Ordering for low-rank compression

Grégoire Pichon, Bora Uçar & Frédéric Vivien

CNRS, INRIA, Université Lyon 1 & ENS Lyon

CR15: December 2022

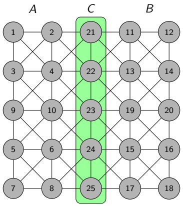
`gpichon.gitlabpages.inria.fr/m2if-numerical_algorithms/`

Outline

Ordering with Nested Dissection

Partition $V = A \cup B \cup C$

1. Order C with larger indices: $V_A < V_C$ and $V_B < V_C$
2. Apply the process recursively on A and B
3. Apply local heuristic such as AMF on small subgraphs



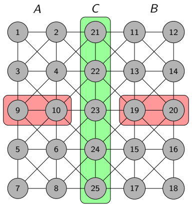
Nested dissection performed by an external partitioner tool

- Find a separator C as small as possible
- Balance subparts A and B

Ordering with Nested Dissection

Partition $V = A \cup B \cup C$

1. Order C with larger indices: $V_A < V_C$ and $V_B < V_C$
2. Apply the process recursively on A and B
3. Apply local heuristic such as AMF on small subgraphs



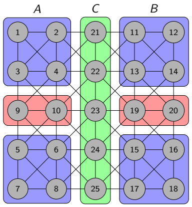
Nested dissection performed by an external partitioner tool

- Find a separator C as small as possible
- Balance subparts A and B

Ordering with Nested Dissection

Partition $V = A \cup B \cup C$

1. Order C with larger indices: $V_A < V_C$ and $V_B < V_C$
2. Apply the process recursively on A and B
3. Apply local heuristic such as AMF on small subgraphs



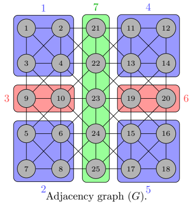
Nested dissection performed by an external partitioner tool

- Find a separator C as small as possible
- Balance subparts A and B

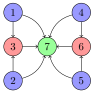
Block Symbolic Factorization

General approach

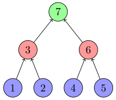
1. Build a partition with the nested dissection process
2. Compress information on data blocks
3. Compute the block elimination tree using the block quotient graph



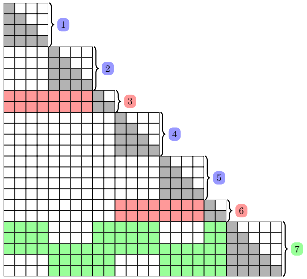
Adjacency graph (G).



Quotient graph $(G^*/P) = (G/P)^*$

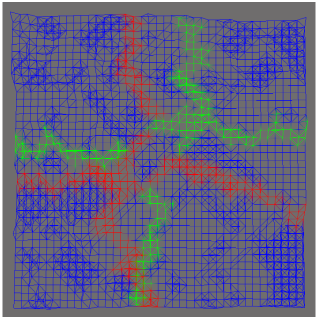


Elimination tree (T).



Factorized matrix (L).

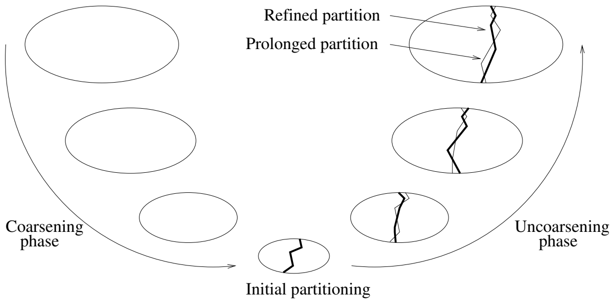
Partitioning tools: irregular separators



Non-smooth separators

- Blue: first separator of a regular 3D cube
- Green: interaction with second-level (2) separators
- Red: interaction with third-level (4) separators

Partitioning tools: multilevel approach



Approach

- Coarsening into a small graph
- Expensive heuristics on the small graph (greedy-graph growing)
- Uncoarsening with refinement (Fiduccia-Mattheyses)

Outline

Reordering problem

Existing approaches

- Ordering of separators on the local graph
- Reverse Cuthill-McKee performs a Breadth First Search to order unknowns
- It is not designed for direct solvers since matrices of separators will become full with the fill-in

