A Multi-level Nonlinear Domain Decomposition Solver for the Analysis of Large Aerostructures with Local Nonlinearities

Prepared and presented by *Philippe CRESTA, EADS Innovation Works*

1st EASN Workshop on Aerostructures

Suresnes, France

October 7-8, 2010

MAAXIMUS: More Affordable Aircraft through eXtended, Integrated and Mature nUmerical Sizing.

Maaximus Consortium ***:*** **57 Partners**, **18 Countries :**

- Aircraft Manufacturers and Software providers
- Test and R&T Centres
- Academic Institutions

The project budget is **67 M€** and the project duration is **5 years** (2008-2013).
It is funded by the European Commission through FP7 and led by **AIRBUS**.

MAAXIMUS is organized in **a Virtual and a Physical Platform** that are developed in a **fully synchronized manner,** to ensure **coherency** between their specific and dependent achievements**.**

MAAXIMUS Objectives

Faster Development

- Reduce by 20% the current development timeframe of aircraft composite structures from preliminary design up to full-scale test
- Reduce by 10% the non-recurring cost of aircraft composite structures from preliminary design up to full-scale test (ALCAS reference)

Right-First-Time Structure

- Reduce the airframe development costs by 5% compared with the equivalent development steps in an industrial context

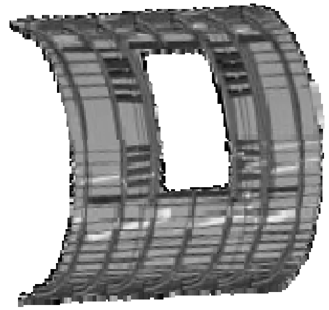
Highly-Optimised Composite Fuselage

- Enable a high-production rate: 50% reduction of the assembly time of fuselage section
- Reduce the manufacturing/assembly recurring costs by 10% compared to the ALCAS equivalent reference
- Reducing weight by 10%, compared to best available solutions on similar fuselage sections (F7X, A320 and TANGO fuselage)

Challenges

Maaximus Physical Platform

The MAAXIMUS Physical Platform will allow the development and validation of the appropriate composite technologies for low-weight and high production rate aircraft. It will deliver a new set of manufacturing and assembly capabilities, balancing weight - cost - production rate criteria.

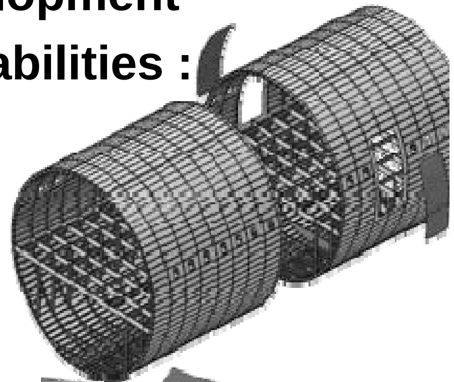


- Highly integrated sub-components and optimised subcomponent assembly, with test demonstration of structural details and panels
- Automatic section measurements for right-first-time best-fit positioning and assembly
- Manufacturing of one one-shot fuselage section
- In-depth testing of the fuselage barrel

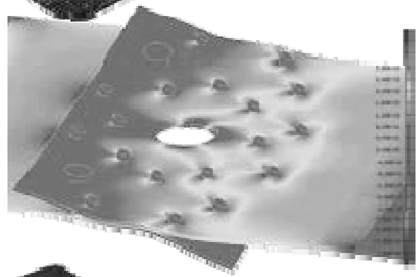
Challenges

Maaximus Virtual Platform

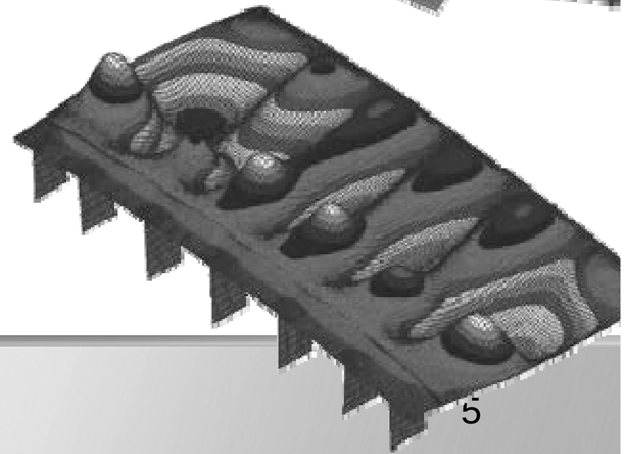
MAAXIMUS will deliver a multi-skill framework for the fast development of composite airframes, with a wide spectrum of advanced capabilities :



- Predictive capabilities from material properties and allowables of structural details and panels behaviour up to full-scale barrel behaviour
- Multi-objective and multi-scale design optimization capabilities
- Innovative composite structure optimisation enablers
- 10^9 Degrees of freedom solver capabilities in Non-Linear range
- Multi scale damage modelling techniques
- Close loop between design, analysis, manufacturing, NDI testing



The MAAXIMUS Virtual Platform will allow faster identification and earlier validation of the best solutions.



