

Contents lists available at [ScienceDirect](https://www.sciencedirect.com)

# Economic Modelling

journal homepage: [www.journals.elsevier.com/economic-modelling](http://www.journals.elsevier.com/economic-modelling)

## The ‘curse of dimensionality’ resolved: The effects of climate change and trade barriers in large dimensional modelling

Tom Kompas <sup>a,b,\*</sup>, Pham Van Ha <sup>c</sup>

<sup>a</sup> Centre of Excellence for Biosecurity Risk Analysis, School of Biosciences and School of Ecosystem and Forest Sciences, University of Melbourne, Melbourne, Australia

<sup>b</sup> Australia Crawford School of Public Policy, Australian National University, Canberra, Australia

<sup>c</sup> Crawford School of Public Policy, Australian National University, Canberra, Australia

### ARTICLE INFO

#### Keywords:

Climate change  
Trade barriers  
Large-scale intertemporal CGE models  
GTAP

### ABSTRACT

Solving large scale optimisation problems over space and time quickly generates a computational impasse, termed the ‘curse of dimensionality’. This severely limits the practical use of economic models, especially for determining the effects of climate change and protectionist trade policies. In this paper, we employ an innovative approach to solving (otherwise unsolvable) large scale systems through the use of parallel processing methods and a proper ordering of variables and equations in a ‘Nested Doubly Bordered Block Diagonal’ form. We illustrate how the approach can be used to solve an intertemporal CGE model with more than 500 million equations. Using existing damage functions, the framework allows us to determine the impact of climate change on long-run economic growth for 112 countries as a result of the effect of sea-level rise on land endowments, variation in crop yields and productivity and shifts in the demand for energy and transportation. We also compare our solution to more common (and smaller dimensional) recursive methods, in terms of both the economic effects of climate change and potential increases in trade barriers, showing the power and efficiency of our computational approach and parallel processing routine.

### 1. Introduction

It is not unusual to find numerical simulations that are dimensionally very large, especially in studies that examine potential distortions in international trade and the economic impacts from climate change. However, in economics, large scale models cause difficulties due to the need to incorporate optimal behaviour and optimising routines. This is especially the case in Computable General Equilibrium (CGE) models. Solving such large scale optimisation problems over space and time quickly generates a computational impasse, termed the ‘curse of dimensionality’. The scale of the optimising model is simply too large to solve, and this alone severely limits the practical use of many economic models.

In this paper, we employ an innovative approach to solving large scale systems through an ordering of variables and equations in a ‘Nested Doubly Bordered Block Diagonal’ (NDBBD) form. The NDBBD approach allows for a highly efficient manipulation of a matrix of first-order terms, saving computational space and allowing for substantial

increases in dimension. As such, we extend [Ha and Kompas \(2016\)](#) and [Ha et al. \(2017\)](#), more regional and country-specific trade models, to solve a very large intertemporal and regional CGE model, showing how the first-order differential matrix of intertemporal regional CGE models can be transformed into a NDBBD matrix. We combine this matrix reordering with parallel processing techniques to facilitate computation and illustrate how the approach can be used to solve an intertemporal CGE model with more than 500 million equations. Using existing damage functions, the framework allows us to determine the impact of climate change on long run changes in GDP for 112 countries. We also compare our solution to more common (non-optimal and smaller dimensional) recursive methods in terms of both the economic effects of climate change and potential increase in trade barriers, showing the power and efficiency of our computational approach and parallel processing routine.

The background on CGE modelling is fairly well known. During the last 60 years, from the pioneering work of [Johansen’s \(1960\)](#) simple model with 22 production sectors and one representative consumer,

\* Corresponding author. Centre of Excellence for Biosecurity Risk Analysis, School of Biosciences and School of Ecosystem and Forest Sciences, University of Melbourne, Melbourne, 3010, Australia.

E-mail addresses: [tom.kompas@unimelb.edu.au](mailto:tom.kompas@unimelb.edu.au) (T. Kompas), [ha.pham@anu.edu.au](mailto:ha.pham@anu.edu.au) (P. Van Ha).

<https://doi.org/10.1016/j.econmod.2018.08.011>

Received 9 July 2017; Received in revised form 5 July 2018; Accepted 29 August 2018

Available online 5 September 2018

0264-9993/© 2018 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

CGE models have grown in size and complexity. This was especially the case after the introduction of the System of National Accounts in 1953 (United Nations, 1953), and its latest version in 2008 (EC et al., 2009), by the United Nations, providing an extensive input-output table database for many countries. These input-output databases have enabled the construction of globally integrated economic and input-output databases for trade analysis (Narayanan and Walmsley, 2008) and global CGE models such as GTAP (Hertel, 1997), or the Global Trade and Analysis Project, and its intertemporal version GTAP-INT (Ha et al., 2017). The size of such intertemporal regional CGE databases poses a challenge to computational capacity, severely limiting computational routines that use optimising frameworks.

Even modern software packages such as GEMPACK or GAMS, which use a serial direct LU solver from HSL (2013) or LUSOL, a sparse LU solver for system of linear equations (Saunders et al., 2018), become incapable of solving intertemporal CGE models of even a modest size. For those models, for example, with over 100 industries, it is only possible to solve the system simultaneously over a small number of time periods (roughly 5) by the conventional LU decomposition method (i.e., the so-called ‘lower-upper’ method for factoring a matrix as the product of a lower triangular matrix and an upper triangular matrix) (Dixon et al., 2005).

Recursive methods have thus been employed instead as a second best solution to overcome this computational challenge. In this setting, using adaptive or static price level expectations, the expected future values of variables are specified as functions of current or past values (Dixon et al., 2005). The model can, therefore, be solved forward one period at a time, making the dimension more manageable. Nevertheless, even in this setting, the computational burden is not negligible as the number of time periods increases. Furthermore, with static or adaptive expectations, agents fail to react to announced policy changes, with known or projected future impacts, until they actually occur. This is clearly dynamically inefficient.

Dixon et al. (2005) subsequently introduced a rational expectations approach to repair the deficiency of static or adaptive expectations modelling. Dixon et al. (2005) applies an iterative solution method (Fair, 1979; Fair and Taylor, 1983) by ‘guessing’ all the future values of expected variables, solving forward, and adjusting the values for these variables after each iteration until convergence. With proper expectational rules, the rational expectation solution method may converge to a full intertemporal solution and, therefore, can be used to analyse announced policy changes in advance, at least in principal. However, in practice, this rational expectations approach has been used to solve dynamic CGE models for only a handful of periods. Its numerical stability is also in doubt since the studied CGE models do not always have steady state outcomes Dixon et al. (2005). In short, the iterative procedure is a relatively cumbersome process and does not always guarantee convergence.

