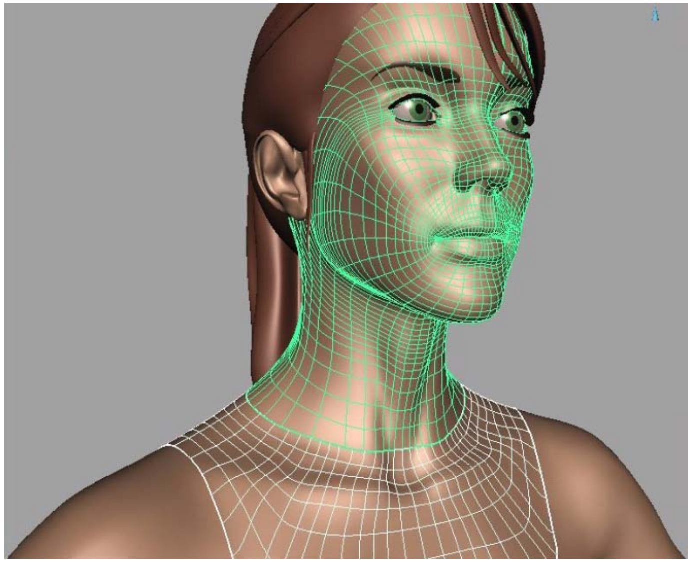
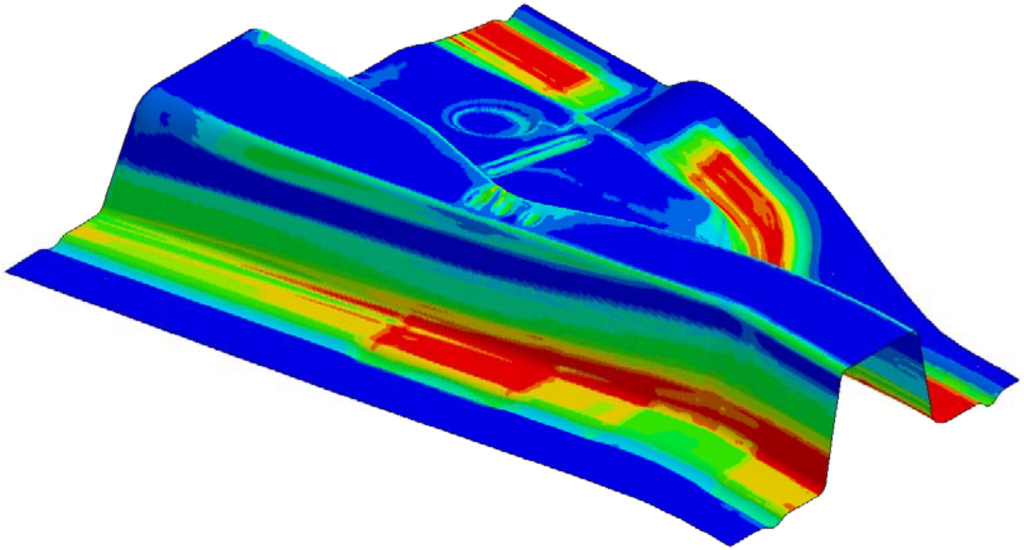
Introduction to Isogeometric Elements in LS-DYNA



T.J.R. Hughes



Entwicklerforum

October 12th, 2011, Stuttgart, Germany

Stefan Hartmann

Development in cooperation with:
D.J. Benson: Professor of Applied Mechanics, University of California, San Diego, USA

Some slides borrowed from:
T.J.R. Hughes: Professor of Aerospace Engineering and Engineering Mechanics, University of Texas at Austin, USA

Outline

- Isogeometric Analysis
  - motivation / definition / history
- From B-splines to NURBS (T.J.R. Hughes)
  - basis functions / control net / refinements
- NURBS-based finite elements in LS-DYNA
  - *ELEMENT_NURBS_PATCH_2D
- Example 1: Square Tube Buckling
  - description / deformation
- Example 2: Underbody Cross Member (Numisheet 2005)
  - description / comparison of results / summary
- Summary and Outlook

Isogeometric Analysis – motivation & definition

- reduce effort of geometry conversion from CAD into a suitable mesh for FEA

- ISOPARAMETRIC (FE-Analysis)
  use same approximation for geometry and deformation
  (normally: low order Lagrange polynomials ---- in LS-DYNA basically only linear elements)

GEOMETRY	←→	DEFORMATION

- ISOGEOMETRIC (CAD - FEA)
  same description of the geometry in the design (CAD) and the analysis (FEA)

CAD	←→	FEA

- common geometry descriptions in CAD
  - NURBS (Non-Uniform Rational B-splines) → most commonly used
  - T-splines → enhancement of NURBS
  - subdivision surfaces → mainly used in animation industry
  - and others

Isogeometric Analysis - history

- start in 2003
  - summer: Austin Cotrell starts as PhD Student of Prof. T.J.R. Hughes at the University of Texas, Austin
  - autumn: first NURBS-based FE-code for linear, static problems provides promising results, the name „ISOGEOMETRIC" is used the first time

- 2004 up to now: many research activities to various topics
  - non-linear structural mechanics
    - → shells with and without rotational DOFs
    - → implicit gradient enhanced damage
    - → XFEM
  - shape- und topology-optimization
  - efficient numerical integration
  - turbulence and fluid-structure-interaction
  - acoustics
  - refinement strategies
  - …

- January 2011: first thematic conference on *Isogeometric Analysis*
  - "Isogeometric Analysis 2011: Integrating Design and Analysis", University of Texas at Austin

From B-splines to NURBS

- B-spline basis functions
  - constructed recursively
  - positive everywhere (in contrast to Lagrange polynomials)
  - shape of basis functions depend on: knot-vector and polynomial degree
  - knot-vector: non-decreasing set of coordinates in parameter space
  - normally $C^{(P-1)}$-continuity
    - → e.g. lin. / quad. / cub. / quart. Lagrange: → C^0 / C^0 / C^0 / C^0
    - → e.g. lin. / quad. / cub. / quart. B-spline: → C^0 / C^1 / C^2 / C^3

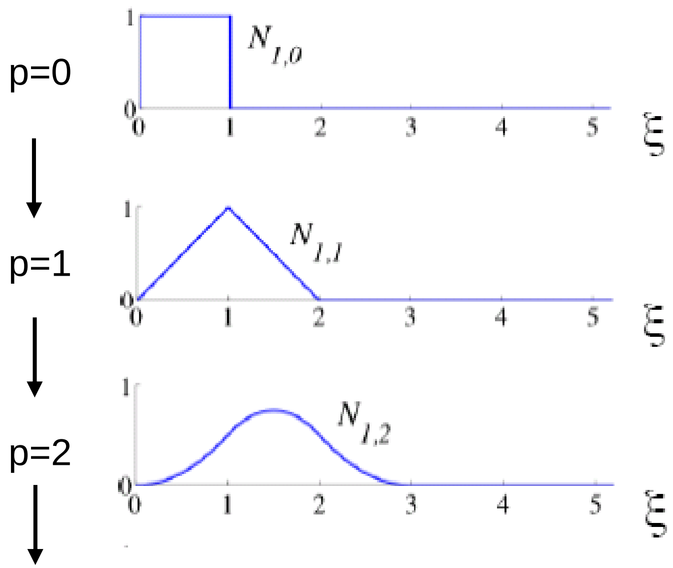
Example of a uniform knot-vector: $\Xi = \{0,1,2,3,4,\ldots\}$

$p = 0:$

$$N_{i,0}(\xi) = \begin{cases} 1 & if \;\; \xi_i \le \xi < \xi_{i+1} \\ 0 & otherwise \end{cases}$$

$p > 0:$

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1(\xi)} + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1(\xi)}$$



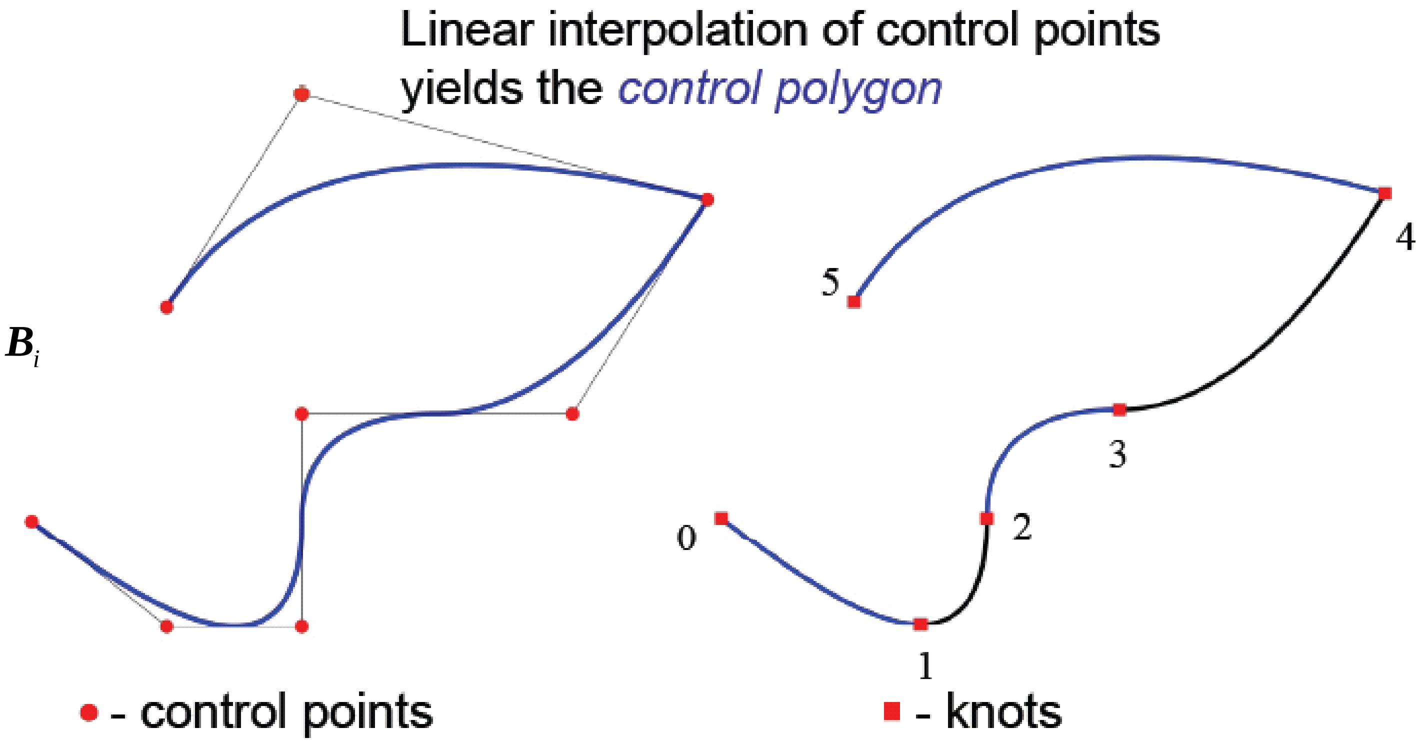
T.J.R. Hughes

From B-splines to NURBS

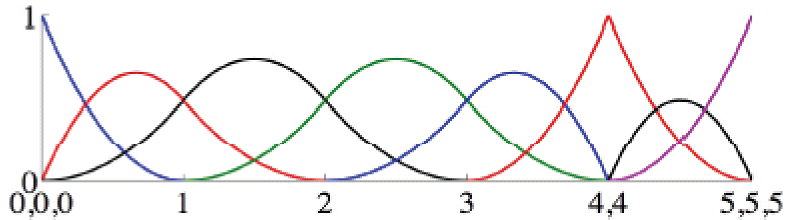
- B-spline curves
  - control points $\boldsymbol{B}_i$ / control polygon (control net)
  - knots

linear combination:

$$\boldsymbol{C}(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi)\, \boldsymbol{B}_i$$



