

INTALES



Preparation of branching analysis for complex nonlinear analysis, C++

Clemens Domanig

1st Workshop on Nonlinear Analysis of Shell Structures

Current status I

From the projects *ICONA* and *ACOSTA* we have:

- New knowledge in theorie on
 - New elements
 - Improved arc length method
 - Improved Branching analysis
 - ...

implemented and tested ...

INTALES



Current status II

INTALES

- ... in practice with several
 - MatLab-programs
 - Python-programs
 - C++-Programs

positiv:

+ well working

+ new knowledge tested

+ implemented with easy-to-use-programming-languages
(maybe except C++)



Current status III

INTALES

„negativ“:

- lots of small tools for different problems
- expert-tools and not for everyday-engineering
- some can only deal with small-scale problems
- poor performance (except C++)
- most of them use just one CPU (core)
- branching analysis has never been tested on large-scale problems



Our goal |

INTALES

Build our own FEM-tool that

- uses / combines all our (new) knowledge
- can deal with large-scale problems
- has good performance (at least)
- uses many CPUs (cores)
- is an easy-to-use program for everyday-industrial-engineering



Our goal II

INTALES

Build our own FEM-tool that

- can import and export different fileformats of other products
- serves as a platform to implement new ideas in short time
- gives us some / many ways indepence from other prodcuts which actually don't fulfill our needs



How we plan to do this ...

INTALES

Step I

- implementation of existing MatLab-programs in C++
- development of a prototype program

Step II

- stable and effective use of multi-core-hardware

Step III

- lets not talk about the future now



Some details I

INTALES

I Platform

- Linux – calculation and GUI
- Windows – just GUI

II Programming language

C++

- standard ★ lots of existing verified libraries
- fast
- easy to port ★ lots of projects to learn from



Some details II

INTALES

III Solver

Petsc

- good results by Andreas Grassl
- uses MPI
- Tested with more than $500 \cdot 10^6$ unknowns



IV Parallelisation

- joint venture with University of Innsbruck for many-core-compiler-project (Radu Prodan)



V GUI

- not decided yet

Status I

INTALES

What is already done?

- working prototype for nonlinear analysis
- two elements implemented
- program can read Abaqus-input-files
- implementation of Petsc-Library (first many-core-tests)



What comes next?

- Abaqus-odb-file as output-standard
- implementation of user-elements
- implementation of branching analysis



Status II

What else has to be done?

- Quite a lot, but with every step we improve.



INTALES



Questions / End

Thank you...



INTALES