Another major attempt to solve full scale intertemporal CGE models uses a so-called MSG (McKibbin-Sachs Global) algorithm (McKibbin, 1987; McKibbin and Sachs, 1991). This solution method solves the model in first-order Taylor expansion form by writing an approximation around a known solution path. The model is then condensed into a system of linear differential equations by substituting out all other endogenous variables. The system of linear differential equations is solved using a special block diagonal structure where every equation will have two consecutive values of flow or stock variables. Therefore, at some time  $T$  (large enough for the model to converge to a steady state), the flow variables will be solved backward one period at a time in terms of current stock variables, all future exogenous variables and the terminal value of the flow variables. When reaching period zero, the system can be solved forward again to obtain solutions for flow and stock variables for all future periods.

Although the MSG algorithm has a clear advantage over the traditional LU decomposition method, its weakness is its use of a first-order approximation and although much larger in dimension than all previ-

ous intertemporal CGE models, it also is dimensionally constrained as the method can not be employed to solve larger regional models efficiently. There are no existing models, for example, with more than 12 or 15 regions.

Recently, Ha and Kompas (2016) and (Ha et al., 2017) apply a different approach to solving intertemporal and regional models, proposing a basic reordering of variables and equations method to shape the first-order differential matrix arising from intertemporal or regional CGE models into DBBD (Doubly Bordered Block Diagonal) or SBBD (Singly Bordered Block Diagonal) form. SBBD is a special case case of a DBBD reordering in the sense that there are only few intertemporal equations in the model that can be exploited to transform a DBBD matrix into SBBD form without significant loss in performance. The SBBD matrix can be easily solved by HSL’s MP\_48 solver HSL (2013) following Ha and Kompas (2016). The extension to the NDBBD approach in the current paper, along with the use of parallel processing techniques, provides an enormous increase in model dimension along with its portability. The reordering method can be applied to regional or intertemporal models, and even non-optimal recursive approaches to CGE modelling, greatly increasing computational efficiency.

Section 2 of the paper briefly provides the structure for the full intertemporal CGE model. Section 3 details the NDBBD approach and section 4 applies the NDBBD form to an intertemporal CGE model. Section 5 illustrates the benefits of our approach in terms of numerical performance. Section 6 shows the effects of climate change on the growth in GDP in a large dimensional setting and in comparison to more standard recursive approaches, and section 7 shows comparisons to recursive models in terms of the potential imposition of trade barriers across two regions. Section 8 concludes.

## 2. GTAP-INT

Put simply, a CGE model is a system of nonlinear equations that connect the demands for goods and services (or commodities), in a given economy, from households, firms and governments to the supplies of these goods and services from various producers in that economy and overseas. Firms combine inputs or factors of production such as land, labour and capital with intermediate products to produce final output which can then be sold domestically or in international markets. Taxes are levied and countries and regions are connected through a flow of imports and exports.

Solving a CGE model, either static, recursive or intertemporal, is as basic as solving any non-linear system. However, the underlying first-order matrix as a result of optimising behavior can be very complex (Hertel, 1997; Ha et al., 2017), with equations for equilibrium prices and quantities across a number of goods, along with supplies and demands over potentially many different countries, connected through time.

### 2.1. Regional GTAP

The GTAP model (Hertel, 1997) is a special version of the CGE model, a bottom-up regional model for the world economy. In its simplest representational form, a GTAP model consists of two interacting agents: consumers (regional households) and producers, along with government, across more than 50 commodity groups. Regional households obtain income from factor incomes and pay taxes and buy goods and services from both domestic and international producers (via private and government expenditures). Producers, in turn, as in any CGE model, combine factors of production and other goods to produce output, in nested CES form, and sell their products to domestic regional households and governments, or internationally. The demands for goods depend on prices, incomes and other structural parameters.

The model itself consists of blocks of supply and demand equations for regional producers, households, investment demand, net-imports and the government, defined in an Armington-like structure or a nested

consumption structure where the agents' distribute their budget to types of commodities first, then to point of sale (domestic, imported) (Armington, 1969). The demand and supply for goods and services is connected via price linkages and a set of market clearing conditions. The Armington structure itself is nothing more than a particular form of the consumer's utility function over commodities, in CES form, ensuring that consumers satisfy conditions for optimality (e.g., equating marginal rates of substitution to relative prices), generating demand for both foreign and domestic goods.

For the purpose of our paper, we follow (Ha and Kompas, 2014) and look at a GTAP model as a regional CGE model, which consists of separable regional economies with similar structures. There will be equations which exist only within regions and equations which connect two or more regional economies together. Similarly, there will be variables with values that appear only in regional equations or will be spread across regions in a single equation. Different structures will have implications for the structure and reordering of the relevant first-order matrix when we solve the system (Ha and Kompas, 2014).

### 2.2. Intertemporal GTAP

A fully defined intertemporal version of the GTAP model is developed in (Ha et al., 2017), where we replace fixed capital formation and given allocations of investment across regional equation blocks in the GTAP model by a long run profit (dividend) maximisation routine for the producer, resulting in sets of two motion equations for capital accumulation and its shadow price, or (using GTAP notation):

$$\dot{QO}(\textit{capital}, r, t) = CAPINCR(r, t) - \delta(r)QO(\textit{capital}, r, t) \tag{1}$$

$$\dot{\mu}(r, t) = \mu(r, t)[RORG(t) + \delta(r)] - \frac{\phi(r, t)}{2} \left( \frac{CAPINCR(r, t)}{QO(\textit{capital}, r, t)} \right)^2 - PS(\textit{CGDS}, r, t) - PS(\textit{capital}, r, t) \tag{2}$$

where  $QO(\textit{capital}, r, t)$  is the capital stock in region  $r$  at time  $t$ ;  $RORG(t)$  is the world interest rate at time  $t$ ;  $CAPINCR(r, t)$  is the capital increment (investment) in region  $r$  at time  $t$ ;  $\delta(r)$  is the depreciation rate;  $\mu(r, t)$  is the shadow price of capital in region  $r$  at time  $t$ ;  $\phi(r, t)$  is the investment coefficient in region  $r$  at time  $t$ , which shows how much extra investment must be made to obtain \$1 increase in capital increment  $CAPINCR(r, t)$ ;  $PS(\textit{CGDS}, r, t)$  is the price of capital goods (CGDS) in region  $r$  at time  $t$ ; and lastly,  $PS(\textit{capital}, r, t)$  is the rental price of capital in region  $r$  at time  $t$ .