Quadratic basis



T.J.R. Hughes

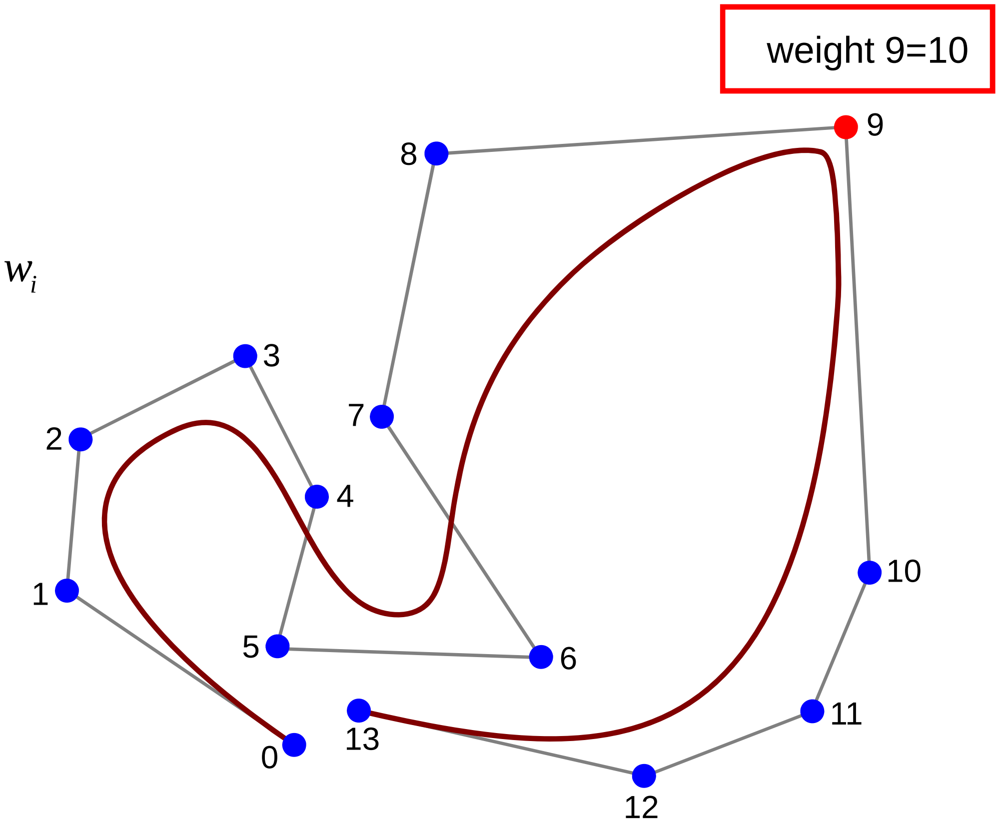
From B-splines to NURBS

- NURBS – Non-Uniform Rational B-splines
  - weights at control points leads to more control over the shape of a curve
  - projective transformation of a B-spline

$$R_i^p(\xi) = \frac{N_{i,p}(\xi)\, w_i}{W(\xi)}$$

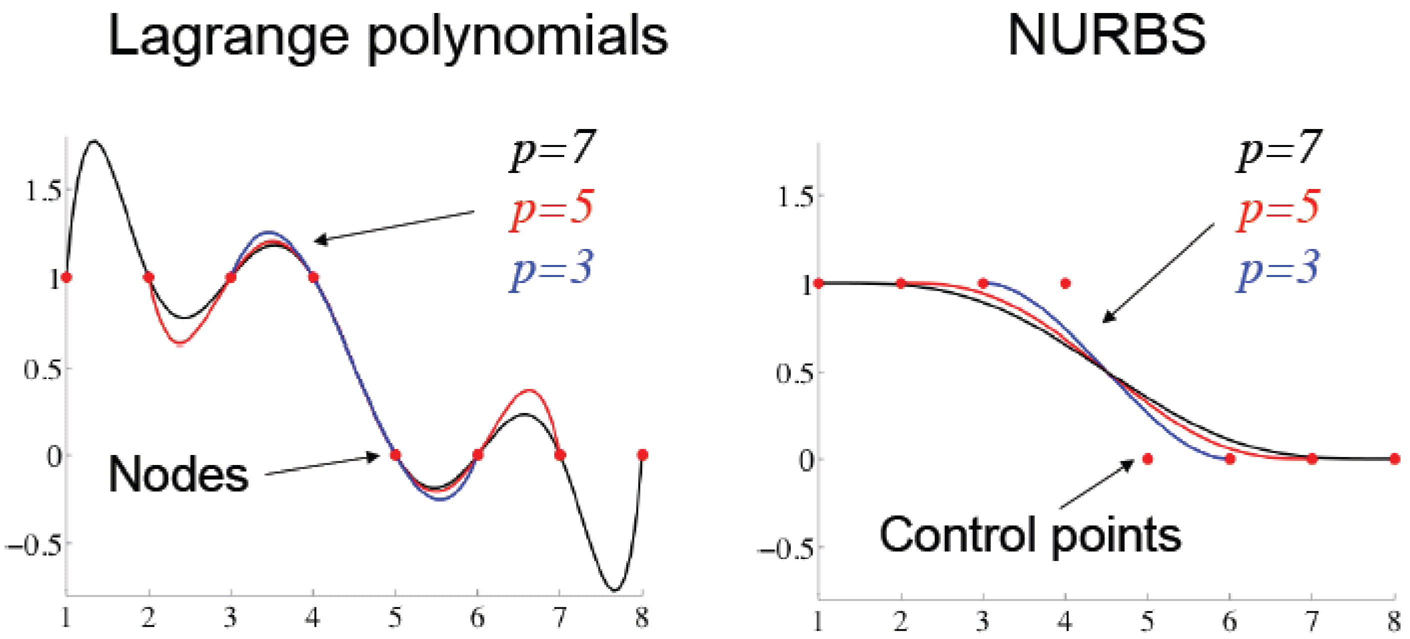
$$with: \; W(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi)\, w_i$$

$$\boldsymbol{C}(\xi) = \sum_{i=1}^{n} R_i^p(\xi)\, \boldsymbol{B}_i$$



From B-splines to NURBS

- Smoothness of Lagrange polynomials vs. NURBS



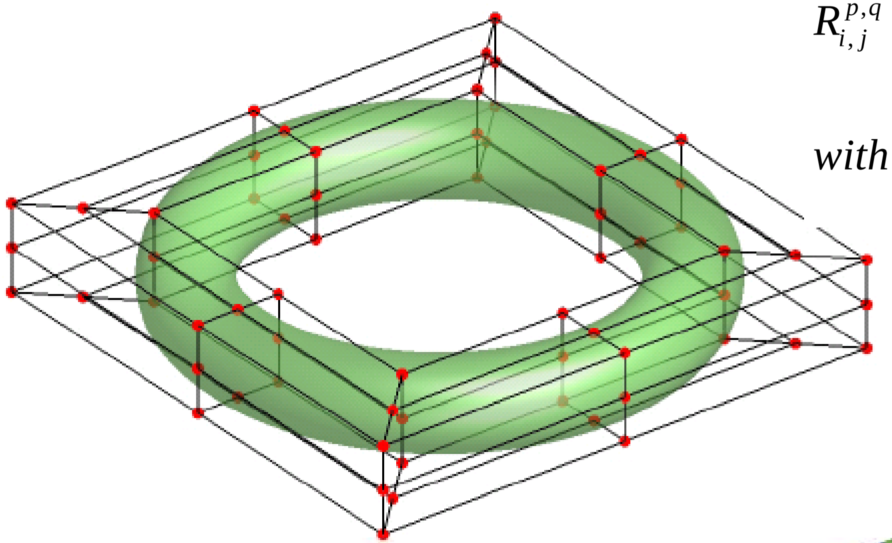
T.J.R. Hughes

From B-splines to NURBS

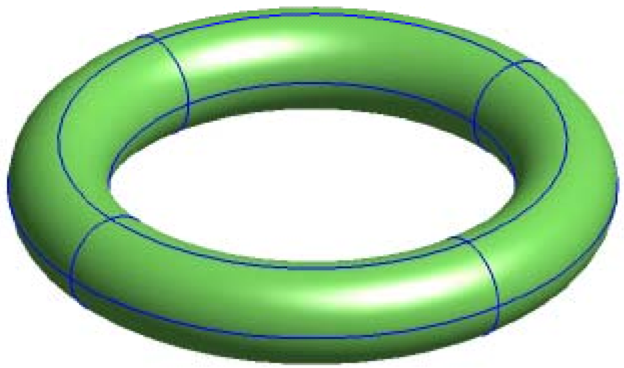
- NURBS – surfaces (tensor-product of univariate basis)

$$R_{i,j}^{p,q}\left(\xi,\eta\right)=\frac{N_{i,p}\left(\xi\right)M_{j,q}\left(\eta\right)w_{i,j}}{W\left(\xi,\eta\right)}$$

$$with: W\left(\xi,\eta\right)=\sum_{i=1}^{n}\sum_{j=1}^{m}N_{i,p}\left(\xi\right)M_{j,q}\left(\eta\right)w_{i,j}$$



