block Kalman Filtering for Large-Scale DSGE Models, *Computational Economics*, 33, 277-304.
- West, D. B. (1999): *Introduction to Graph Theory* (2nd ed.), Prentice Hall, Chapter 3