

Unsymmetric Greedy Orderings for Stable Sparse LU Factorization

Patrick R. Amestoy *

Xiaoye S. Li †

Stéphane Pralet ‡

We consider the **LU** factorization of a sparse unsymmetric matrix \mathbf{A} based on three-phase approaches (analysis, factorization, solve). The analysis phase transforms \mathbf{A} into a new matrix $\bar{\mathbf{A}}$ with better properties for sparse factorization. It exploits the structural information to reduce the fill-ins in the **LU** factors and exploits the numerical information to reduce the amount of numerical pivoting needed during factorization. Two consecutive treatments are commonly used to reach these two objectives. Firstly, scaling and maximum transversal algorithms are used to transform \mathbf{A} into \mathbf{A}_1 with large entries on the diagonal. Secondly, a symmetric fill-reducing ordering, which preserves the large diagonal, is used to permute \mathbf{A}_1 into \mathbf{A}_2 so that the factors of \mathbf{A}_2 are sparser than those of \mathbf{A}_1 . Note that during factorization, numerical instabilities can still occur and will be handled either by partial pivoting resulting in extra fill-ins in the factor matrices or by static pivoting resulting in a potentially less accurate factorization.

This approach has two drawbacks: (i) during analysis, the numerical treatment requires the fill-reducing ordering to limit its pivots choice to the diagonal of \mathbf{A}_1 , (ii) the ordering phase does not have numerical information to select the pivots.

We presented in [1] the basic ideas of a constrained unsymmetric greedy ordering method and showed some very preliminary results. The main features of our approach are:

- Based on a numerical pre-treatment of the matrix \mathbf{A} , we extract a set of numerically acceptable pivots, referred to as a **constraint matrix \mathbf{C}** (described by Algorithm 1).
- At each step k of the ordering, our pivot choice is not restricted to the diagonal of the matrix. Furthermore our choice is guided by both structural information given by the structure of the reduced matrix \mathbf{A}^k and numerical information given by the reduced constrained matrix \mathbf{C}^k (see Algorithm 2).

Algorithm 1: generic pre-processing phase (*NumThresh, StructThresh*)

Compute row and column scalings of \mathbf{A} , $\mathbf{A} \leftarrow \mathbf{D}_r \mathbf{A} \mathbf{D}_c$.

Build matrix \mathbf{C} :

- $\text{Pattern}(\mathbf{C}) = \{(i, j) \text{ st } |a_{ij}| > \text{NumThresh}\}$ and store numerical values in \mathbf{C} if needed,
- add entries from \mathbf{A} , s.t. maximum matching $\mathcal{M} \subset \mathbf{C}$,
- if needed suppress entries from \mathbf{C} (not in \mathcal{M}), until $|\mathbf{C}| < \text{StructThresh}$.

Algorithm 2: constrained unsymmetric ordering

while $k \leq n$

- Select best pivot in \mathbf{C}^k w.r.t metric($\mathbf{A}^k, \mathbf{C}^k$)
- $\mathbf{C}^{k+1} \leftarrow \text{Update}(\mathbf{C}^k)$
- $\mathbf{A}^{k+1} \leftarrow \text{Update}(\mathbf{A}^k)$
- Update metric values

end while

*Patrick.Amestoy@enseeiht.fr, ENSEEIHT-IRIT, 2, rue Camichel, BP 7122 - F 31071 Toulouse Cedex 7, France.

†xqli@lbl.gov, Lawrence Berkeley National Lab, MS 50F-1650, 1 Cyclotron Rd., Berkeley, CA 94720.

‡Stephane.Pralet@enseeiht.fr, CNRS UMR 5505, ENSEEIHT-IRIT, 2, rue Camichel, BP 7122 - F 31071 Toulouse Cedex 7, France.

To have an efficient implementation of Algorithm 2 we need modify some existing algorithms and develop some new ones. In particular, we need answer the following questions:

- (1) Which metric do we want to use at step (a) ?
- (2) Which data structure is suitable for reduced matrices \mathbf{A}^k ?
- (3) Which data structure is suitable for reduced matrices \mathbf{C}^k ?
- (4) How can we limit the cost of metric updates at step (d) ?

We will explain the algorithms that have been modified/adapted and the new algorithms that have been developed, and will show the consequences of these algorithmic choices. We will compare our new implementation which is referred to as Constrained Markowitz with Local Symmetrization (CMLS) with an earlier approach where the pivot choice is limited to the diagonal. The latter approach will be referred to as Diagonal Markowitz with Local Symmetrization (DMLS) [2]. We will focus on large matrices whose structural symmetry is lower than 0.5.

- To answer (1), we have developed several **hybrid metrics**. For example we can look for a large enough entry (i, j) in \mathbf{C}^k that minimizes the fill-in that will occur in \mathbf{A}^k if it were eliminated.
- To answer (4), we have developed approximation of the minimum fill and will explain our contribution to that. Our metrics have better tie-breaking properties and the other popular ordering algorithms such as AMD could benefit from them.
- We then briefly discuss our choices of the data structures for the reduced submatrices. To answer (3), a **weighted bipartite graph** is used to access the metric of each entry in \mathbf{C} . We sometimes perform **incomplete Gaussian elimination** to preserve the in-place property of our algorithms.
- To answer (2), a **bipartite quotient graph** is used to handle the matrix \mathbf{A} and **local symmetrization** [2] is applied to limit the complexity (search path of length at most three in the bipartite quotient graph). We study different ways of pruning the quotient graph structures and show that our pruning improves the reducibility detection. Moreover, as CMLS uses fully unsymmetric structures, our row and column supervariables are not correlated.

We will report results with the unsymmetrized multifrontal solver MA41_UNNS [3]. Our ordering is competitive with respect to DMLS since it requires approximately only 75% more time. In summary, for our test set, we have observed average gains of 14% for the factorization time, of 9% for the triangular solution time, of 12% for the memory usage, and of 12% for the sparsity of the factors. We also illustrate that our approach improves the accuracy of the static pivoting strategy of SuperLU_dist [4].

References

- [1] P. R. Amestoy, X. S. Li, and S. Pralet. Unsymmetric orderings using a constrained Markowitz scheme, February 27-28 2004. SIAM conference on Parallel Processing for Scientific Computing, Workshop of Combinatorial Scientific Computing, San Francisco (California).
- [2] P. R. Amestoy, X. S. Li, and E.G. Ng. Diagonal Markowitz scheme with local symmetrization. Technical Report RT/APO/03/5, ENSEEIHT-IRIT, October 2003. Also appeared as Lawrence Berkeley Lab report LBNL-53854.
- [3] P. R. Amestoy and C. Puglisi. An unsymmetrized multifrontal LU factorization. *SIAM Journal on Matrix Analysis and Applications*, 24:553–569, 2002.
- [4] X. S. Li and J. W. Demmel. SuperLU_DIST: A scalable distributed-memory sparse direct solver for unsymmetric linear systems. *ACM Transactions on Mathematical Software*, 29(2), 2003.