Control net



Mesh

T.J.R. Hughes

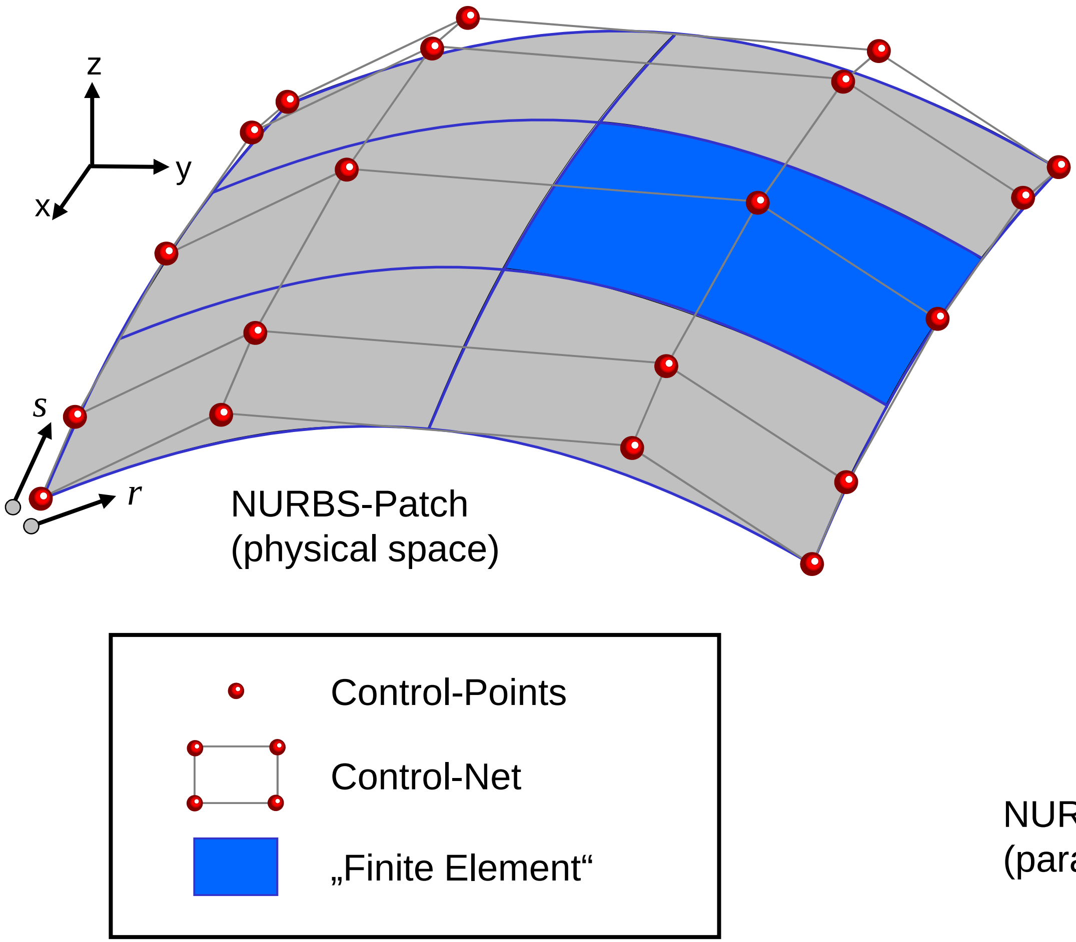
From B-splines to NURBS - summary

- B-spline basis functions
  - recursive
  - dependent on knot-vector and polynomial order
  - normally $C^{(P-1)}$-continuity
  - „partition of unity“ (like Lagrange polynomials)
  - refinement (h/p and k) without changing the initial geometry → adaptivity
  - control points are normally not a part of the physical geometry (non-interpolatory basis functions)

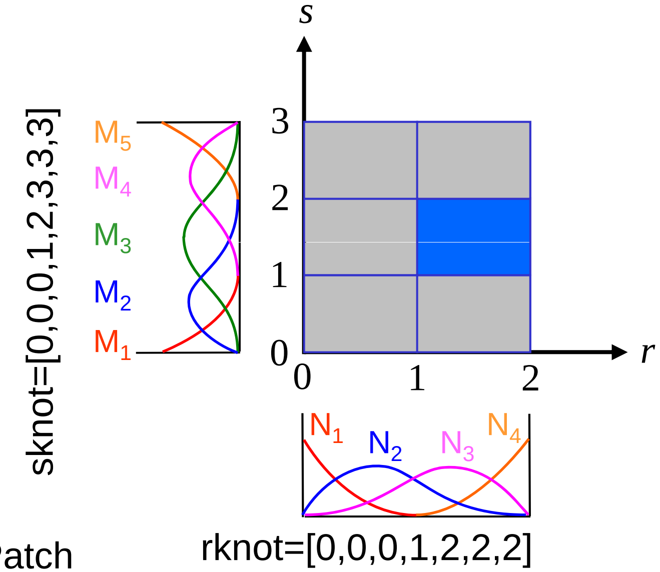
- NURBS
  - B-spline basis functions + control net with weights
  - all mentioned properties for B-splines apply for NURBS

NURBS-based finite elements in LS-DYNA

- A typical NURBS-Patch and the definition of elements
  - elements are defined through the knot-vectors (interval between different values)
  - shape functions for each control-point



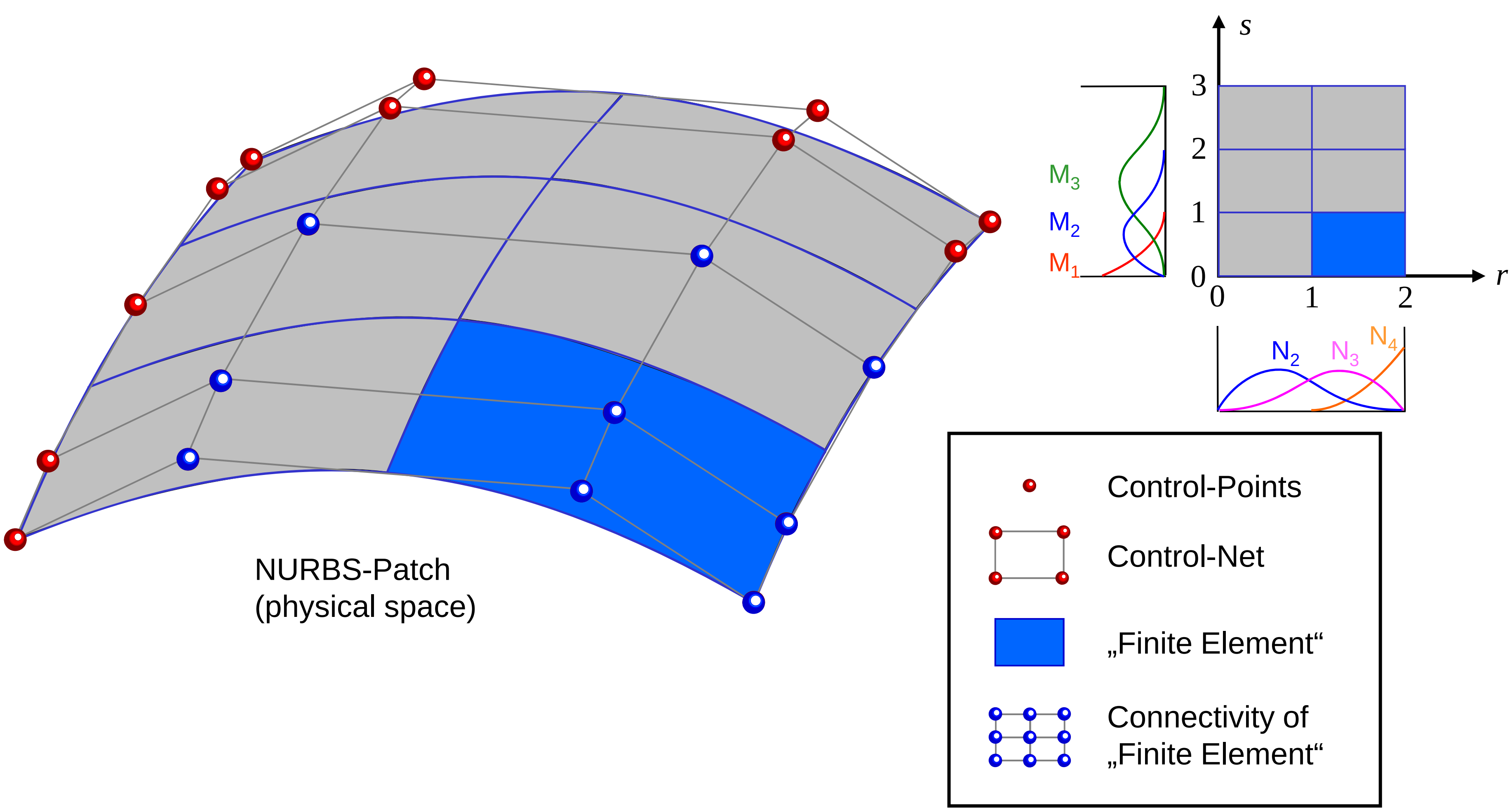
polynomial order:
- quadratic in r-direction (pr=2)
- quadratic in s-direction (ps=2)



NURBS-Patch
(parameter space)

NURBS-based finite elements in LS-DYNA

- A typical NURBS-Patch – Connectivity of elements
  - Possible „overlaps“ (→ higher continuity!)
  - Size of „overlap“ depends on polynomial order (and on knot-vector)



NURBS-based finite elements in LS-DYNA

- New Keyword: *ELEMENT_NURBS_PATCH_2D
  - definition of NURBS-surfaces
  - 4 different shell formulations with/without rotational DOFs

- Pre- and Postprocessing
  - work in progress for LS-PrePost … current status (lspp3.2beta)
    → visualization of 2D-NURBS-Patches
    → import IGES-format and construct *ELEMENT_NURBS_PATCH_2D
    → modification of 2D-NURBS geometry
    → … much more to come!

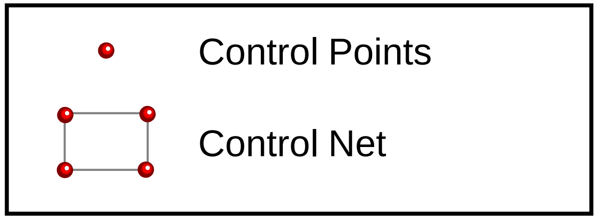
- Postprocessing and boundary conditions (i.e. contact) currently with
  - interpolation nodes (automatically created)
  - interpolation elements (automatically created)

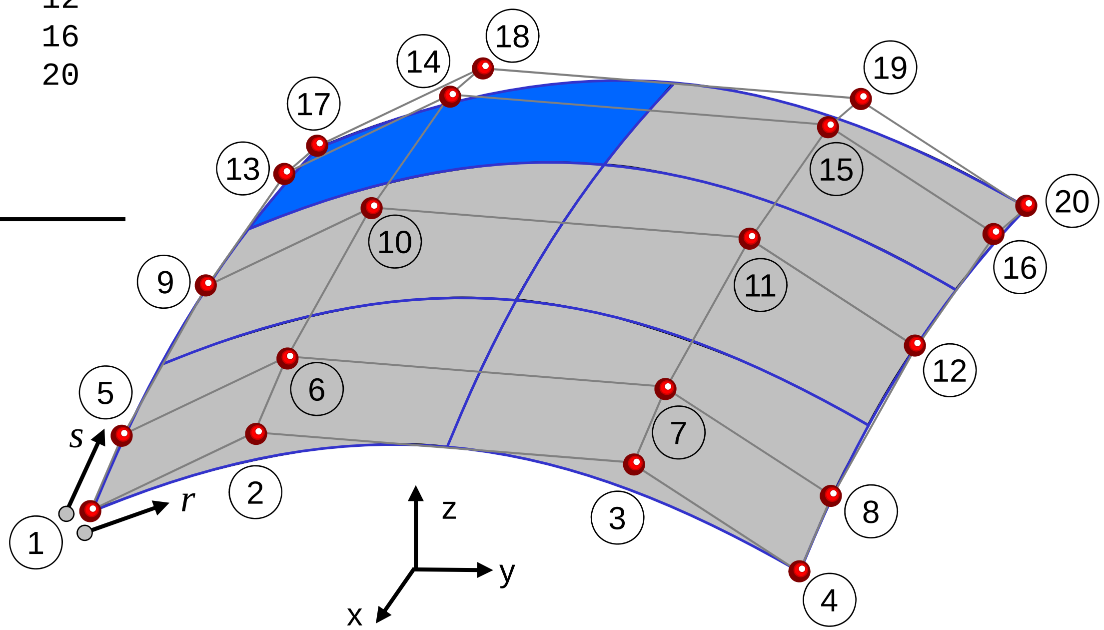
- Analysis capabilities
  - implicit and explicit time integration
  - eigenvalue analysis
  - other capabilities (e.g. geometric stiffness for buckling) implemented but not yet tested

- LS-DYNA material library available (including umats)