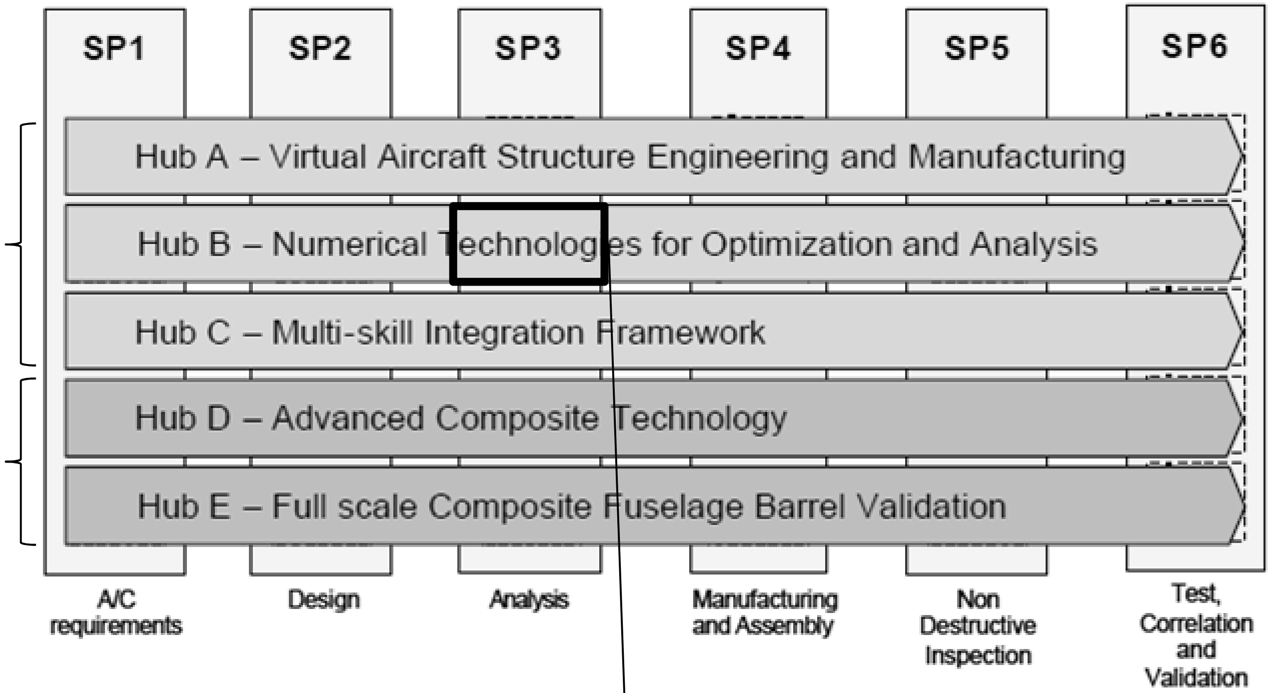
Project Organization

Subprojects (SP) **:**

Thematic hubs **:**

Virtual Platform

Physical Platform



<u>WP3.12</u>
Advanced Computational Mechanics Strategies

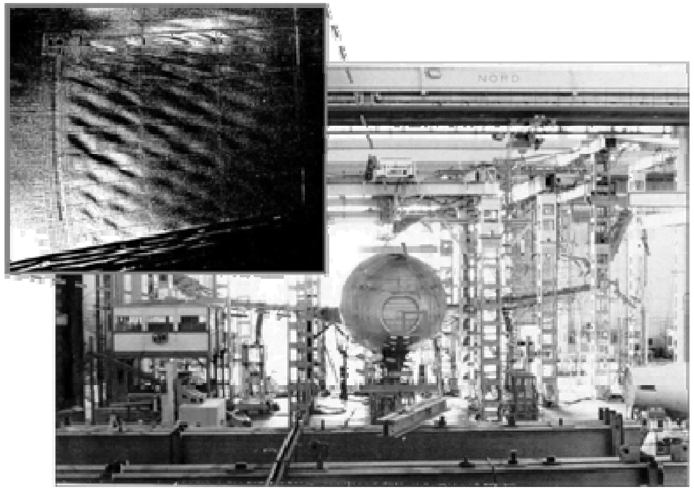
Task 3.12.12 : ***Domain Decomposition Methods for Nonlinear Problems***

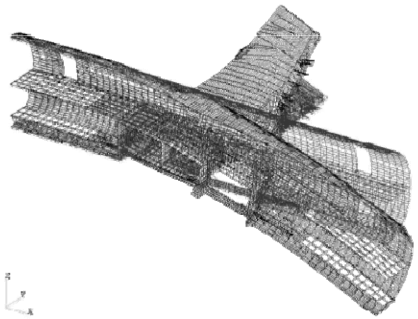
Challenge : Efficient Analysis of Large Nonlinear Problems

Complex Aerostructures

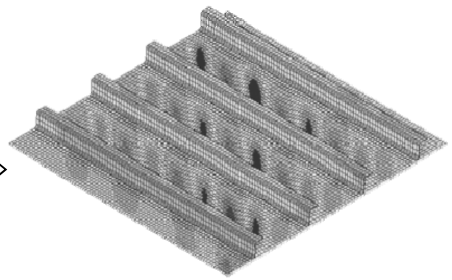
– **Very large structures with fine details *(several millions of d.o.f.)***

– **Complex *Nonlinear* behavior at both *local* and *global* scales (local buckling, damage, contact,...) with stress redistribution**