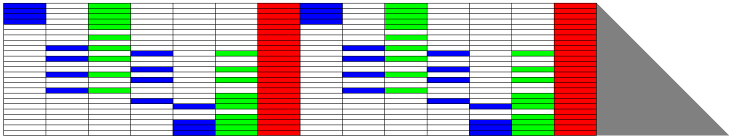
Proposition

- Ordering of separators to minimize the number of off-diagonal blocks
- Does not impact memory consumption or the number of operations
- Reduce the number of low-rank updates
- Increase granularity
    - Enhance the use of heterogeneous architectures (GPUs, Xeon Phi)
    - Reduce the overhead associated with runtime systems

Modeling of the problem

Proposition

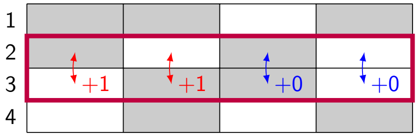
- Define a distance between rows: the number of differences between off-diagonal blocks
- Express the problem as a Traveling Salesman Problem (TSP) to sort rows in order to minimize the overall distance
- Use heuristics to perform TSP with low complexity



Modeling of the problem

Proposition

- Define a distance between rows: the number of differences between off-diagonal blocks
- Express the problem as a Traveling Salesman Problem (TSP) to sort rows in order to minimize the overall distance
- Use heuristics to perform TSP with low complexity

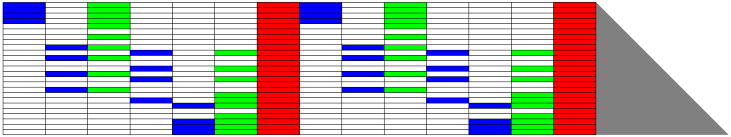


	1	2	3	4
1	0	-	-	-
2	3	0	-	-
3	3	2	0	-
4	1	4	2	0

Modeling of the problem

Notations for the ℓ^{th} diagonal block C_ℓ

- Contributing supernodes are included in $(C_k)_{k \in [1, \ell-1]}$
- We define w_i as the weight of row i and $d_{i,j}$ the distance between rows i and j



Modeling of the problem

Notations for the ℓ^{th} diagonal block C_ℓ

- Contributing supernodes are included in $(C_k)_{k\in[1,\ell-1]}$
- We define w_i as the weight of row i and $d_{i,j}$ the distance between rows i and j

Quality: Number of off-diagonal blocks

$$odb^\ell = \frac{1}{2}(w_1^\ell + \sum_{i=1}^{|C_\ell|-1} d_{i,i+1}^\ell + w_{|C_\ell|}^\ell)$$

Modeling of the problem

Notations for the ℓ^{th} diagonal block C_ℓ

- Contributing supernodes are included in $(C_k)_{k\in[1,\ell-1]}$
- We define w_i as the weight of row i and $d_{i,j}$ the distance between rows i and j

Quality: Number of off-diagonal blocks

$$odb^\ell = \frac{1}{2}(w_1^\ell + \sum_{i=1}^{|C_\ell|-1} d_{i,i+1}^\ell + w_{|C_\ell|}^\ell)$$

Optimal solution to minimize odb^ℓ

- Shortest Hamiltonian Path problem: find the shortest path visiting once each line, with a constraint on first and last line
- Complete symmetric graph: $d_{ij}^\ell = d_{ji}^\ell$ and $d_{ij}^\ell \leq d_{ik}^\ell + d_{kj}^\ell$

Proposition

Traveling Salesman Problem

- Find a cycle minimizing

$$\sum_{i=1}^{|C_\ell|} d^{\ell}_{i,(i+1)[|C_\ell|]}$$

- Add a fictive vertex S_0, without any contribution to build a cycle instead of a path

Algorithm: Insertion algorithm

1. Build the set B_i^ℓ for each line i of C^ℓ
2. Compute the distance matrix
3. Insert lines to minimize the cycle length
4. Split the cycle at the fictive vertex to get the path

Complexity

Results

- For graphs respecting a n^{σ}-separation theorem
- Numerical factorization in $\Theta(n^{3\sigma})$
- Reordering bounded by $\Theta(n^{\sigma+1})$