NURBS-based finite elements in LS-DYNA

```
*ELEMENT_NURBS_PATCH_2D
$---+--EID----+--PID----+--NPR----+---PR----+--NPS----+---PS----+----7----+----8
        11        12         4         2         5         2
$---+--WFL----+-FORM----+--INT----+-NISR----+-NISS----+IMASS----+----7----+----8
         0         0         1         2         2         0
$rk-+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
       0.0       0.0       0.0       1.0       2.0       2.0       2.0
$sk-+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
       0.0       0.0       0.0       1.0       2.0       3.0       3.0       3.0
$net+---N1----+---N2----+---N3----+---N4----+---N5----+---N6----+---N7----+---N8
         1         2         3         4
         5         6         7         8
         9        10        11        12
        13        14        15        16
        17        18        19        20
```

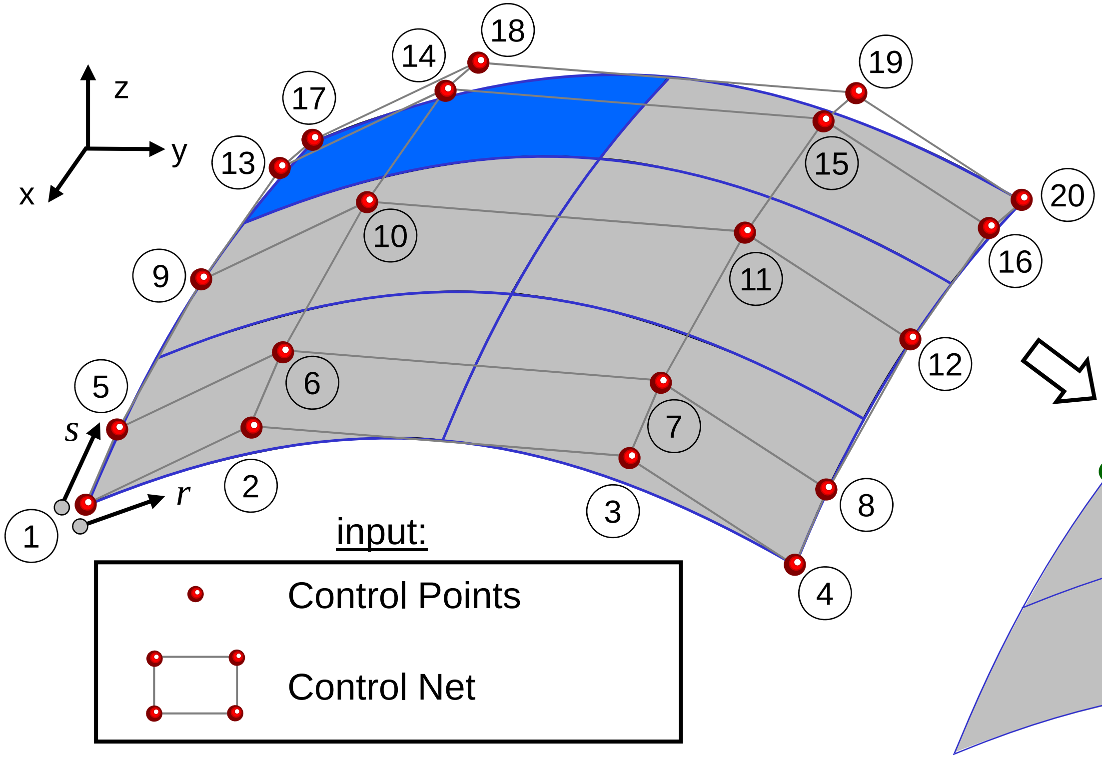


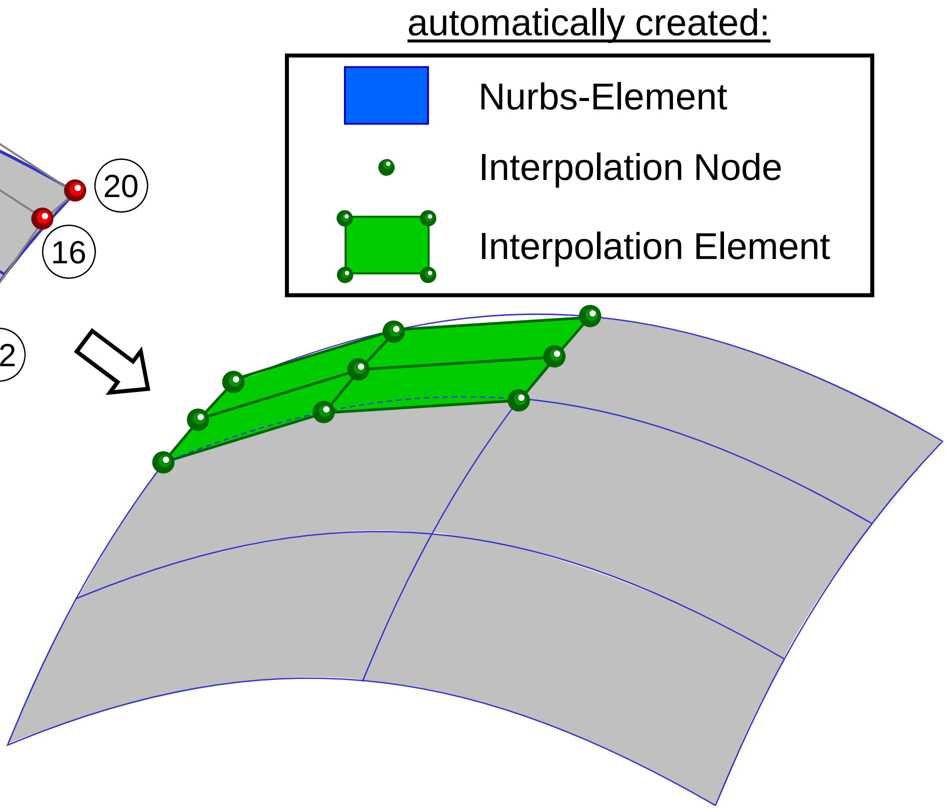


NURBS-based finite elements in LS-DYNA

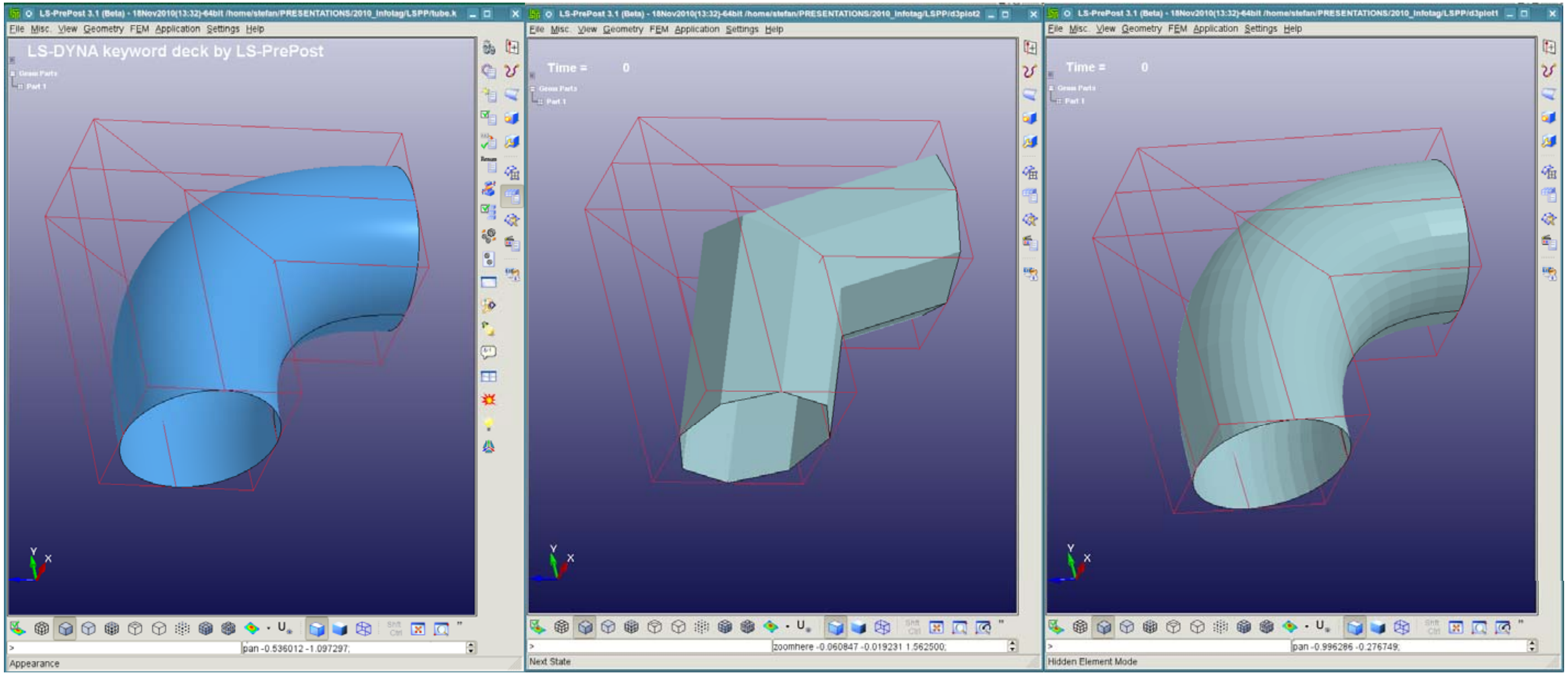
```
...
$---+--WFL----+-FORM----+--INT----+-NISR----+-NISS----+IMASS
         0         0         1         2         2         0
```

- NISR/NISS – Number of Interpolation Elements per Nurbs-Element (r-/s-dir.)
  important for post-processing, boundary conditions and contact treatment





NURBS-based finite elements in LS-DYNA



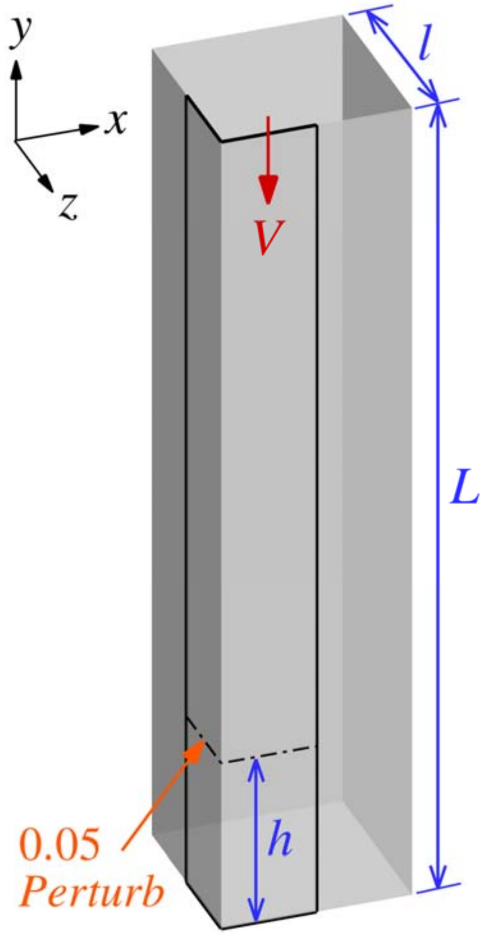
LSPP: Preprocessing
- control-net
- nurbs surface