By employing Equations (1) and (2) and time index for all of the GTAP model equations, we thus link all global economies over time using forward (Equation (1)) and backward (Equation (2)) connections. For the system to be solvable, there will be two additional transversality equations for each pair of Equations (1) and (2): one initial condition (fixing initial capital  $QO(\textit{capital}, r, 0)$ ); and one terminal condition:  $\dot{\mu}(r, t) = 0$ .

As Ha and Kompas (2016) show, the resulting system of GTAP equations over time can be viewed as identically separable structural economies for each period of time. The equations and variables can thus be classified into intra-temporal and intertemporal equations and variables and reordered to shape the first-order differential matrix in terms of DBBD or, as a special case, an SBBD matrix. We now show how this classification of equations and variables by region and time is needed to solve a model with very large matrices more efficiently, in NDBBD form.

### 3. The NDBBD matrix and direct method for solving a linear system

In our previous work, Ha and Kompas (2014), we detailed the structure of a DBBD matrix. In this paper, we will look at a nested structure

with two layers of DBBD matrices. To illustrate, we begin with a linear problem with a matrix in DBBD form:

$$\begin{pmatrix} A_1 & & & C_1 \\ & A_2 & & C_2 \\ & & \dots & \dots \\ & & & A_I & C_I \\ B_1 & B_2 & \dots & B_I & D \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \\ \dots \\ X_I \\ X_D \end{pmatrix} = \begin{pmatrix} Y_1 \\ Y_2 \\ \dots \\ Y_I \\ Y_D \end{pmatrix} \tag{3}$$

where  $A_i$  ( $i = 1 \dots I$ ) and  $D$  are rectangular matrices. The interface matrix  $D$  has dimension  $S \times S$ .

The nested structure begins with matrices  $A_i$ . In their turn,  $A_i$  are also DBBD matrices:

$$\begin{pmatrix} a_1^i & & & c_1^i \\ & a_2^i & & c_2^i \\ & & \dots & \dots \\ & & & a_L^i & c_L^i \\ b_1^i & b_2^i & \dots & b_L^i & d^i \end{pmatrix}$$

where  $a_l^i$  ( $l = 1 \dots L$ ) and  $d^i$  are also rectangular matrices. The interface matrices  $d^i$  have dimension  $s^i \times s^i$ .

The solution method for a NDBBD matrix involves a backward and forward recursive solution of DBBD matrices, where smaller inner DBBD matrices are 'pre-solved' first, then the outer DBBD matrix is 'pre-solved' to generate an 'outer-interface problem'. The solution of the outer-interface problem will be used to 'back-solve' for all leftover unknowns. The solution algorithm is similar to the one that has been used by Yamazaki and Li (2011) and later by Ha and Kompas (2014), with the exception that now the solution algorithm is nested. In particular, the solution algorithm has five major steps as follows.

The LU decomposition and linear equation solving procedure in Steps 1, 2, 3 and 5 can be done in parallel. That is one key to our better numerical performance. Another is the size of interface problems in step 1c and step 4 (Ha and Kompas, 2014). The matrices in these steps (i.e., interface matrices) are not sparse and difficult to solve if they are large, so the ordering of variables and equations should be optimised for smaller interface matrices. Furthermore, the matrices  $v_{is}$  and  $V_i$  are solutions of a multiple right-hand side problem, hence they also are not sparse and should not be stored explicitly to conserve memory. Instead, we store its product with matrices  $b_l^i$  or  $B_i$  to reduce storage requirements as suggested by Ha and Kompas (2014).

### 4. GTAP-INT model and its NDBBD form

From Ha and Kompas (2016) and Ha and Kompas (2014), we know how to reorder equations and variables of intertemporal and regional CGE models to shape the first-order differential matrix to solve the system efficiently. In this section, we take a step forward by reordering the equations and variables of an intertemporal regional CGE model both by time and region to shape the first-order differential matrix in NDBBD form.

Following Ha and Kompas (2016) and Ha and Kompas (2014), we assume that we can classify equations and variables into intra-regional and inter-regional and intertemporal and intra-temporal variables. We then will have three kinds of variables: (1) those variables that are both inter-temporal and inter-regional; (2) Those variables that are intra-temporal but not inter-regional; and lastly, (3) the rest of the variables (which are usually inter-temporal variables). We also classify equations into three comparable types: (1) those equations that are both inter-temporal and inter-regional; (2) Those equations that are intra-temporal but not inter-regional; and lastly, (3) the rest of the equations. The rule of thumb is if a variable has a value in two or more periods in a single equation, then the variable can be classified as an inter-temporal variable and it will be moved to the right-hand side to form the column

**Algorithm 1** Solution for NDBBD problem.

Step 1: Preparation:

Step 1a: LU decomposition  $a_i^i$  save LU factor matrices.

Step 1b: Using the same LU decomposition solve the multiple right hand side problem:  $a_i^i v_{is}^i = c_{is}^i$ ,  $v_{is}^i$  are temporary variables and  $s \in s^i$ .  $v_{is}^i$  forms the matrix  $v_i^i$ .

Step 1c: Perform LU decomposition of interface matrix  $ip_i = d^i - \sum_l^L b_l^i v_l^i$  save LU factor matrices. Note that  $v_l^i$  can be a very dense matrix, hence will never be stored to save memory.  $b_l^i v_l^i$  will be saved instead.

Step 2: Solve problem  $A_i Z_i = Y_i$ , where  $Z_i$  is a temporary variable:

Step 2a: Let  $Y_i = (y_{i1}^i, y_{id}^i)$  and  $Z_i = (z_{i1}^i, z_{id}^i)$ . Solve  $a_i^i u_l^i = y_{i1}^i$  problem using saved LU factor matrices in Step 1a.  $u_l^i$  is a temporary variable.

Step 2b: Solve the problem:  $ip_i z_{id}^i = y_{id}^i - \sum_l^L b_l^i u_l^i$  using saved LU factor matrices for  $ip_i$  in Step 1c.