Global Coarse Analysis



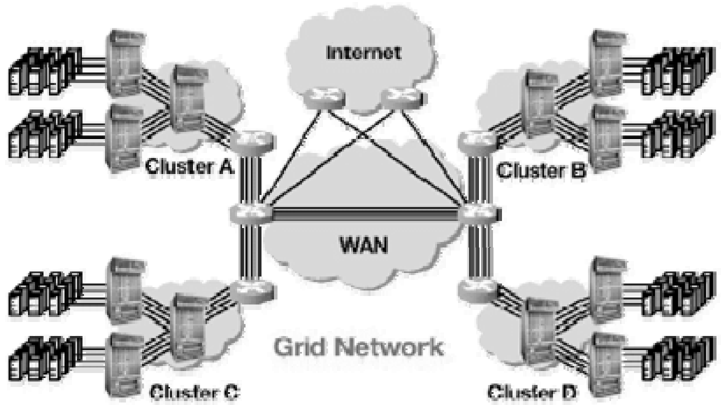
Local Nonlinear Analysis

– **Direct computations are unaffordable**

– **Classical global-local approaches suffer a lack of data exchange between models/scales *(one-way strategies, no data exchange)***

Industrial Needs in Computational Mechanics

- **Numerical strategies need to be adapted to new computational resources for HPC** :
clusters of workstations, distributed resources, etc…



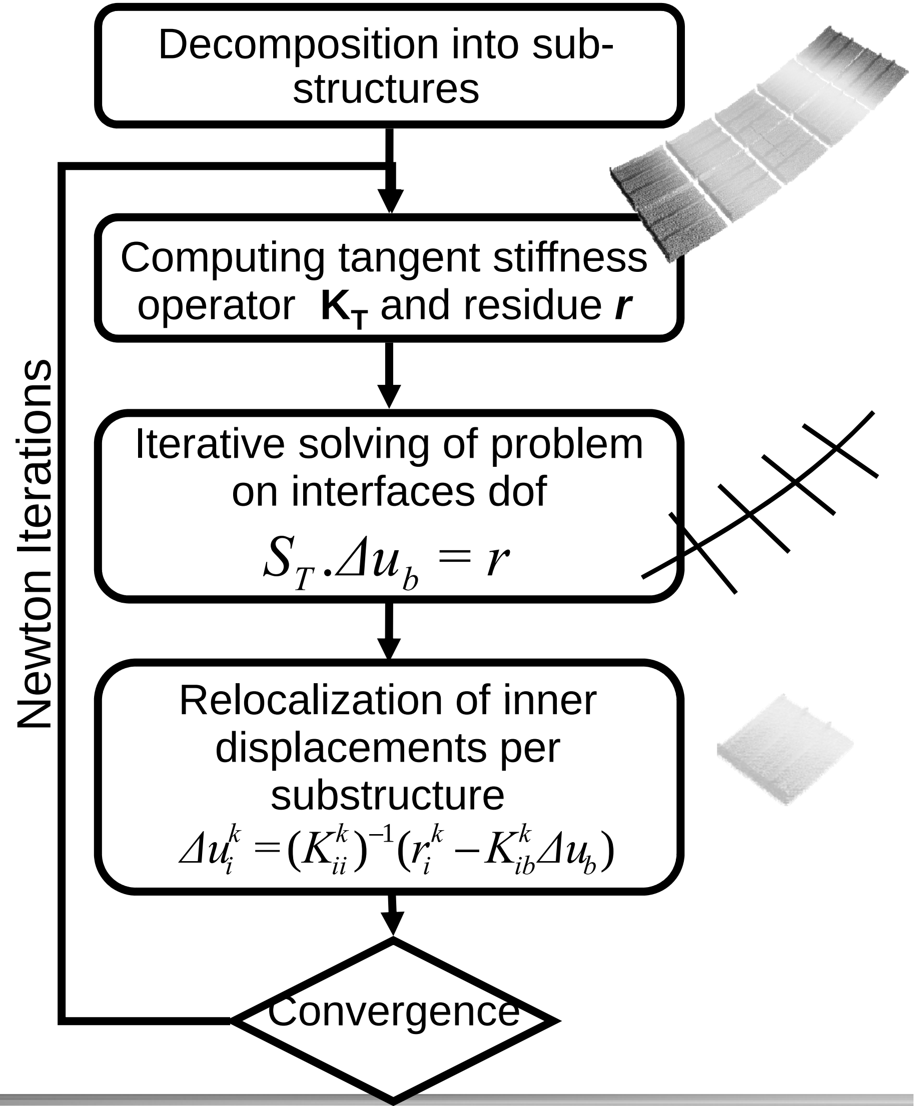
Propose & Validate new numerical strategies adapted to *large structures* with *non-linear* behavior at *different scales* (local buckling,…) through *multiscale* and *parallel* algorithms.

State-of-the-art in Large Nonlinear computations

Newton-Krylov-Schur methods

- **Classical DDM used within a Newton iteration procedure for each tangent linear problem (FETI, etc)**
- **Enable parallel computations of large nonlinear problems**

- ***Poor convergence*** for problems with ***unbalanced nonlinear*** effects : Global convergence is ***controlled by local phenomena***
- Very ***large number of iterations*** and global computations
- Performance penalized by ***data exchanges*** between substructures



