Open Archive TOULOUSE Archive Ouverte (OATAO)
OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: [http://oatao.univ-toulouse.fr/](http://oatao.univ-toulouse.fr/)
Eprints ID: 15168

The contribution was presented at SEG 2013:

To cite this version: Weisbecker, Clément and Amestoy, Patrick and Boiteau, Olivier and Brossier, Romain and Buttari, Alfredo and L'Excellent, Jean-Yves and Operto, Stéphane and Virieux, Jean 3D frequency-domain seismic modeling with a Block Low-Rank algebraic multifrontal direct solver. (2013) In: Annual meeting of the Society of Exploration Geophysicists (SEG 2013), 22 September 2013 - 27 September 2013 (Houston, Texas, United States).

Any correspondence concerning this service should be sent to the repository administrator: staff-oatao@listes-diff.inp-toulouse.fr
3D frequency-domain seismic modeling with a Block Low-Rank algebraic multifrontal direct solver.

C. Weisbecker†, P. Amestoy‡, O. Boiteau§, R. Brossier∥, A. Buttari‡, J. Y. L’Excellent¶, S. Operto§, J. Virieux∗∗
∗INPT/ENSEEIHT-IRIT; ‡CNRS-IRIT; §INRIA-LIP; §Geozazur-CNRS-UNSA; ¶LIGT-UJF; ∥EDF R&D; ‡Speaker

SUMMARY

Three-dimensional frequency-domain full waveform inversion of fixed-spread data can be efficiently performed in the visco-acoustic approximation when seismic modeling is based on a sparse direct solver. We present a new algebraic Block Low-Rank (BLR) multifrontal solver which provides an approximate solution of the time-harmonic wave equation with a reduced operation count, memory demand and volume of communication relative to the full-rank solver. We show some preliminary simulations in the 3D SEG/EAGE overthrust model, that give some insights on the memory and time complexities of the low-rank solver for frequencies of interest in full-waveform inversion (FWI) applications.

INTRODUCTION

Seismic modeling and FWI can be performed either in the time domain or in the frequency domain (e.g., Virieux and Operto, 2009). One distinct advantage of the frequency domain is to allow for a straightforward implementation of attenuation in the linear system by exploiting the low-rank properties of elliptic partial differential operators. Their approach exploits the regular pattern of the impedance matrix built with finite-difference stencils on uniform grid. In this study, we present a new algebraic Block Low-Rank (BLR) multifrontal solver (Duff and Reid, 1983). The algebraic approach is amenable to matrices with a non regular pattern such as those generated with finite element methods on unstructured meshes. In the first part, we review the main features of the BLR multifrontal solver and highlight the main pros and cons relative to the approach of Wang et al. (2011). Second, application to the 3D SEG/EAGE overthrust model gives quantitative insights on the memory and operation count savings provided by the BLR solver.

BLK LOW-RANK MULTIFRONTAL METHOD

Multifrontal method

The multifrontal method was first introduced by Duff and Reid (1983). Being a direct method, it computes the solution of a sparse system \( Ax = b \) by means of a factorization of \( A \) under the form \( A = LU \) (in the unsymmetric case). This factorization is achieved through a sequence of partial factorizations, performed on relatively small, dense matrices, called fronts. With each front are associated two sets of variables: the fully-summed (FS) variables, whose corresponding rows and columns of \( L \) and \( U \) are computed within the current front, and the non fully-summed (NFS) variables, which receive updates resulting from the elimination of FS variables. At the end of each partial factorization, the partial factors \( [L_{11}L_{21}] \) and \( [U_{11}U_{12}] \) are stored apart and a Schur complement referred to as a contribution block (CB) is held in a temporary memory area called CB stack, whose maximal size depends on several parameters. As the memory needed to store the factors is incompressible (in full-rank), the CB stack can be viewed as an overcost whose peak has to be minimized. The structure of a front before and after partial factorization is shown in Fig. 1.

Figure 1: A front before (a) and after (b) partial factorization.

The computational and memory requirements for the complete factorization strongly depend on how the fronts are formed and on the order in which they are processed. Reordering techniques such as nested dissection are used to ensure the effi-
ciency of the process: a tree, called elimination tree (Schreiber, 1982) is created, with a front associated with each of its nodes. Any post-order traversal of this tree gives equivalent properties in terms of factors memory and computational cost.