LSPP: Postprocessing
- Interpolation nodes/elements

nisr=niss=2

nisr=niss=10

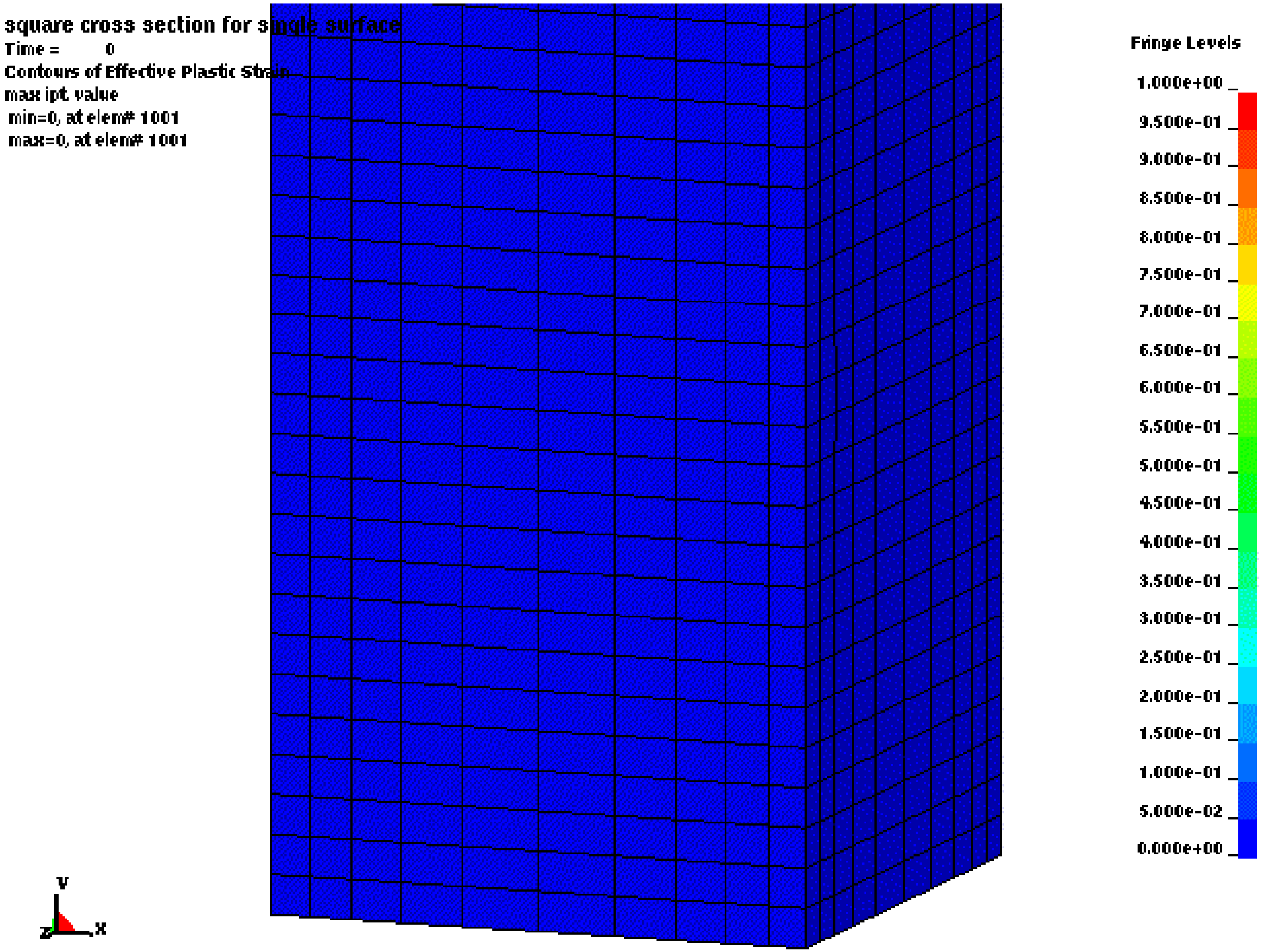
Square Tube Buckling – Description



D.J. Benson

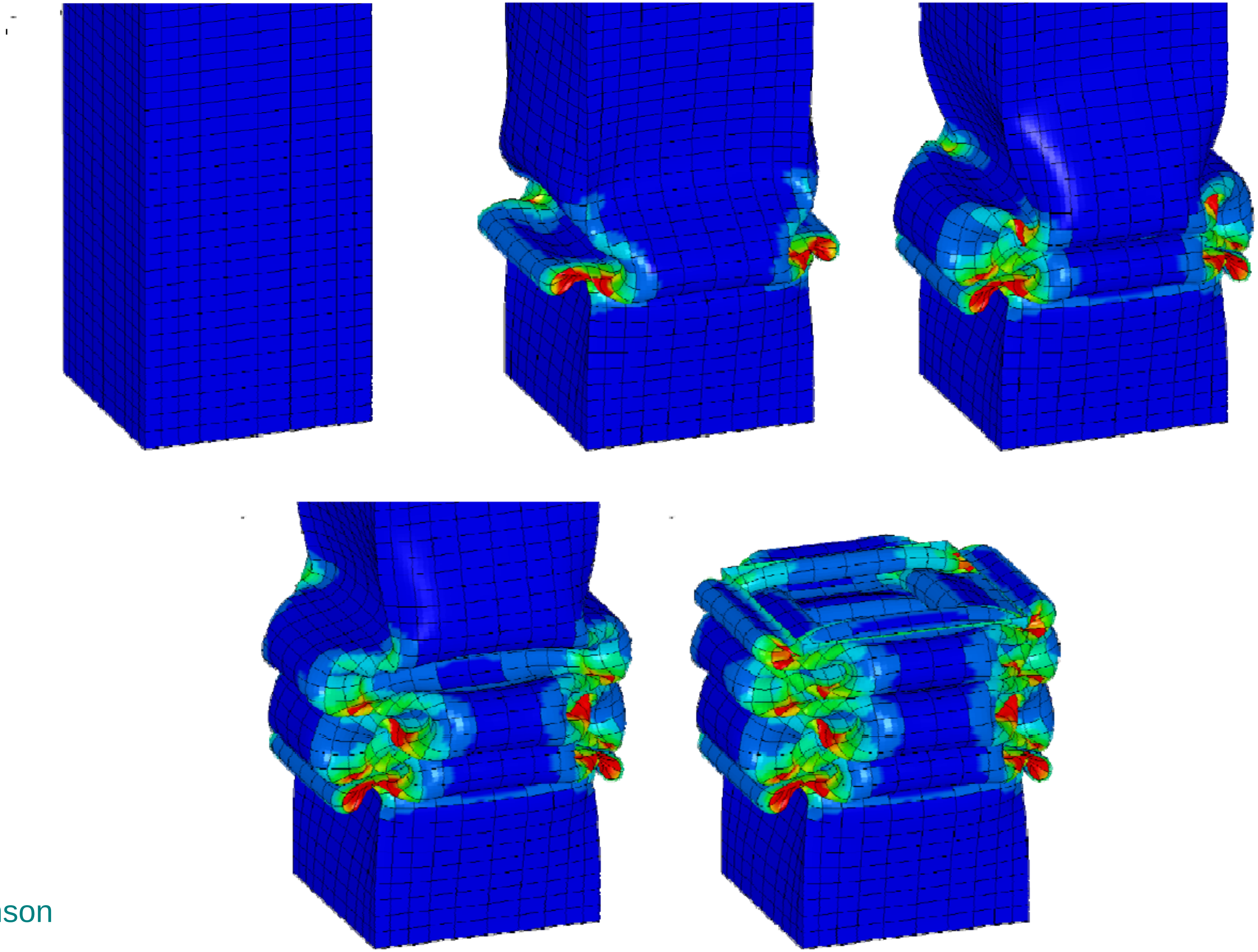
- standard benchmark for automobile crashworthiness
- quarter symmetry to reduce cost
- perturbation to initiate buckling mode
- J_2 plasticity with linear isotropic hardening
- mesh:
  - 640 quartic (P=4) elements
  - 1156 control points
  - 3 integration points through thickness

Square Tube Buckling



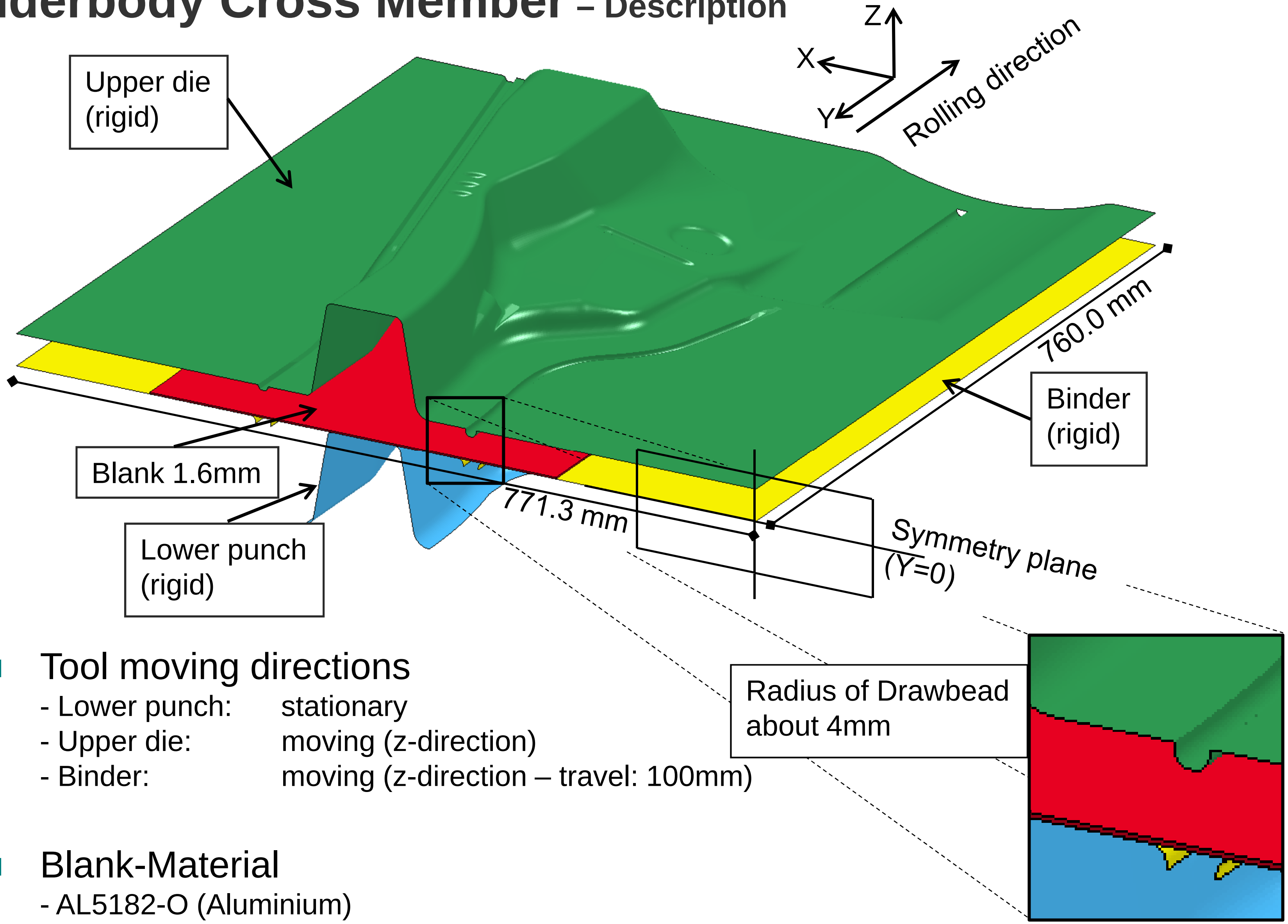
D.J. Benson

Square Tube Buckling



D.J. Benson

Underbody Cross Member – Description



- Tool moving directions
  - Lower punch: stationary
  - Upper die: moving (z-direction)
  - Binder: moving (z-direction – travel: 100mm)