Step 2c: Calculate  $z_{i1}^i = u_{i1}^i - v_{i1}^i z_{id}^i = u_{i1}^i - \frac{1}{a_i^i} c_{i1}^i z_{id}^i$  using the same LU decomposition in Step 1a as the above.

Step 3: Solve the multiple right hand side problem:  $A_i V_i = C_i$ , where  $V_i$  is a temporary variable, and multiply  $V_i$  to  $B_i$  to produce matrix  $BV_i$ .  $V_i$  can be very dense matrix, hence will never be stored to save memory.  $BV_i$  will be saved instead. Step 3 is similar to Step 2 but with a multiple right hand side. Let  $C_{is} = (c_{ils}, c_{ids})$  and  $V_{is} = (v_{ils}, v_{ids})$  as element vectors of  $C_i$  and  $V_i$ , where  $l, d$  denote the part of  $C_i, V_i$  that belong to the diagonal  $L$  and interface parts of matrices  $A_i$ . Note that we repeat index set  $s$  here and it can be understood as the index of set  $S$ . Similarly, let  $B_i = (b_{il}, b_{id})$ .

Step 3a: Solve  $a_i^i u_{ils} = c_{ils}$  problem using saved LU factor matrices from Step 1a.  $u_{ils}$  is a temporary variable.

Step 3b: Solve the problem:  $ip_i v_{ids} = c_{ids} - \sum_l^L B_l^i u_{ils}$  using saved LU factor matrices of interface matrices  $ip_i$  from Step 1c.

Step 3c: Calculate  $v_{ils} = c_{ils} - u_{ils} v_{ids} = c_{ils} - \frac{1}{a_i^i} c_{il}^i v_{ids}$  using the same LU decomposition from Step 1a as the above.

Step 3d: Form matrix  $V_i$  from  $V_{is}$  and multiply  $B_i V_i$  and add together to form matrix  $BV_i$

Step 4: Solve the ‘outer interface problem’:  $(D - \sum_i^I BV_i) X_D = Y_D - \sum_i^I B_i Z_i$ .

Step 5: Calculate  $X_i = Z_i - V_i X_D = Z_i - \frac{1}{A_i} C_i X_D$  by resembling Step 3 above. This is to avoid storing the dense matrix  $V_i$ . Calculate a new vector  $CX_i = C_i X_D$ . We will solve the problem  $A_i CZ_i = CX_i$ , where  $CZ_i$  is a temporary variable. Let  $CX_i = (cx_{il}, cx_{id})$  and  $CZ_i = (cz_{il}, cz_{id})$  as in Step 3.

Step 5a: Solve  $a_i^i u_{il} = cx_{il}$  problem (using saved LU matrices in Step 1a), where  $u_{il}$  is another temporary variable.

Step 5b: Solve the problem:  $ip_i cz_{id} = cx_{id} - \sum_l^L b_{il} u_{il}$  using the saved LU decomposition of interface matrices  $ip_i$  in steps 1c.

Step 5c: Calculate  $cz_{il} = u_{il} - v_{il}^i cz_{id} = u_{il} - \frac{1}{a_i^i} c_{il}^i cz_{id}$  using the same LU decomposition as the above.

Step 5d: Form  $CZ_i$  from  $cz_{il}, cz_{id}$  and calculate  $X_i = Z_i - CZ_i$ .

border, or the equation will be classified as an intertemporal equation and will be moved downward to form bottom border blocks. The choice of either an equation or a variable to be intertemporal is there to minimise the border block involved. If, for example, the size of a variable is larger than the size of an equation (each equation and variable can have many other indices so their size will be different), then we choose the equation to be intertemporal and vice versa. Similarly with regional indexes. The detailed rules for equation and variable classification can be found in [Ha and Kompas \(2016\)](#) and [Ha and Kompas \(2014\)](#).

With the variables reordered by time for intra-temporal variables, so that every intra-temporal variable in a single period is ordered to be combined together in the same continuous space, all the rest of variables (e.g. intertemporal variables) are shifted to the far right hand side, after the last period. Within the same period, we again order all variables in the same regions (intra-regional) to be combined together in the same continuous space. Inter-regional (and intra-temporal) will be ordered last after the last region. We do the same for all equations.

For a single period of time, for example, when we apply a differential operator to equations, the first-order differential matrix has a DBBD form as matrix  $A_i$  in Section 3, or:

$$\begin{pmatrix} a_{r_1}^i & & & c_1^i \\ & a_{r_2}^i & & c_2^i \\ & & \dots & \dots \\ & & & a_{r_L}^i & c_L^i \\ b_{r_1}^i & b_{r_2}^i & \dots & b_{r_L}^i & d^i \end{pmatrix}$$

Off-border block, intra-regional variables thus have values only within one region and the intra-regional equation can house values for intra-regional variables only in one region. Therefore, diagonal blocks form together in regional spaces ( $r_1 \dots r_L$ ). On the right hand side and along the bottom of the matrix, inter-regional variables can have values in many regions, together, within a single equation, so that the right-hand side blocks,  $c_l^i$  ( $l \in 1 \dots L$ ), will not have block diagonal form. Similarly, inter-regional equations can house variables with values across

regions and they (or  $b_l^i$ ) will also not have block diagonal form.

Stepping through time and space gives:

$$\begin{pmatrix} A_{t_1} & & & C_1 \\ & A_{t_2} & & C_2 \\ & & \dots & \dots \\ & & & A_{t_i} & C_i \\ B_{t_1} & B_{t_2} & \dots & B_{t_i} & D \end{pmatrix}$$

where  $A_i$  now becomes  $A_{t_i}$  and we have the same (as above) DBBD matrix, albeit in a nested or NDBBD form. In the next section we show how this simple reordering method can help solve an intertemporal regional GTAP model more efficiently, in parallel.