Proposed Approach

Nonlinear Relocalization Strategy

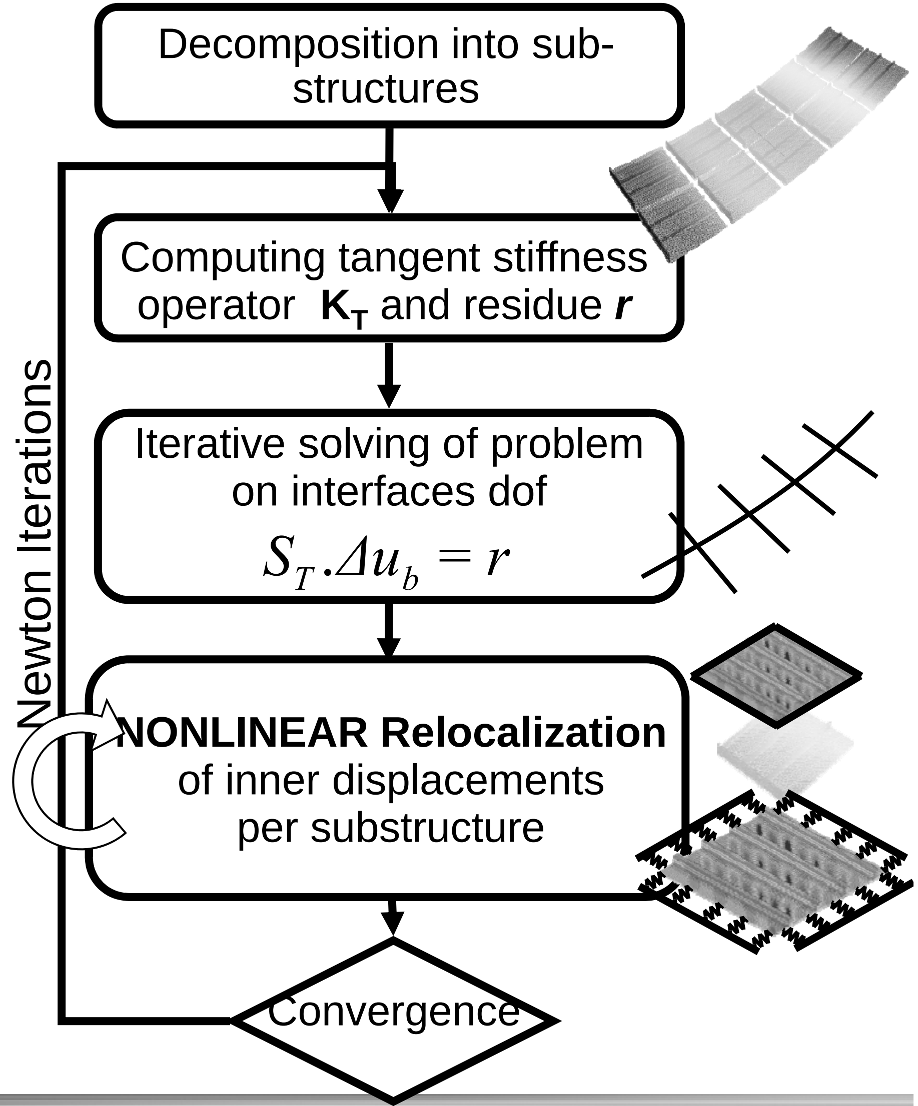
Targets :

- ✓ ***Decouple local nonlinear computations***
- ✓ **Ensure computational efforts *where they are really needed***
- ✓ **Treat non-linear buckling mechanisms *at the right scale***
- ✓ **Iterate at the global scale to ensure *admissibility* of global solution**

Question : ***Which data to transfer between local and global scales ?***

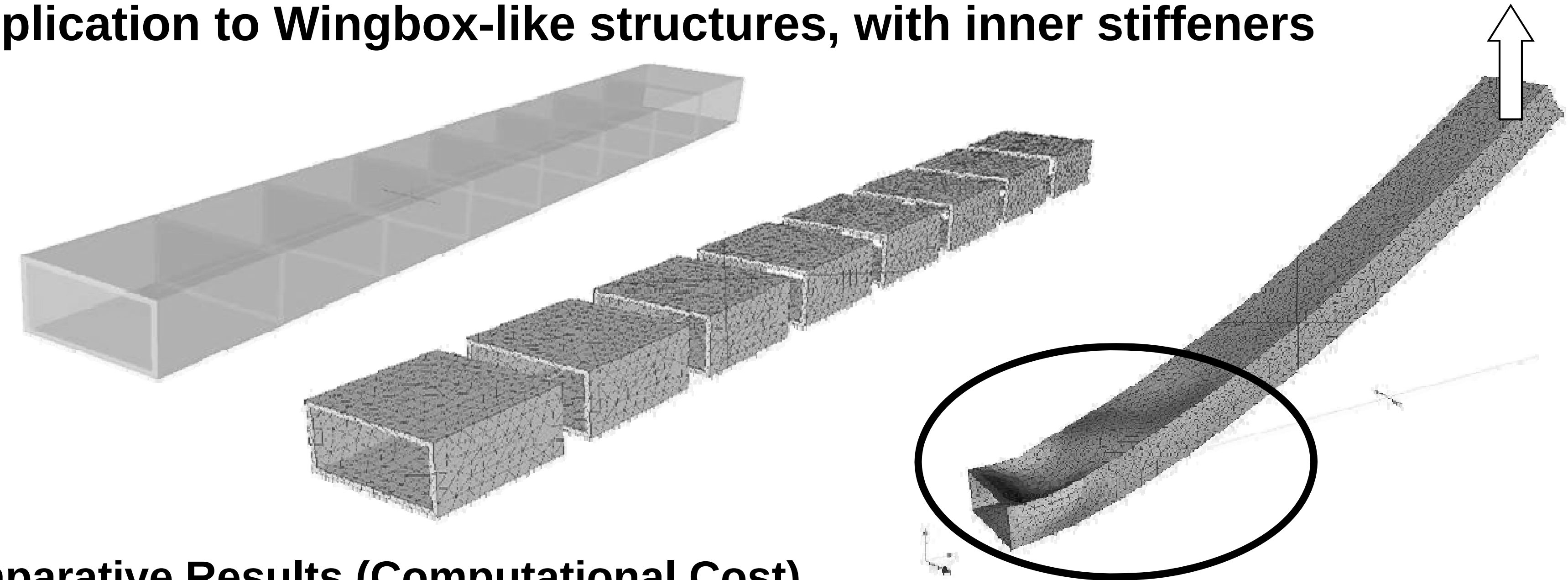
1- *First version* : *Primal* approach
Dirichlet boundary conditions on interfaces