Type	σ	Reordering	Factorization
2D	$\frac{1}{2}$	$\Theta(n\sqrt{n})$	$\Theta(n\sqrt{n})$
3D	$\frac{2}{3}$	$\Theta(n^{\frac{5}{3}})$	$\Theta(n^{2})$

Table: Complexity for regular meshes

Asymptotically faster than the numerical factorization for $\sigma > \frac{1}{2}$
Remind that RCM is well working in 2D case

Resulting solution - Example

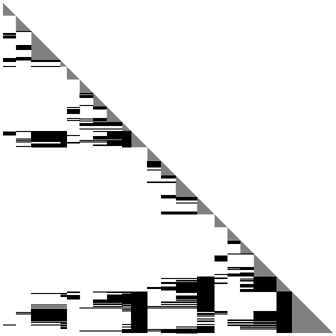


Figure: Without reordering (RCM)

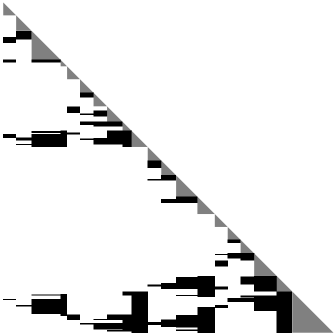


Figure: With reordering

Figure: Reordering on a $8 \times 8 \times 8$ Laplacian

Works with any initial seed

Experimental setup

Set of matrices

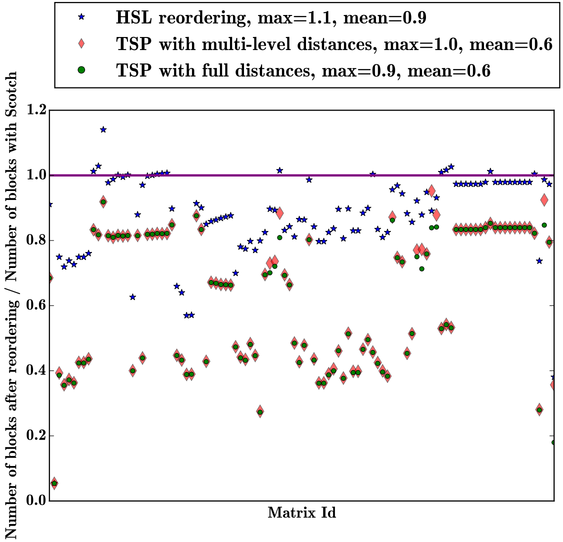
- 104 matrices issued from the SuiteSparse Matrix Collection
- Matrices with $500,000 \leq N \leq 10,000,000$
- Web and DNA matrices were removed

Strategy studied

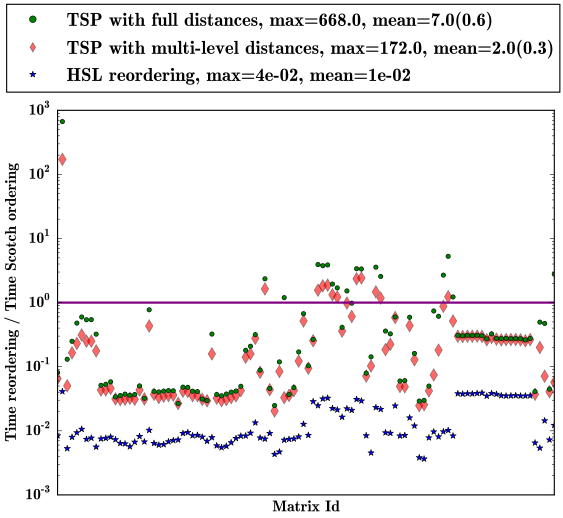
- TSP
- TSP with multi-level distances: heuristic to reduce the cost of TSP when computing distances between rows
- Reordering in HSL, developed at STFC

Another strategy introduced by M. Jacquelin et al. (2018) is faster than TSP while quality is close to TSP in most cases

Number of off-diagonal blocks



Reordering cost (sequential)



Summary

Results

- Reduce the number of off-diagonal blocks and thus the overhead associated with low-rank updates
- Lead to larger data blocks suitable for modern architectures
- Always increase performance wrt SCOTCH