**5. Numerical analysis**

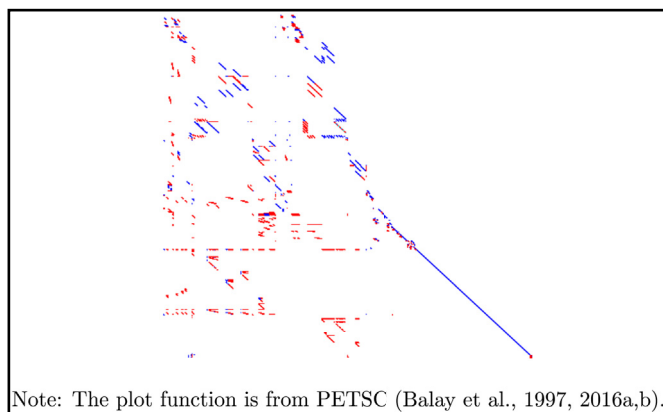
**5.1. GTAP model’s database**

We use GTAP database version 7.1 ([Narayanan and Walmsley, 2008](#)). The full database includes 112 countries and regions and 57 commodities. The GTAP model with its full database translates into a very large intertemporal CGE model with more than half a billion endogenous variables and 260 million exogenous variables (Model ID 6 in [Table 1](#)). Note, as well, as specified previously ([Ha and Kompas, 2016](#)), the intertemporal model needs to be solved for at least 40 periods to get accurate results. Therefore, for testing of numerical performance, we drop all informational equations in the GTAP model appendices, since these variables (mostly trading price, quantity and welfare indices) can be calculated easily from the model results. Trimming the appendices also removes some of the inter-regional equations (i.e., mostly equations with summations over regional indexes), which helps to improve computational performance.

**Table 1**  
GTAP model with varying levels of aggregation.

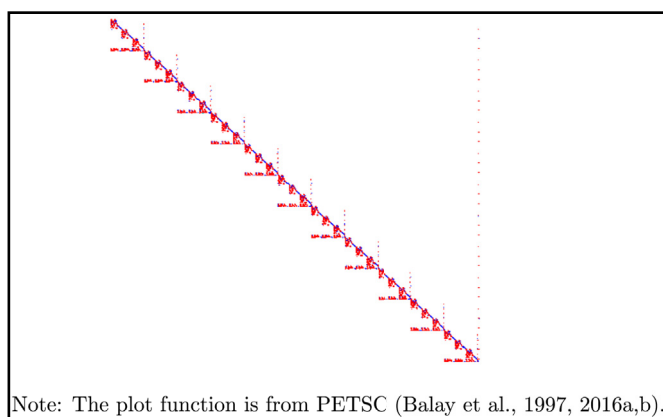
ID	Model Size	Number of endogenous variables	Number of exogenous variables	Number of non-zeros <sup>†</sup>
1	112 regions, 3 commodities, 47 periods	13939336	7850081	58,681,307
2	112 regions, 4 commodities, 47 periods	18,592,712	10,450,826	78,830,677
3	112 regions, 5 commodities, 47 periods	23,319,784	13,083,155	97,089,894
4	112 regions, 26 commodities, 47 periods	139612072	75,657,968	547479803
5	112 regions, 34 commodities, 47 periods	192462632	103159736	749419439
6	112 regions, 57 commodities, 47 periods*	505112836	260697767	1.815*10 <sup>9</sup>

Note: <sup>†</sup> Number of non-zeros is an approximation.  
Source: Authors' calculation.



Note: The plot function is from PETSC (Balay et al., 1997, 2016a,b).

**Fig. 1.** The matrix without ordering.  
Source: Authors' calculation.



Note: The plot function is from PETSC (Balay et al., 1997, 2016a,b).

**Fig. 2.** The reordered nested matrix.  
Source: Authors' calculation.

### 5.2. Numerical performance

Figs. 1 and 2 show a graphical representation of the original first-order derivative matrix of the GTAP-INT model and the reordered matrix by our NDBBD method. For simplicity of illustration, the figure is drawn from the GTAP model with only three regions and three commodities with eleven time periods. The red and blue dots in Figs. 1 and 2 represent negative and positive elements of the first-order differential matrix.

Table 2 shows the computational performance of our NDBBD method versus the SBBB solution method in Ha and Kompas (2016). We do not show the computational solution for the conventional LU

**Table 2**  
Calculation time in seconds.

ID	SBBB reordering	NDBBD reordering	Performance ratio
1	46.133858	115.809060	39.84%
2	85.192744	149.345154	57.04%
3	116.959756	205.926200	56.8%
4	3043.344883	3984.619880	76.38%
5	5352.380219	7500.384889	71.36%
6	–	26934.744446	–

Notes: (a) See Appendix B for machine configuration and Table 1 for model size and number of variables. (b) All numerical experiments are carried out with a one-step Johansen method or (equivalently) based on the first-order Taylor expansion approximation (Pearson, 1991; Dixon et al., 1992). Time is counted for the linear system (matrix) solution only.  
Source: Authors' calculation.

decomposition method (for example MA48 solver in Harwell Subroutine Library HSL, 2013), as Ha and Kompas (2016) have shown a clear advantage of the SBBB method over the conventional LU decomposition for intertemporal CGE models. We also skip showing the numerical results of the DBBD method (Ha and Kompas, 2014) for the GTAP-INT model, even though it is also a bottom-up regional model. When we reorder GTAP-INT by region instead of by time, the netcut Ha and Kompas (2014) usually is much higher than reordering the model by time. The computational performance of DBBD in this case is worse than SBBB.

We design the NDBBD solution method with the aim of solving very large models. It has less advantage in the solution of smaller models because we will use a hard disk as a temporary space to store variables. Our aim is to solve very large models, which can not be solved by a single computer and requires parallel processing techniques. Table 2 shows that our NDBBD method is slower than the SBBB method with a smaller model size. However, the performance gap narrows as the size of the model increases. Finally, with more than half a billion equations, the SBBB method is unable to solve the system as the model is too large to feed into the memory of a single computer alone.

Ha and Kompas (2016) use a parallel solver in Harwell Subroutine Library (HSL, 2013) to solve intertemporal CGE models using the SBBB ordering technique. The solver requires the matrix to feed-in a single process before it redistributes the matrix blocks to other processes to solve the matrix in parallel. This bottleneck prevents matrix size from unlimited growth. In fact, an indexation convention of that solver prevents models with very large block sizes to be solved. We have amended this procedure to be able to handle models with nearly 200 million equations (the SBBB case). However, when the matrix size reaches half a billion equations, SBBB fails. Only our NDBBD approach can solve the model.