2- *Second version* : *Mixed* approach
LaTIn-type iterations [Ladevèze, 1987]

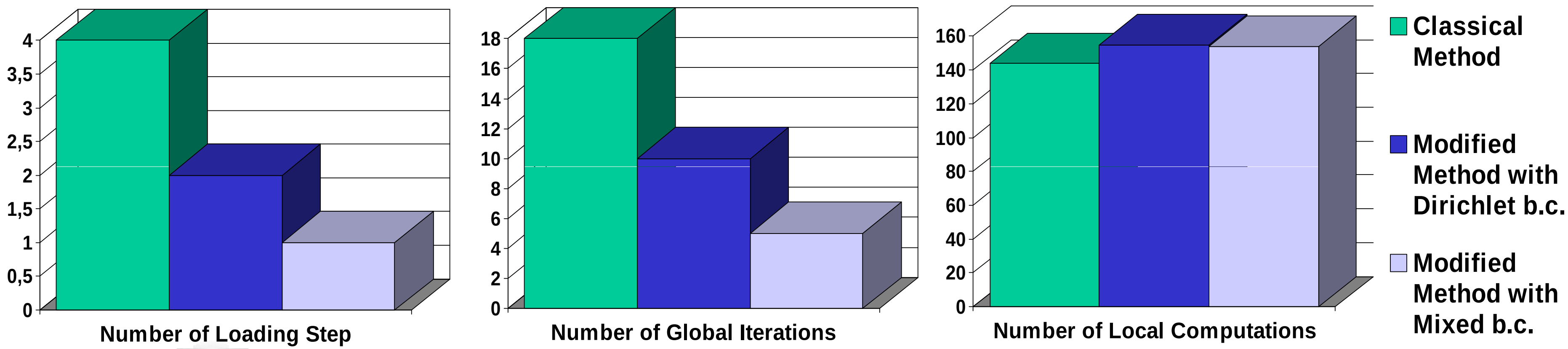


Past Results
C++ Research Platform

Application to Wingbox-like structures, with inner stiffeners



Comparative Results (Computational Cost)



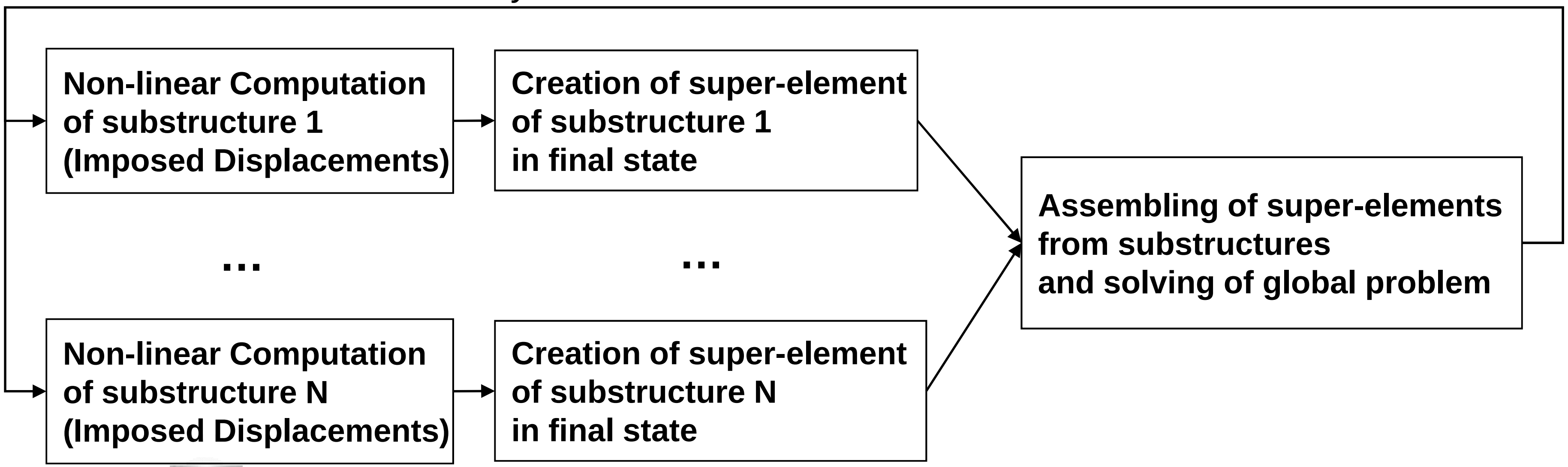
Implementation with Third-party FE solver