- Blank-Material
  - AL5182-O (Aluminium)

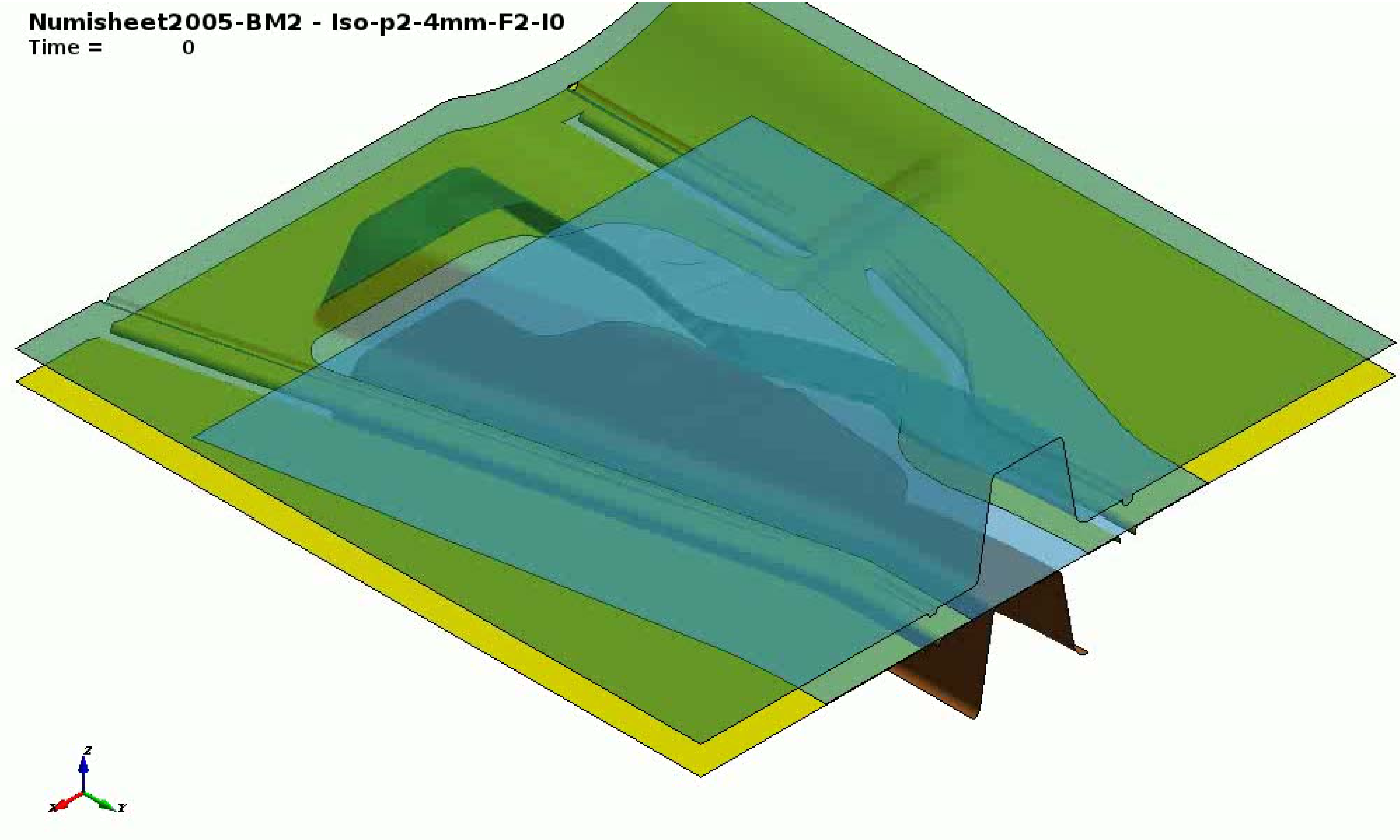
Underbody Cross Member – Simulation models

- identical for all
  - material model: *MAT_TRANSVERSELY_ANISOTROPIC_ELASTIC_PLASTIC (*MAT_037)
  - nip=5 number of integration points through the thickness
  - istupd=0 no thickness update
  - imscl=0 no “selective” mass scaling (no mass scaling at all!)
  - SMP, double precision, ncpu=4 (Dual Core AMD Opteron, 2.2 GHz)

- standard elements
  - ELFORM=16: fully integrated (4-noded) shell-elements with assumed strain formulation
  - discretizations: with adaptivity (mesh size: 4mm → 2mm → 1mm) as reference solution
    without adaptivity: mesh-sizes: 2mm; 4mm; 8mm

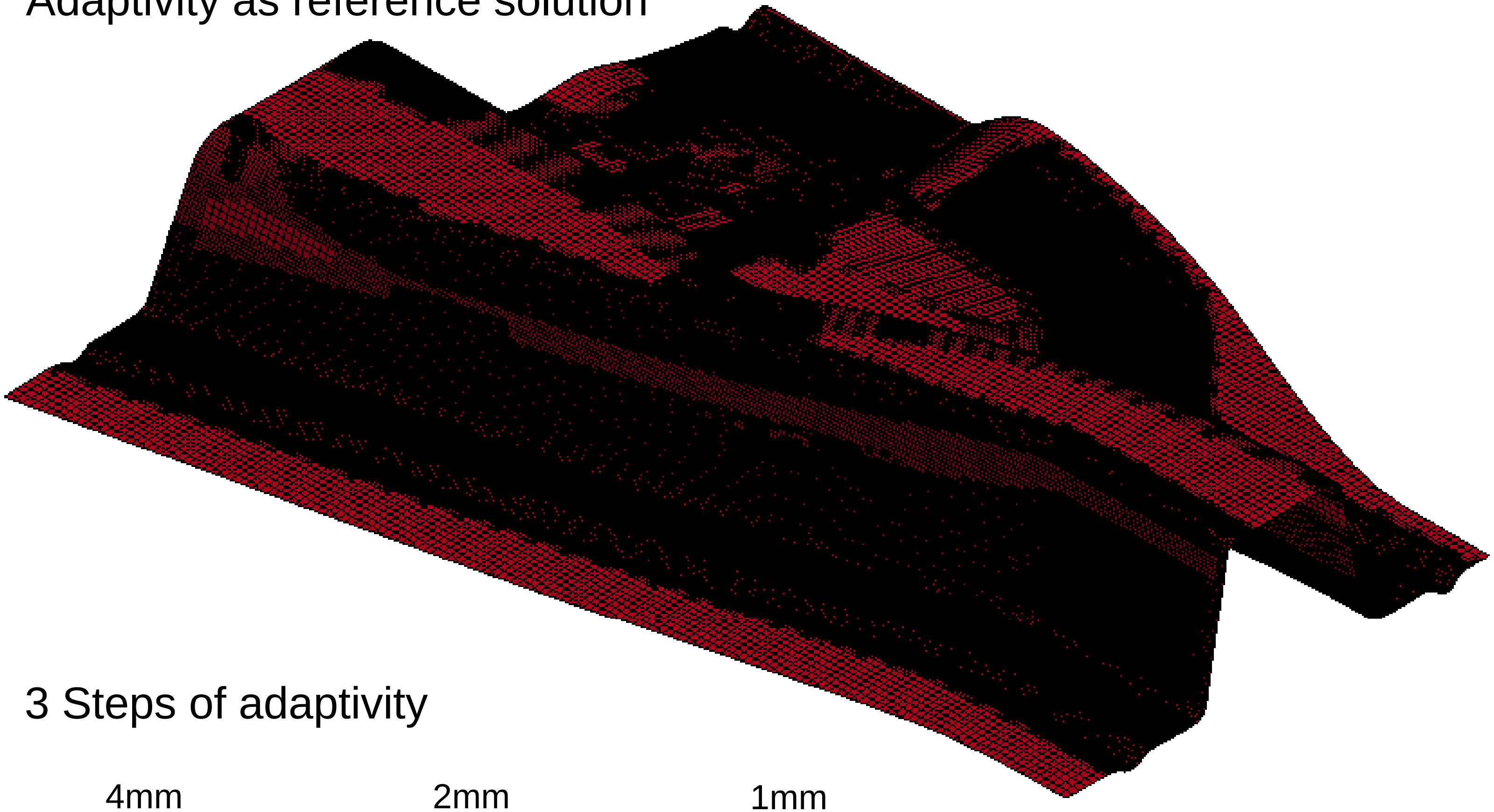
- 2D-NURBS elements
  - Formulation: FORM=2 (rotation free formulation)
  - Integraion rule: INT=0 (reduced integration)
  - Polynomial: p2 (quadratic), p3 (cubic), p4 (quartic), p5 (quintic)
  - discretizations: mesh-sizes: 4mm; 8mm; 16mm
  - number of interpolation elements/ NURBS-elements: NISR=PR; NISS=PS