### 6. Solving for world climate change effects

Climate change is a world-wide phenomenon and requires a coordinated global action. The IPCC (2014) report has shown a surge in averaged combined land and ocean surface temperature by 0.85 [0.65 to 1.06]°C over the period 1880 to 2012. As a result, changes in climate have caused wide-spread impacts on all continents and oceans, e.g., changes in precipitation levels, shifts in the ice cap, altered hydrological systems, threats to the livelihood of terrestrial, freshwater and marine species and negative impacts on crop yields. Changes in extreme weather events have also been observed in the last few decades (IPCC, 2014).

Nevertheless, climate change affects countries differently. This fact alone is undoubtedly an obstacle to consensus and may partly explain why conflicting arguments remain despite the Paris Agreement. There is a vast literature evaluating the effects of climate change on the world economy in a CGE setting: Bigano et al. (2008), for example, evaluate the impact of changing sea levels and tourism flows to the world economy using a modified static GTAP-E model; Burniaux and Truong (2002) and Bosello et al. (2006) evaluate the impact of climate change via changes in human health also using a GTAP-E model; Roson and Sartori (2016) take a more broader approach to climate change impacts by estimating climate change effects via six different pathways, i.e., sea level rise, variation in crop yields, heat and labor productivity, human health and tourism, and household energy demand. However, Roson and Sartori (2016) can only evaluate a first-order approximation of the impacts on real GDP by evaluating the impact of the above pathways to each of the GDP components.

In this section, we take advantage of our new solution approach to evaluate the impact of climate change using an intertemporal CGE model for the entire world. This provides a clear advantage over the traditional static aggregate approach (e.g., Bigano et al. (2008) was

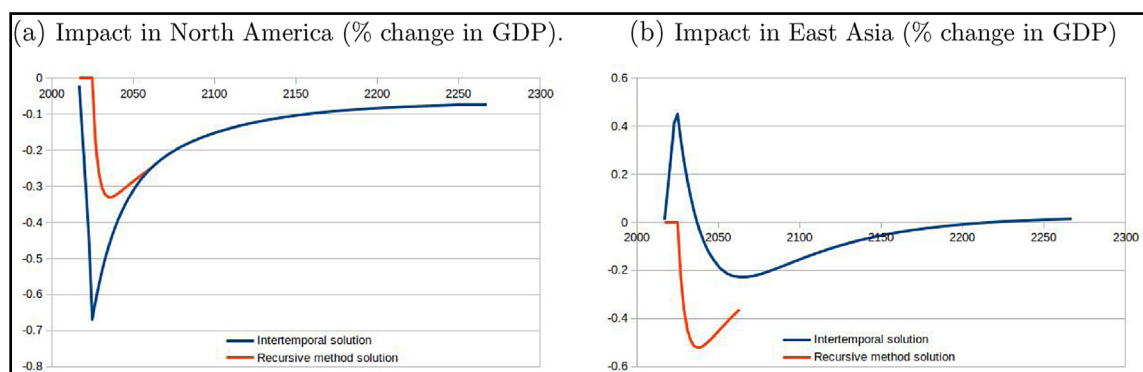
able to run a GTAP model for only 16 regions). For the climate change shock, we use the Roson and Sartori (2016) estimation of the six climate change impact pathways as 3-category shocks to our model. Specifically, we allow for (a) land losses due to sea level rises via changes in factor endowments for land in the model; (b) variations in crop yields, heat and labor productivity and human health impacts through shifts in model-specific productivity variables; and, (c) demand shifts of selected goods (electricity, gas, transportation and recreation activities etc.). The climate change shocks were applied gradually in linear form from now until 2100. An overall 3°C increase in the global temperature was adopted.

Table 3 lists the top ten ‘winners and losers’ from climate change impacts, as a comparator to the first-order recursive results in Roson and Sartori (2016). Results for the full set of 112 countries are available from the authors on request. While the impact patterns (negative or positive) are roughly comparable between the two approaches, the magnitudes of the impacts are very different. For some of countries, in particular those where climate change effects are small or even positive, the first-order recursive estimation tends to overstate the climate change impact on GDP in comparison with the GTAP-INT approach. This is not surprising since agents in the intertemporal CGE model are adjusting their consumption and production behaviours to partially mitigate the climate change impact. For other countries, however, the negative effects of climate change in the intertemporal GTAP-INT model dominate, and are much larger than the first-order recursive estimates. It depends on location and the distance to the equator (e.g., Indonesia, Sri Lanka, India, the Philippines and the Rest of Western Africa). Indeed, despite any mitigation behaviour, climate change impacts are clearly severe for some regions, especially for West Africa, South Asia and Southeast Asia. Results for GDP falls in the USA are -0.48 and -0.16, or with falls in GTAP-INT three times larger than the results in Roson and Sartori (2016).

**Table 3**  
Top ten ‘winners and losers’ from climate change: intertemporal vs first-order estimation (% change in GDP).

Countries/regions	GTAP-INT*	First order**	Countries/regions	GTAP-INT*	First order**
Rest of Western Africa	-24.00	-8.93	Canada	0.16	1.27
Senegal	-19.15	-9.58	Poland	0.16	0.96
Nigeria	-15.54	-13.93	Austria	0.21	1.95
Sri Lanka	-13.34	-6.40	Czech Republic	0.24	1.44
Philippines	-12.93	-7.42	Latvia	0.27	0.97
Malawi	-12.90	-9.76	Rest of Europe	0.34	0.01
India	-12.76	-6.24	Slovakia	0.63	1.23
Rest of Southeast Asia	-12.71	-7.32	Malta	0.65	-6.55
Tanzania	-12.51	-8.79	Lithuania	0.66	1.10
Indonesia	-12.45	-6.80	Belarus	1.17	0.18

Source: \* Authors’ calculation. \*\* From Roson and Sartori (2016).



**Fig. 3.** Intertemporal vs recursive solution methods: trade policy analysis.  
Sources: Authors’ calculation.

## 7. Intertemporal vs recursive solutions: trade barriers

In this section we further examine the difference between the recursive and our intertemporal analysis. Table 3 has already indicated the differences between GTAP-INT and the first-order recursive approaches for climate change impacts. Here, we concentrate on the effects of trade barriers. In order to minimise the ‘noise’ in our comparison, we adopt a naive static expectations assumption for our recursive model, i.e., we replace the shadow price of capital ( $\mu$ ) in Equation (2) as an expectation variable and assume a naive static price expectation, so that the next period shadow price is equal to that of the current period ( $\mu_{t+1} = \mu_t$ ). By doing so, we simply assume  $\dot{\mu}_t = 0$  and the motion Equation (2) for  $\mu$  becomes the intra-temporal equation. We solve the model forward by the capital motion equation, so that next period’s capital will be equal to the current period capital stock plus investment minus depreciation.