Towards a demonstrator on Industrial Cases

✓ Work with ***Abaqus*** solver, making use of Python scripting capabilities
✓ Enables computations of realistic structures with all structural FE features and solution techniques

Some difficulties

✓ No direct access to Domain Decomposition Methods, or any internal computational strategy
→ ***make use of other global solver (super-elements, …)***
✓ No built-in mixed boundary conditions



Implementation with Third-party FE solver

Technical details

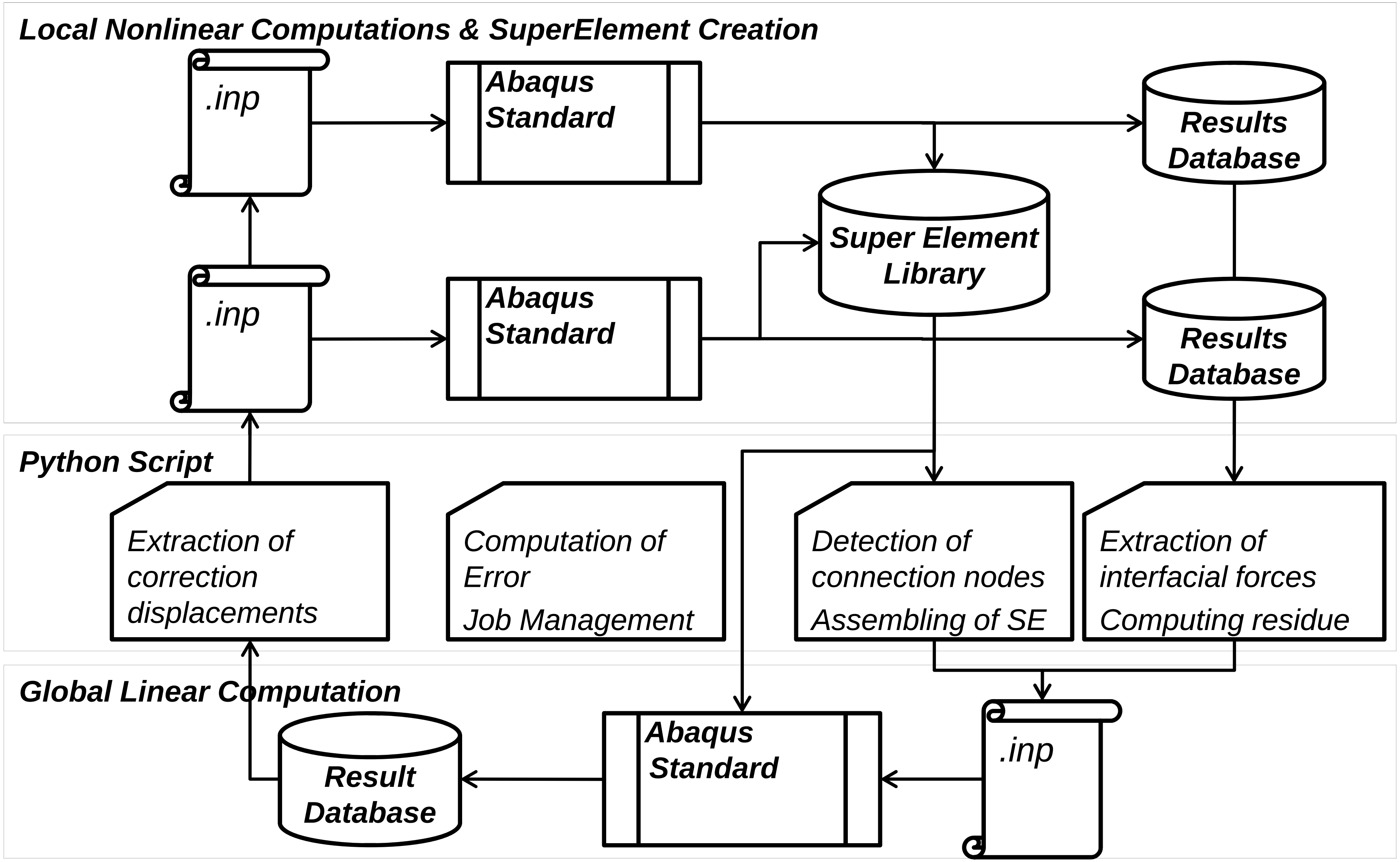
-**Two types of Abaqus computations :**

- **Local (per substrucutre) nonlinear quasi-static computations (`*Step, *Static`) with prescribed displacements, followed by Superelement generation (`*Step, *Substructure Generate`).**
    - ***Inputs :*** - prescribed displacement on boundaries,
      - if applicable, last solution from previous local computation
    - ***Outputs*** : - displacement solution (odb file) and forces on retained nodes
      - super-element library files with condensated linearized operator

- **Global condensated linear problem (`*Step, perturbation, *Static`)**
    - ***Inputs :*** - super-element library files from all local nonlinear computations
      - interface loading based on residue
    - ***Outputs :*** - displacement solution (odb file)

- **A Python script offers following fonctionnalities :**
    - writes / modifies Abaqus input files (boundary conditions, …)
    - launches and monitors Abaqus processes
    - read results from Abaqus Output Databases
    - compute residue, checks convergence, pilotes overall process