**Block Low-Rank (BLR) matrices**

A flexible, efficient technique can be used to represent fronts with low-rank subblocks based on a storage format called Block Low-Rank (BLR, see Amnesty et al. (2012)). Unlike other formats such as ℋ-matrices (Hackbusch, 1999) and HSS matrices (Xia et al., 2009), the BLR one is based on a flat, non-hierarchical blocking of the matrix which is defined by conveniently clustering the associated unknowns. A BLR representation of a dense matrix is shown in equation (1) where we assume that p subblocks have been defined. Subblocks $B_{ij}$ of size $m_{ij} \times n_{ij}$ and numerical rank $k_{ij}$ are approximated by a low-rank product $X^T_{ij} K_{ij}^T$ at accuracy $\varepsilon$, when $k_{ij} (m_{ij} + n_{ij}) \leq m_{ij} n_{ij}$ is satisfied.

$$
\tilde{F} = 
\begin{bmatrix}
\tilde{B}_{11} & \tilde{B}_{12} & \ldots & \tilde{B}_{1p} \\
\tilde{B}_{21} & \ldots & \ldots & \ldots \\
\vdots & \ldots & \ldots & \ldots \\
\tilde{B}_{p1} & \ldots & \ldots & \tilde{B}_{pp}
\end{bmatrix}
$$

(1)

In order to achieve a satisfactory reduction in both the complexity and the memory footprint, subblocks have to be chosen to be as low-rank as possible (e.g., with exponentially decaying singular values). This can be achieved by clustering the unknowns in such a way that an admissibility condition (Bebendorf, 2008) is satisfied. This condition states that a subblock $B_{ij}$, interconnecting variables of $i$ with variables of $j$, will have a low rank if variables of $i$ and variables of $j$ are far away in the domain, intuitively, because the associated variables are likely to have a weak interaction, as illustrated in Fig. 2(a). Subblocks which represent self-interactions (typically the diagonal ones) are for this reason always full-rank. In practice, the subgraphs induced by the FS variables and the NFS variables are algebraically partitioned with a suitable strategy.

![Figure 2](image_url)

**Block Low-Rank multifrontal solver**

A BLR multifrontal solver consists in approximating the fronts with BLR matrices, as shown Figure 2(b). BLR representations of $[L_{11}U_{11}], [L_{21}, U_{12}]$ and $CB$ are computed separately. Note that in $L_{21}$ and $U_{12}$, there are no self-interactions. The partial factorization is then adapted to benefit from the compressions using low-rank products instead of full-rank standard ones. An example of a BLR partial factorization algorithm is given in Algorithm 1. For sake of clarity, an unblocked version is presented although this algorithm is, in practice, applied panelwise with numerical pivoting.

**Algorithm 1** Unblocked FSCU incomplete factorization of a front.

1. **Input**: a front $F$
2. **Output**: $[L_{11}L_{21}], [U_{11}U_{12}]$ and a Schur complement $CB$
3. Factor ‘$F$’: $F_{11} = L_{11}[U_{11}U_{12}]$
4. Solve ‘$S$’: $L_{21} = F_{21}U_{11}^{-1}$
5. Compress ‘$C$’: $L_{21} \simeq X_1^T Y_1^T; U_{12} \simeq X_2^TY_2^T$
6. Update ‘$U$’: $CB = F_{22} - X_2^T (Y_2^T Y_1^T) X_1^T$

Although compression rates may not be as good as those achieved with hierarchical formats, BLR offers a good flexibility thanks to its simple, flat structure. Many variants of Algorithm 1 can be easily defined, depending on the position of the ‘$C$’ phase. For instance, it can be moved to the last position if one needs an accurate factorization and an approximated, faster solution phase. This might be a strategy of choice for FWI application, where a large number of right-hand sides must be processed during the solution phase. In a parallel environment, the BLR format allows for an easier distribution and handling of the frontal matrices. Pivoting in a BLR matrix can be more easily done without perturbing much the structure. Lastly, converting a matrix from the standard representation to BLR and vice versa, is much cheaper with respect to the case of hierarchical matrices (see Table 1 for the low global cost of compressing fronts into BLR formats). This allows to switch back and forth from one format to the other whenever needed at a reasonable cost; this is, for example, done to simplify the assembly operations that are extremely complicated to perform in any low-rank format. As shown in Fig. 3, the $O(N^2)$ complexity of a standard, full rank solution of a 3D problem (of $N$ unknowns) from the Laplacian operator discretized with a 3D 11-point stencil is reduced to $O(N^{3/3})$ when using the BLR format. All these points make BLR easy to adapt to any multifrontal solver without a complete rethinking of the code.