There are two points to be noted with the recursive method. First, there will be no guarantee of convergence in the long run even though both models are constrained (by our design) to have the same steady state. The recursive model consists of a system of first-order differential equations for capital stocks (in each country and region) and depending on the form of the expectation function, it is difficult to prove convergence. Even with rational expectations, a recursive analysis does not imply a steady state outcome Dixon et al. (2005).

Second, it is interesting to see that our NDBBD method can also apply to a recursive methodology and that we can employ parallel computing resources to accelerate the solution time in this setting. A recursive model, in a sense, is just an intertemporal model without the shadow price (or jump) motion equation. Therefore, it can be declared in the form of an intertemporal model and its equation and variables can be reordered by time and region to form an NDBBD matrix and solved by our NDBBD method. That makes recursive modelling much easier to do and we do not need to obtain special software packages to generate recursive solutions.

To compare the two intertemporal and recursive solution methods, we simulate a typical hypothetical trade protectionism policy change, where, for example, North America applies a 40% increase in the tariff barrier for goods and services from East Asia, implemented (in full) ten years from now. Fig. 3 shows quite different impacts on GDP for the two regions and across models. Since there is no feedback from the future, the recursive model is not responsive to the change in an announced policy (the horizontal line at the beginning of the sequence) until the actual increase in the tariff rate; while agents in intertemporal model can adjust their current consumption and production behaviors in advance, to partly mitigate the impact of an announced change in policy.

The above analysis indeed shows a clear advantage of the intertemporal solution in comparison with a recursive solution method, even in the format where we use a static expectations assumption. At the very least, if we have to solve models recursively, our NDBBD method also helps to reduce computational time by employing parallel resources, something that is not possible for the traditional ‘one period after another’ sequential recursive method.

## 8. Concluding remarks

In this paper, we provide an algorithm to solve NDBBD matrices in parallel. By reshaping the first-order matrix arising from intertemporal regional CGE models into NDBBD form, and employing parallel processing techniques to accelerate computation time, the algorithm can help to solve very large regional and intertemporal models, which can not be solved otherwise.

We have thus shown that we do not need to reduce the size of CGE models, either in terms of size (i.e., aggregating by region) or by adopting static or adaptive price level expectation assumptions, and hence losing valuable information, for the sake of computational time and capacity. Although this paper is largely a technical exercise, the model approach should benefit both researchers and policy makers around the world. In our numerical analysis, for example, we show how GTAP-INT can be used to solve a global trading system with more than 500 million equations; a size large enough to determine the potential economic effects of climate change for each of 112 countries to 2100. That provides a level of detail not possible until now. We also compare our method to the standard recursive approach for both climate and trade effects and show that our method is capable of analysing intertemporally optimal or forward-looking behaviour, which cannot be done with traditional static and adaptive expectations or recursive model approaches.

Our next step is to determine the actual losses in GDP (i.e., not just falls or changes in growth rates) from climate change effects for all 140 countries in the GTAP database across different potential global temperature increases. This will allow us to determine the potential gains from complying with the Paris Accord and a full comparative economic analysis of the effects of climate change for each country in the database over time. Existing methods, which aggregate countries into regions mask individual country effects with regional averages. This too loses valuable information needed by policy makers.

## Acknowledgements

The computations and graphical representation in this paper have been carried out with the help of the following software libraries: Portable, Extensible Toolkit for Scientific Computation (PETSc) 3.7.0 at Argonne National Laboratory (Balay et al., 1997; 2016a, b); Message Passing Interface (MPICH) 3.2.7; HSL Mathematical Software Library (HSL, 2013); the GNU Compiler Collection (GCC) 6.3.0; and R R Core Team (2016). The details on software libraries and packages can be found in Appendix A. The authors would like to thank the developers of these software libraries for sharing their work. Appendix B describes our hardware configuration. The computational source code used in our paper is available on request for non-commercial purposes.

This paper was first presented at 3rd International Workshop on ‘Financial Markets and Nonlinear Dynamics’ (FMND), June 1–2, 2017, Paris, France, with special thanks to the organisers.

## Appendix A. Software

**Table A.4**  
The list of software and libraries used in the numerical simulations.

	Name	Version	Source
Operation system	Ubuntu	16.04	<a href="http://www.ubuntu.com">http://www.ubuntu.com</a>
Compiler	GCC	6.2.0	<a href="http://gcc.gnu.org">http://gcc.gnu.org</a>
Message Passing Interface	MPICH2	3.2-7	<a href="http://www.mpich.org">http://www.mpich.org</a>
Matrix manipulation	Petsc	3.7.0	<a href="http://www.mcs.anl.gov/petsc">http://www.mcs.anl.gov/petsc</a>
Matrix solver	HSL <sup>1</sup>	2013	<a href="http://www.hsl.rl.ac.uk">www.hsl.rl.ac.uk</a>
Graphical plot	R <sup>2</sup> (R Core Team, 2016)	3.2.3	<a href="https://www.r-project.org">https://www.r-project.org</a>

Note: (a) Sub-packages used: MA48,HSL\_MC51. (b) Packages used: reshape 0.8.6 (Wickham, 2007) mapprool 0.8.41 (Bivand and Lewin-Koh, 2017); ggplot2 2.2.1 (Wickham, 2009); RColorBrewer 1.1.2 (Neuwirth, 2014).

## Appendix B. Hardware configuration

The parallel computing exercises are carried out with three computers: Lenovo 32 G ram 128 G SSD and 500 G HDD. The network devices are TP-LINK TL-SG1008D Gigabit switch.

## Appendix C. Supplementary data

Supplementary data related to this article can be found at <https://doi.org/10.1016/j.econmod.2018.08.011>.