Implementation with Third-party FE solver

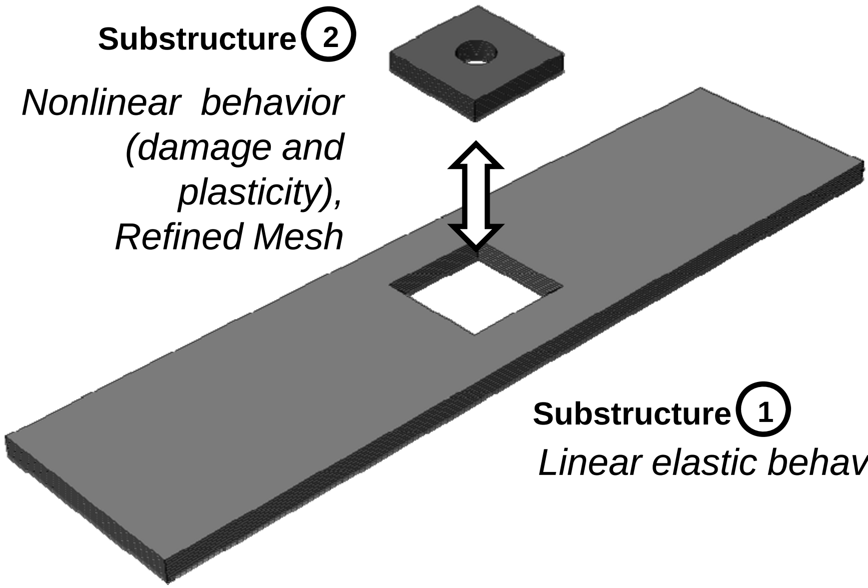


Composite plate with open hole, under traction

3D mesh, 12 plies, approx. 100.000 d.o.f.

Substructure 2

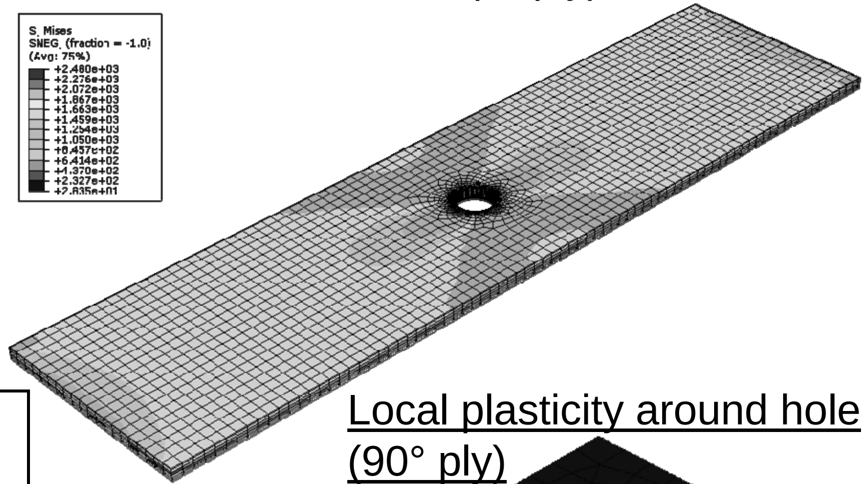
Nonlinear behavior (damage and plasticity), Refined Mesh



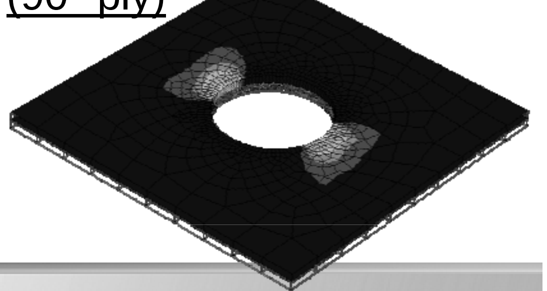
Substructure 1

Linear elastic behavior

Global Stress Solution (0° ply)



Local plasticity around hole (90° ply)



	Number of Global Iterations	Number of Local Computations	Total time (s)
Direct Newton	**38**	38+38=**76** 1 2	38*5.7s =**216s**
Multilevel Newton	**4**	4 +65=**69** 1 2	4*4s+69*1.7s=**133.3s**

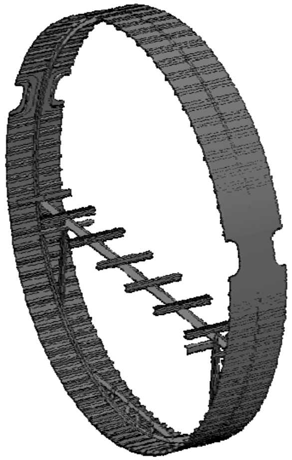
Ongoing & Future Work

Application to much larger structures

From a simple pattern …

650,000 nodes

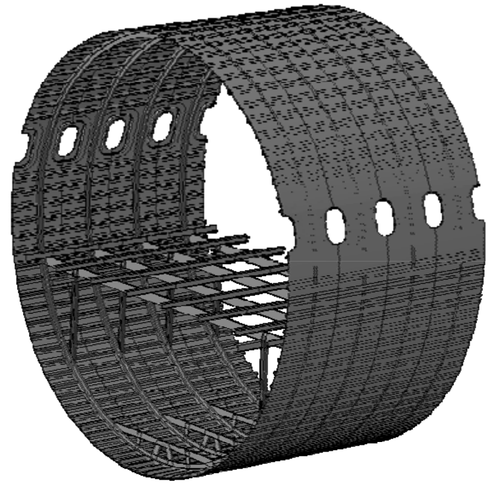
4 million d.o.f.