**NUMERICAL EXAMPLE**

We perform acoustic finite-difference frequency-domain seismic modeling in the 3D SEG/EAGE overthrust model (Aminzadeh et al., 1997) of dimension $20km \times 20km \times 4.65km$ with the 27-point mixed-grid finite-difference stencil (Operto et al., 2007) for the 2-Hz, 4-Hz and 8-Hz frequencies (Figures 4 and 5). Since we use a sequential prototype BLR solver, we perform the simulation on a single 64-core AMD Opteron node equipped with 384GB of shared memory. We use single precision arithmetic to perform the full-rank (FR) and the BLR factorizations. The point source is located at $x$ (dip) = 2 km, $y$
Applications of a new Block Low-Rank (BLR) algebraic sparse direct solver for frequency-domain seismic modeling. The computational time and memory savings achieved during BLR factorization increase with the size of the computational grid (i.e., frequency), suggesting that one order of magnitude of saving for these two metrics can be viewed for large-scale factorization involving several tens of millions of unknowns. Future work involves implementation of MPI parallelism in the BLR solver and a sensitivity analysis of FWI to the BLR approximation before application of visco-acoustic FWI on wide-azimuth data recorded with fixed-spread acquisition geometries (Brossier et al., 2013). The computational efficiency of the BLR solver might still be improved by iterative refinement of the solutions (although this refinement needs to be performed for each right-hand side) and/or by performing the BLR factorization in double precision. The computational savings provided by the BLR solver might allow to view frequency-domain seismic modeling in realistic 3D visco-elastic anisotropic media (Wang et al., 2012).

ACKNOWLEDGMENTS

This study was funded by the SEISCOPE consortium (http://seiscope.oca.eu) and by EDF under collaboration contract 8610-AAP-5910070284.
Table 1: Statistics of the Full-Rank (FR) and Block Low-Rank (BLR) simulations. $F(Hz)$: modeled frequency. *Flop count*: number of flops during LU factorization. *Mem LU*: Memory for LU factors in GigaBytes. *Peak of CB stack*: Maximum size of CB stack during LU factorization in Gigabytes. The metrics (flops and memory) for the low-rank factorization are provided as percentage of those required by the full-rank factorization (first 3 columns - top row).

<table>
<thead>
<tr>
<th>F (Hz)</th>
<th>Mem LU (FR)</th>
<th>Peak memory (FR)</th>
<th>Flop count (FR)</th>
<th>Mem LU (%) (BLR)</th>
<th>Peak of CB stack (%) (BLR)</th>
<th>Average storage CB (%) (BLR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>8.957E+11</td>
<td>3 GB</td>
<td>44.7</td>
<td>3 GB</td>
<td>4.50 GB</td>
<td>5.2 GB</td>
</tr>
<tr>
<td>4</td>
<td>1.639E+13</td>
<td>6 GB</td>
<td>5.769E+14</td>
<td>22 GB</td>
<td>1.40 GB</td>
<td>1.80 GB</td>
</tr>
<tr>
<td>8</td>
<td>5.769E+14</td>
<td>247 GB</td>
<td>4.5</td>
<td>445 GB</td>
<td>0.40 GB</td>
<td>0.50 GB</td>
</tr>
</tbody>
</table>

Figure 5: (a-c) Full-rank solution (real part). (a) 2 Hz, (b) 4 Hz, (c) 8 Hz. (d-f) Difference between full-rank and low-rank solutions for $\varepsilon = 10^{-3}$. (d) 2 Hz, (e) 4 Hz, (f) 8 Hz. (g-i) Same as (d-f) for $\varepsilon = 10^{-4}$. (j-l) Same as (d-f) for $\varepsilon = 10^{-5}$. Amplitudes are clipped to half the mean amplitude of the full-rank wavefield on each panel.

Figure 6: 2-Hz RTM image from full-rank (a) and low-rank (b-d) solutions. (b) $\varepsilon = 10^{-5}$, (c) $\varepsilon = 10^{-4}$, (d) $\varepsilon = 10^{-3}$. 
REFERENCES


