MÉTHODES ET PROGRAMMATION NUMÉRIQUES AVANCÉES

Sparse Linear Algebra

Cédric Chevalier

December 10, 2024

Motivations

Linear Algebra and Numerical Simulation

solution of linear systems of equations$\rightarrow$key algorithmic kernel

Continuous problem

$\downarrow$

Discretization

$\downarrow$

Solution of a linear system $Ax = b$

Main parameters :

Numerical properties of the linear system (symmetry, pos. definite, conditioning, . . .)

Size and structure :

- Large ($>$ 1000000 $\times$ 1000000 ?), square/rectangular
- Dense or sparse (structured / unstructured)
- Target computer (sequential/parallel)

$\rightarrow$Algorithmic choices are critical

Designing efficient Algorithms

- Time-critical applications
- Solve larger problems
- Decrease elapsed time (parallelism ?)
- Minimize cost of computations (time, memory)

Challenges

Access to data :

- Computer : complex memory hierarchy (registers, multilevel cache, main memory (shared or distributed), disk)
- Sparse matrix : large irregular dynamic data structures.

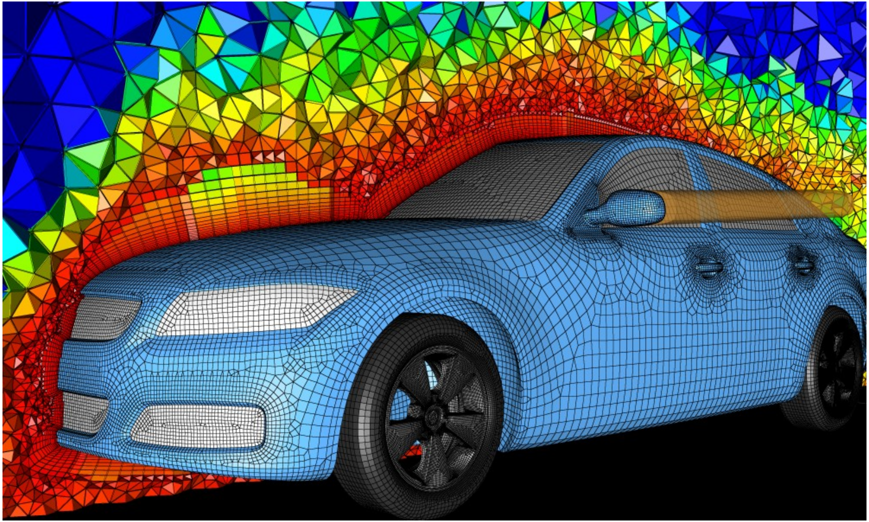
$\rightarrow$ Exploit the locality of references to data on the computer (design algorithms providing such locality)

Efficiency (time and memory)

- Number of operations and memory depend very much on the algorithm used and on the numerical and structural properties of the problem.
- The algorithm depends on the target computer (vector, scalar, shared, distributed, ...)

$\rightarrow$ Algorithmic choices are critical

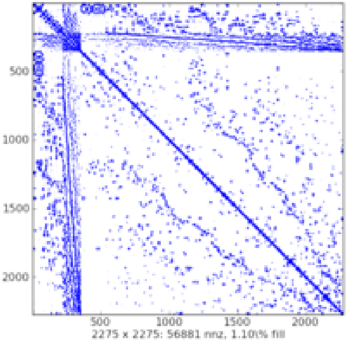
A Mesh

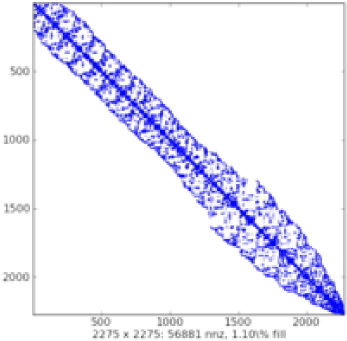


Sparse Matrix

A matrix $A(n \times n)$ where most of the elements are null. Non-null elements are called non-zeros.

$$nnz(A) = O(n)$$





Sparse Matrix Storage

- Explicit storage of zeros is too expensive
- Need for an *ad hoc* solution

Different solutions, depend on the problem

- Coordinates (COO)
- Compressed Storage Row (CSR) or Column (CSR)
- Diagonal
- Skyline Storage (SKS)
- ELLPACK