…to a fuselage barrel

2,600,000 nodes

16 million d.o.f. and more …



Work to be done :

- Adapt the strategy for use on clusters with batch system for jobs
- Identify bottleneck steps to improve scalability (access to files,…)
- Suggest new advanced features to software developers (more interactivity for the solver,…)
- Improve User Interface for Model Description

Conclusion

- **Proposal for a multilevel Newton scheme adapted to domain decomposition strategies and large structures with local nonlinear behaviour.**

- **Results show possible gain in overall simulation time by reducing the number of global iterations and treating nonlinear phenomena at the right scale**

- **Implementation within Third party FE software proves feasibility and enables application to real structures**

In coordination with improved performance parallel solvers,
this strategy offers a path to more efficient computations
for very large structures with many local nonlinear phenomena

Thank you for your attention

The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 under grant agreement n°213371.

www.maaximus.eu

Interface conditions

Nonlinear relocalization with Dirichlet (displacement) boundary conditions on interfaces

- **Global step**: **solving *linear* problem on interface dof**

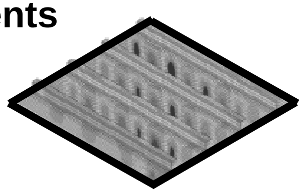
$$\mathbf{S}_T . \Delta u_b = \mathbf{r}$$

where

$$\begin{cases} \mathbf{S}_T = \mathbf{K}_{bb} - \sum_{k=1,2} \mathbf{K}_{bi}^k \mathbf{K}_{ii}^{k\,-1} \mathbf{K}_{ib}^k & \text{Schur complement} \\ \mathbf{r} = f_b - \sum_{k=1,2} \mathbf{K}_{bi}^k \mathbf{K}_{ii}^{k\,-1} f_i^k & \text{Condensated residue} \end{cases}$$

- **Local step**: **solving *nonlinear* problem per substructure, with imposed displacements**

$$\mathbf{K}_{ii}^k \Delta u_i^k = f_i^k - \mathbf{K}_{ib}^k \boxed{\Delta u_b}$$



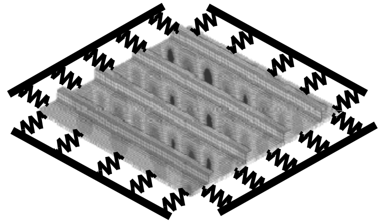
-**Advantages**: Easy to introduce in classical NKS methods. Only relocalization stage (per substructure) is modified. Classical DDM can be used.

-**Disadvantages**: Imposed displacements introduce artificial stiffness in interfaces, which can lead to divergences in the local nonlinear scheme.

Nonlinear relocalization mixed boundary conditions on interfaces

We chose here to work with both *Displacements* and *Forces* on interfaces.

- Local Step: non-linear computations with Robin mixed boundary conditions



$$\begin{cases} f = K(u).u \\ (F-\hat{F}) + \boxed{k}(U-\hat{U}) = 0 \end{cases}$$

($\hat{F}$ and $\hat{U}$ results from previous step)

Search Direction: parameter of the method

- Linear Step: admissibility of interface fields - *Equilibrium* et *Continuity*

Ensure on each interface :

$$\begin{cases} \hat{U}_1 = \hat{U}_2 \\ \hat{F}_1 + \hat{F}_2 = 0 \end{cases}$$

Search Direction :

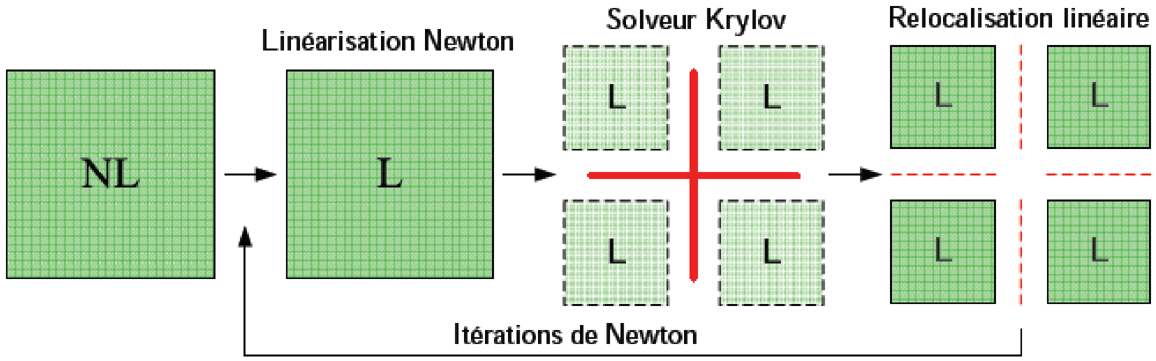
$$\boxed{S_T^{(s)}}(\hat{U}^{(s)} - U^{(s)}) = \hat{F}^{(s)} - F^{(s)}$$

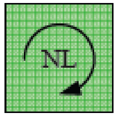
(F and U results from previous step)

Search Direction: Schur complement of tangent operator per substructure

$$\Rightarrow \begin{cases} \boxed{S_T^{global}}\hat{U} = \sum_s A^{(s)}(S_T^{(s)}U^{(s)} - F^{(s)}) \\ \hat{F}^{(s)} = F^{(s)} + S_T^{(s)}(\hat{U}^{(s)} - U^{(s)}) \end{cases}$$

Global problem close to the NKS one


Linéarisation Newton
Solveur Krylov
Relocalisation linéaire
NL
L
L
L
L
L
L
L
L
L
Itérations de Newton


NL