Architecture	Nb. units	Mean gain	Max gain
Westmere	12 cores	2%	6%
Xeon E5	24 cores	7%	13%
Fermi	12 cores + 1 to 3 M2070	10%	20%
Kepler	24 cores + 1 to 4 K40	15%	40%
Xeon Phi	64 cores	20%	40%

Table: Performance gain for the full-rank factorization when using PARSEC runtime system with TSP instead of SCOTCH

Outline

Enhancing low-rank compression

What have we seen for now

- **When** to compress blocks
- How to perform operations on low-rank blocks
- We can reduce significantly execution time and/or memory consumption

Enhancing low-rank compression

- Some off-diagonal blocks are full-rank
- Some blocks are very well compressible

Objective: introduce some condition to define the potential compression (admissibility) of a block

Where does come compressibility ?

Mathematical property

- From some operators
- The ranks depend on the underlying operator

In practice

- Blocks that represent far-away interactions (in the geometry of the problem) are well compressible
- Exhibit clusters with a small diameter and few neighbours

Admissibility criteria

A widely used admissibility condition, named **strong block-admissibility** is defined as follows:

$$\sigma \times \tau \text{ is admissible} \iff \max(\text{diam}(\sigma), \text{diam}(\tau)) \leq \eta \text{ dist}(\sigma, \tau) \quad (1)$$

where η is a fixed parameter, diam() is the geometric diameter of a cluster and dist() the minimum distance between two clusters.

Another used admissibility condition named **weak admissibility** is less strict:

$$\sigma \times \tau \text{ is admissible} \iff \sigma \neq \tau. \quad (2)$$

With this last admissibility condition, only diagonal blocks (representing self-interaction) are not admissible.

Clustering technique: k-way partitioning (1/2)

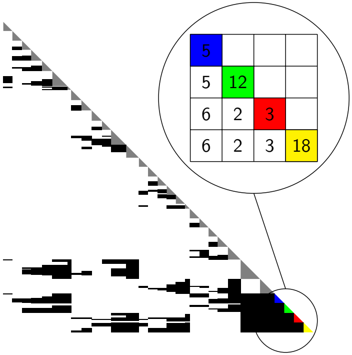
Objectives of a nice clustering

- Clusters with a small diameter
- Only a few neighbours

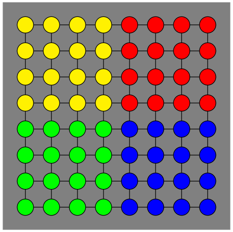
In practice

- Use k-way partitioning on the graph of each separator
- Eventually reconnect the graph before

Clustering technique: k-way partitioning (2/2)



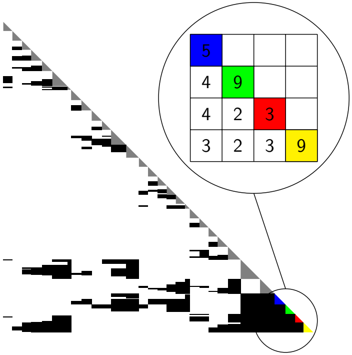
(a) Symbolic factorization



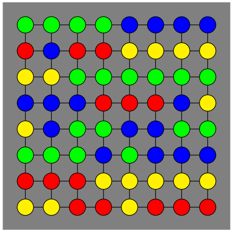
(b) First separator clustering

Figure: $8 \times 8 \times 8$ Laplacian partitioned using SCOTCH and k-way clustering on the first separator

Comparison with reordering + regular splitting



(a) Symbolic factorization



(b) First separator clustering

Figure: 8 × 8 × 8 Laplacian partitioned using SCOTCH and Reordering clustering on the first separator

Dense and full-rank blocks

Results coming from the clustering

- Diagonal blocks represent self-interaction: full-rank
- Neighbours from the clustering represent direct interactions: high rank
- Non-neighbours represent far-away interactions: low rank

Results

- $\Theta(1)$ full-rank blocks per column block
- The rank depends on the underlying operator, can be $\Theta(1)$ for an easy problem, $\Theta(\sqrt{n})$ for most problems or even $\Theta(n)$ for very difficult problems