Underbody Cross Member

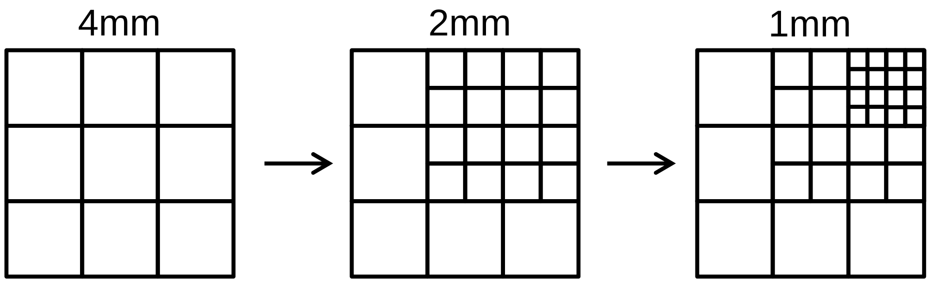


Underbody Cross Member – Standard-Elements

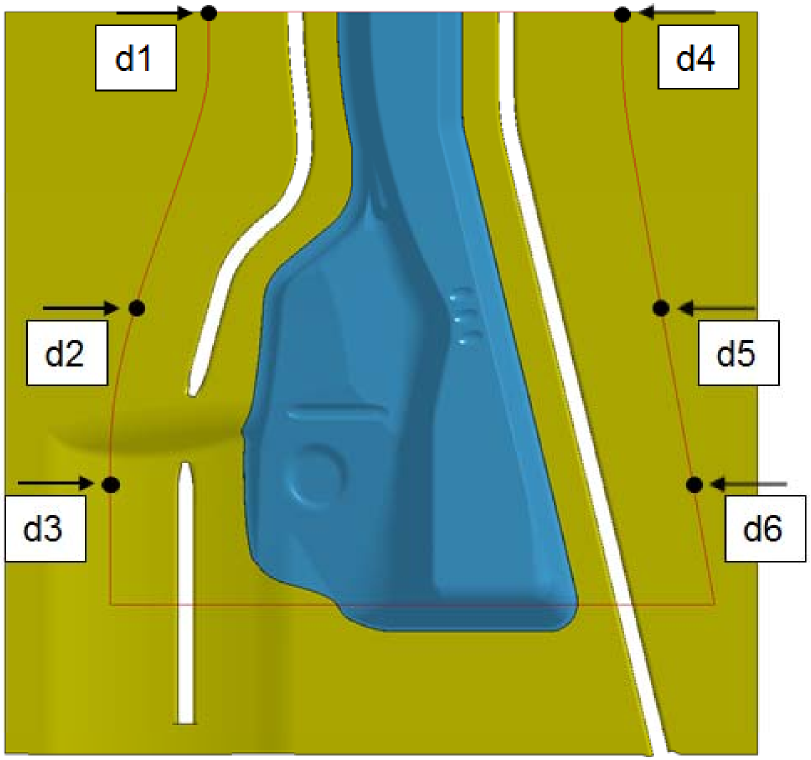
- Adaptivity as reference solution

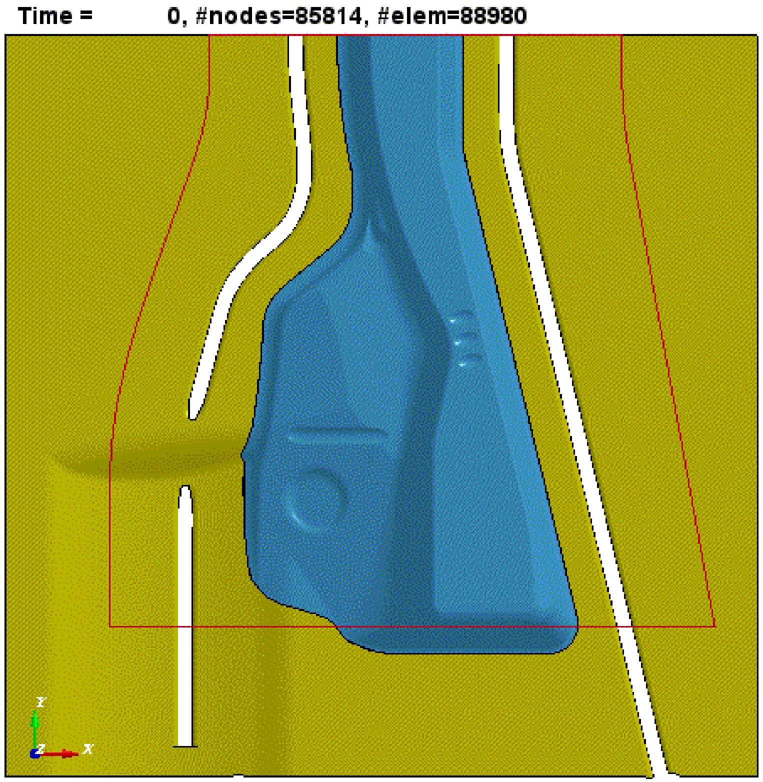


- 3 Steps of adaptivity

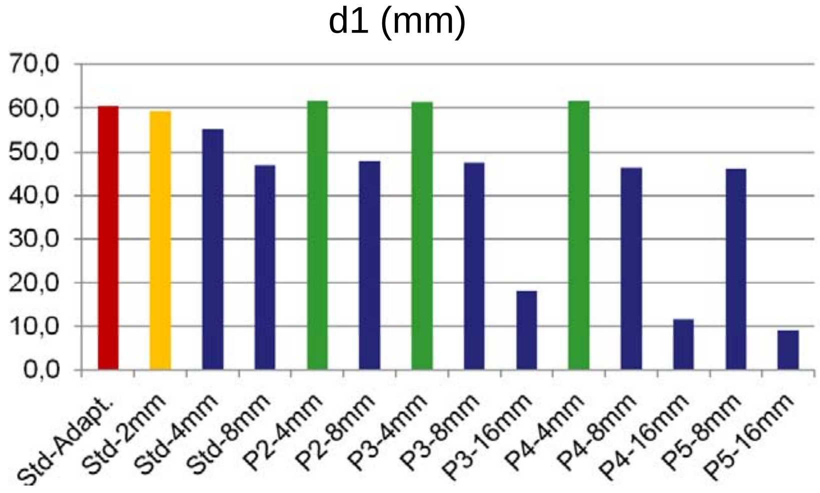


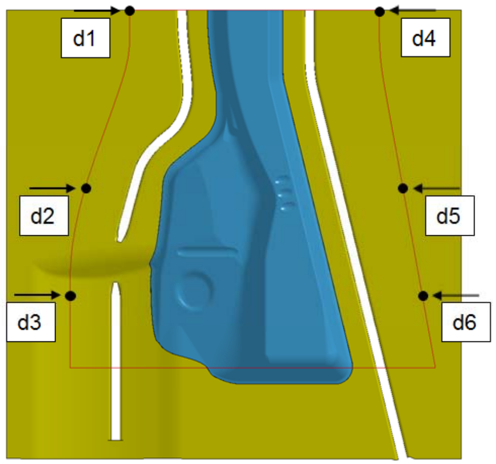
Underbody Cross Member – Draw-in

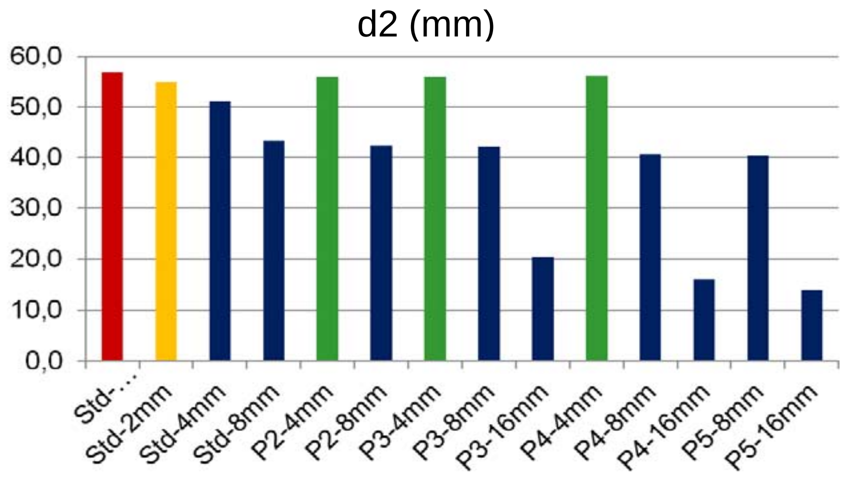


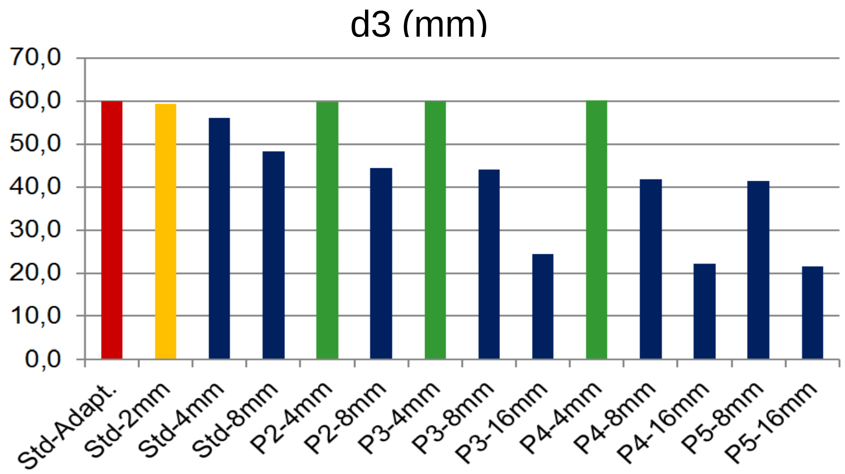


Underbody Cross Member – Draw-in





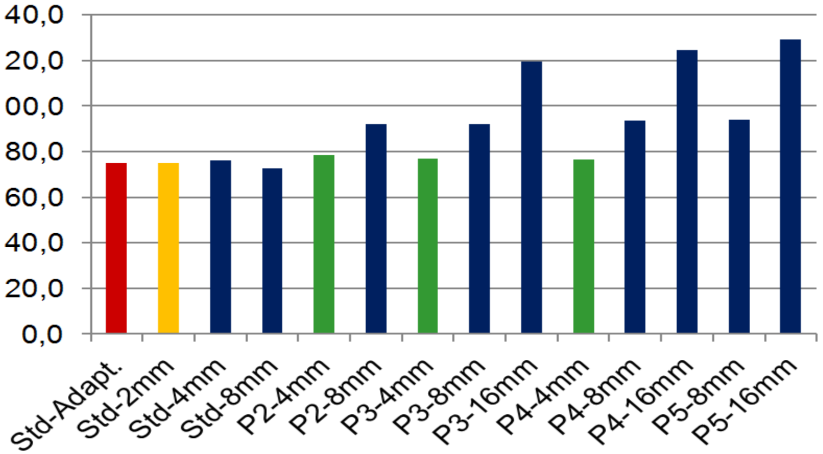


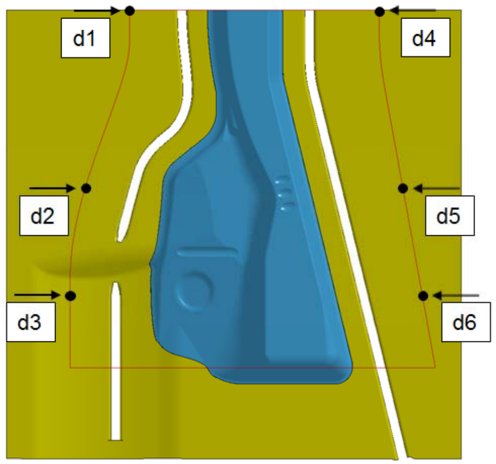


- larger mesh size → less draw-in (behavior is too stiff)

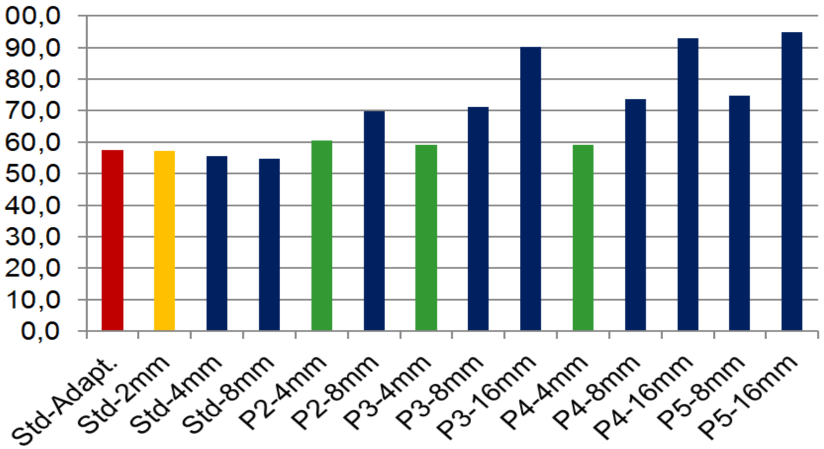
Underbody Cross Member – Draw-in

d4 (mm)