## References

- Armington, P.S., 1969. A theory of demand for products distinguished by place of production (une théorie de la demande de produits différenciés d'après leur origine) (una teoría de la demanda de productos distinguiéndolos según el lugar de producción). Staff Pap. (Int. Monetary Fund) 16 (1), 159–178.
- Balay, S., Abhyankar, S., Adams, M.F., Brown, J., Brune, P., Buschelman, K., Dalcin, L., Eijkhout, V., Gropp, W.D., Kaushik, D., Knepley, M.G., McInnes, L.C., Rupp, K., Smith, B.F., Zampini, S., Zhang, H., Zhang, H., 2016a. PETSc Users Manual, Technical Report ANL-95/11 - Revision 3.7. Argonne National Laboratory.
- Balay, S., Abhyankar, S., Adams, M.F., Brown, J., Brune, P., Buschelman, K., Dalcin, L., Eijkhout, V., Gropp, W.D., Kaushik, D., Knepley, M.G., McInnes, L.C., Rupp, K., Smith, B.F., Zampini, S., Zhang, H., Zhang, H., 2016b. PETSc Web Page. <http://www.mcs.anl.gov/petsc>.
- Balay, S., Gropp, W.D., McInnes, L.C., Smith, B.F., 1997. Efficient management of parallelism in object oriented numerical software libraries. In: Arge, E., Bruaset, A.M., Langtangen, H.P. (Eds.), *Modern Software Tools in Scientific Computing*. Birkhäuser Press, pp. 163–202.
- Bigano, A., Bosello, F., Roson, R., Tol, R.S.J., 2008. Economy-wide impacts of climate change: a joint analysis for sea level rise and tourism. *Mitig. Adapt. Strat. Glob. Change* 13 (8), 765–791.
- Bivand, R., Lewin-Koh, N., 2017. Mapprool: Tools for Reading and Handling Spatial Objects. R package version 0.8-41.
- Bosello, F., Roson, R., Tol, R.S., 2006. Economy-wide estimates of the implications of climate change: human health. *Ecol. Econ.* 58 (3), 579–591.
- Burniaux, J.-M., Truong, T., 2002. GTAP-E: an Energy-environmental Version of the GTAP Model. GTAP Technical Paper No.16.
- Dixon, P.B., Parmenter, B., Powell, A.A., Wilcoxon, P.J., 1992. Notes and problems in applied general equilibrium economics. In: Bliss, C., Intriligator, M. (Eds.), *Advanced Textbooks in Economics*, vol. 32. North-Holland, Amsterdam, London, New York, Tokyo.
- Dixon, P.B., Pearson, K., Picton, M.R., Rimmer, M.T., 2005. Rational expectations for large CGE models: a practical algorithm and a policy application. *Econ. Modell.* 22 (6), 1001–1019.
- EC, IMF, OECD, UN, World Bank, 2009. *System of National Accounts 2008*. New York.
- Fair, R.C., 1979. An analysis of a macro-econometric model with rational expectations in the bond and stock markets. *Am. Econ. Rev.* 69 (4), 539–552.
- Fair, R.C., Taylor, J.B., 1983. Solution and maximum likelihood estimation of dynamic nonlinear rational expectations models. *Econometrica* 51 (4), 1169–1185.
- Ha, P.V., Kompas, T., 2014. Solving GTAP model in parallel using a doubly bordered block diagonal ordering technique. In: 2nd International Workshop on “Financial Markets and Nonlinear Dynamics” (FMND), June 4-5, 2015, Paris (France).
- Ha, P.V., Kompas, T., 2016. Solving intertemporal CGE models in parallel using a singly bordered block diagonal ordering technique. *Econ. Modell.* 52 (Part A), 3–12 Special Issue on Recent Developments in Decision-Making, Monetary Policy and Financial Markets.
- Ha, P.V., Kompas, T., Nguyen, H.T.M., Long, C.H., 2017. Building a better trade model to determine local effects: a regional and intertemporal GTAP model. *Econ. Modell.* 67, 102–113.
- Hertel, T.W. (Ed.), 1997. *Global Trade Analysis: Modeling and Applications*. Cambridge University Press, Cambridge, New York.
- HSL, 2013. *A Collection of Fortran Codes for Large Scale Scientific Computation*. <http://www.hsl.rl.ac.uk>.
- IPCC, 2014. In: Core Writing Team, Pachauri, R.K., Meyer, L.A. (Eds.), *Climate Change 2014: Synthesis Report, Contribution of Working Groups I, II and III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*. IPCC, Geneva, Switzerland, p. 151.
- Johansen, L., 1960. *A Multi-sector Study of Economic Growth*. North-Holland Pub. Co.
- McKibbin, W.J., 1987. *Numerical Solution of Rational Expectations Models with and without Strategic Behaviour*. RBA Research Discussion Papers rdp8706. Reserve Bank of Australia.
- McKibbin, W.J., Sachs, J.D., 1991. *Global Linkages: Macroeconomic Interdependence and Cooperation in the World Economy*. Brookings Institution, Washington, D.C.
- Narayanan, B.G., Walmsley, T.L. (Eds.), 2008. *Global Trade, Assistance, and Production: the GTAP 7 Data Base*, Center for Global Trade Analysis. Purdue University.
- Neuwirth, E., 2014. *RColorBrewer: ColorBrewer Palettes*. R package version 1.1-2.
- Pearson, K., 1991. *Solving Nonlinear Economic Models Accurately via a Linear Representation*. Centre of Policy Studies/IMPACT Centre Working Papers ip-55. Monash University, Centre of Policy Studies/IMPACT Centre.
- R Core Team, 2016. *R: a Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.
- Roson, R., Sartori, M., 2016. Estimation of climate change damage functions for 140 regions in the GTAP 9 database. *J. Global Econ. Anal.* 1 (2), 78–115.
- Saunders, M., Gill, P., Murray, W., Wright, M., O'Sullivan, M., Eikland, K., Zhang, Y., Henderson, N., Ma, D., 2018. *Lusol: Sparse LU for Ax = B*. Dept of Management Science and Engineering (MS&E), Stanford University. Online at: <https://web.stanford.edu/group/SOL/software/lusol/>. (Accessed 4 May 2018).
- United Nations, 1953. *A System of National Accounts and Supporting Tables, Studies in Methods*. Series F No 2, New York.
- Wickham, H., 2007. Reshaping data with the reshape package. *J. Stat. Software* 21 (12).
- Wickham, H., 2009. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag, New York.
- Yamazaki, I., Li, X., 2011. On techniques to improve robustness and scalability of a parallel hybrid linear solver. In: Palma, J., Daydé, M., Marques, O., Lopes, J. (Eds.), *High Performance Computing for Computational Science – VECPAR 2010*, Vol. 6449 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 421–434.