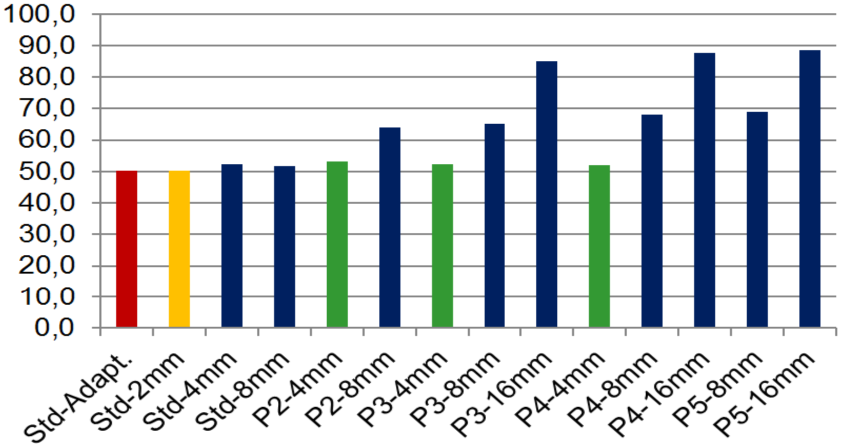




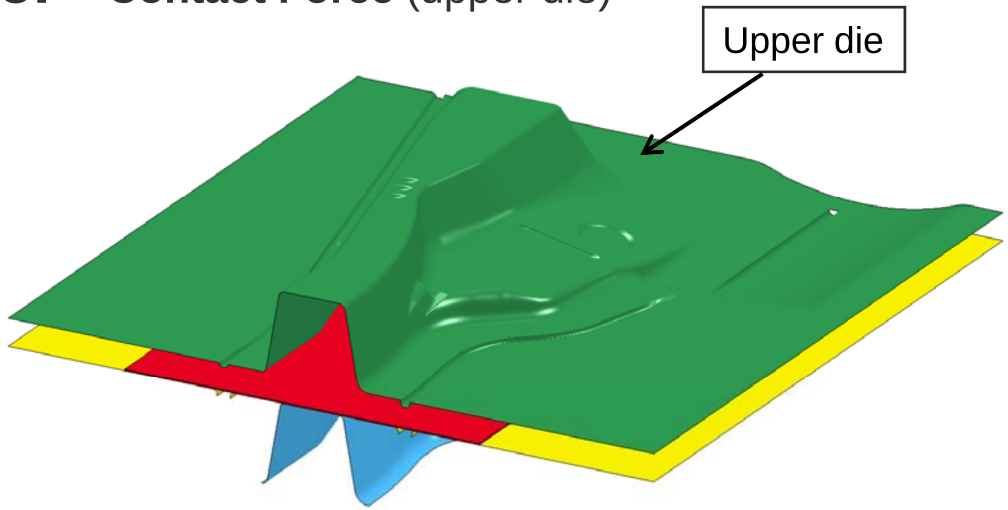
d5 (mm)

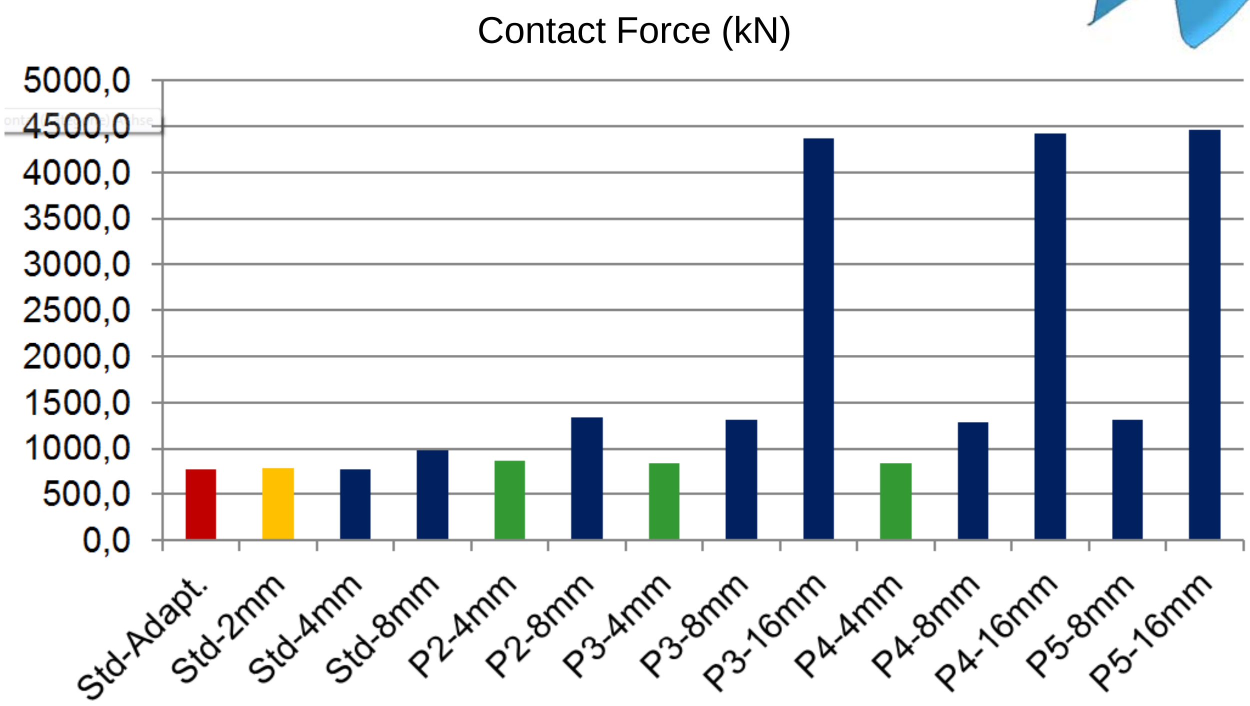


d6 (mm)

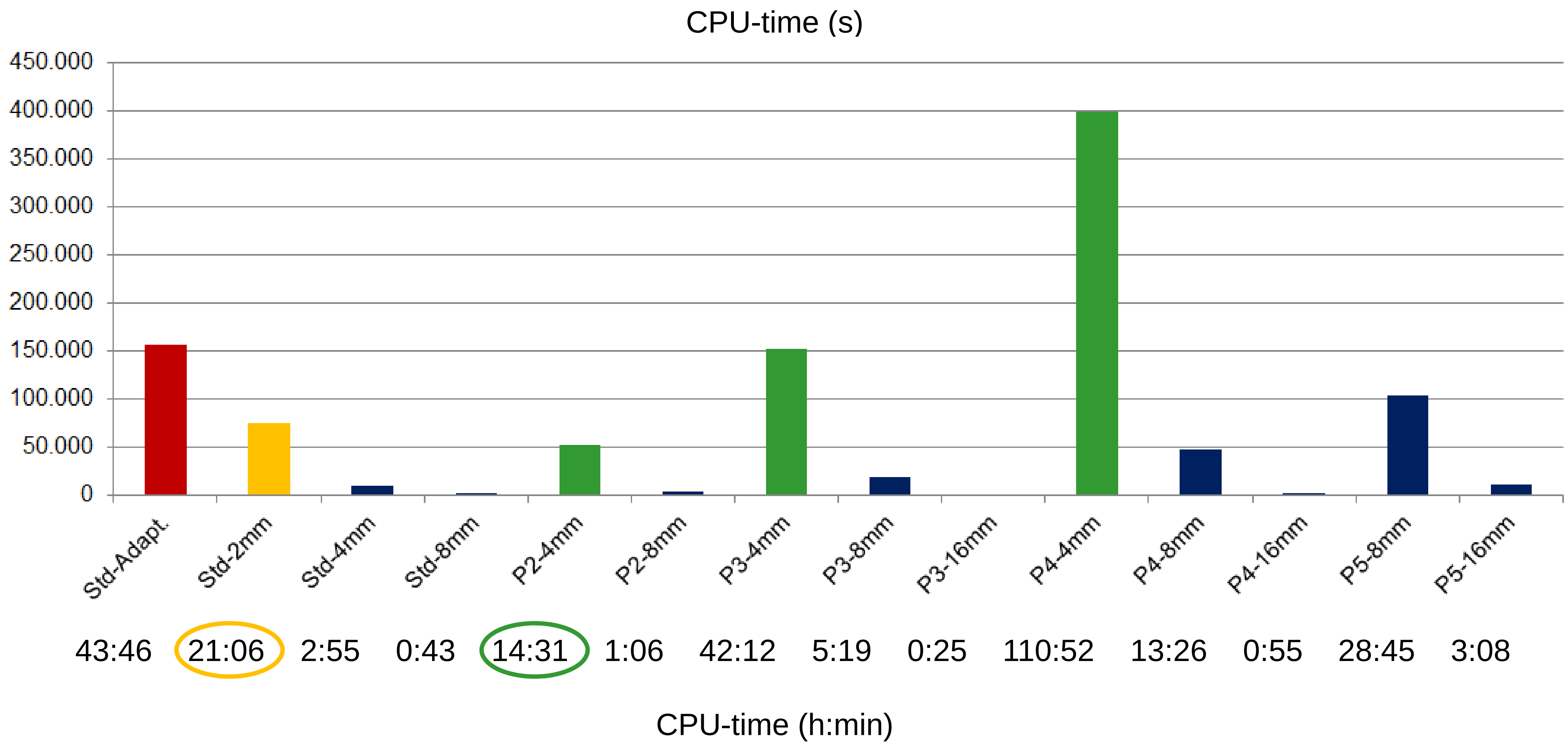


Underbody Cross Member – **Contact Force** (upper die)

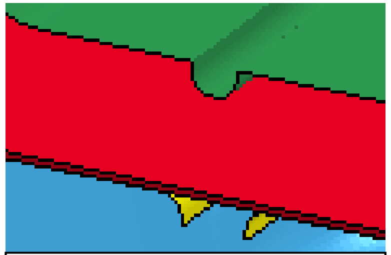




Underbody Cross Member – CPU-time



Underbody Cross Member – Summary



Radius of Drawbead about 4mm

- Detailed discretization of “Drawbead” needs a fine discretization (<=4mm), no matter what type of elements
- Rotation free elements with reduced integration show best behavior
- CPU-time for comparable discretizations (i.e: p1_2mm ←→ p2_4mm) are promising (no CODE optimization yet!)→ cost competitive
- CPU-time increase for NURBS with same discretizations for next order of polynomial (i.e.: p2_4mm→p3_4mm): Factor 2.5-2.8
- Higher order does not help anything in this example (spacing of control points define mesh size)

Summary

- NURBS-based elements run stable
- higher order accurate isogeometric analysis can be cost competitive
  - but missing a couple of "special" issues for industrial sheet metal forming applications
- code optimization necessary to make it faster
- in this example: geometry dictates the mesh size (independent of polynomial order!)
- NEW: mpp-implementation available!

Outlook

- perform a lot more studies in different fields → experience
- motivate customers (and researchers) to "play" with these elements
- further implementation
  - post-processing directly with NURBS
  - (selective) mass scaling
  - use NURBS for contact (instead of interpolation elements)
  - make pre- and post-processing more user-friendly
  - introduce 3D NURBS elements
  - … much more

Thank you!