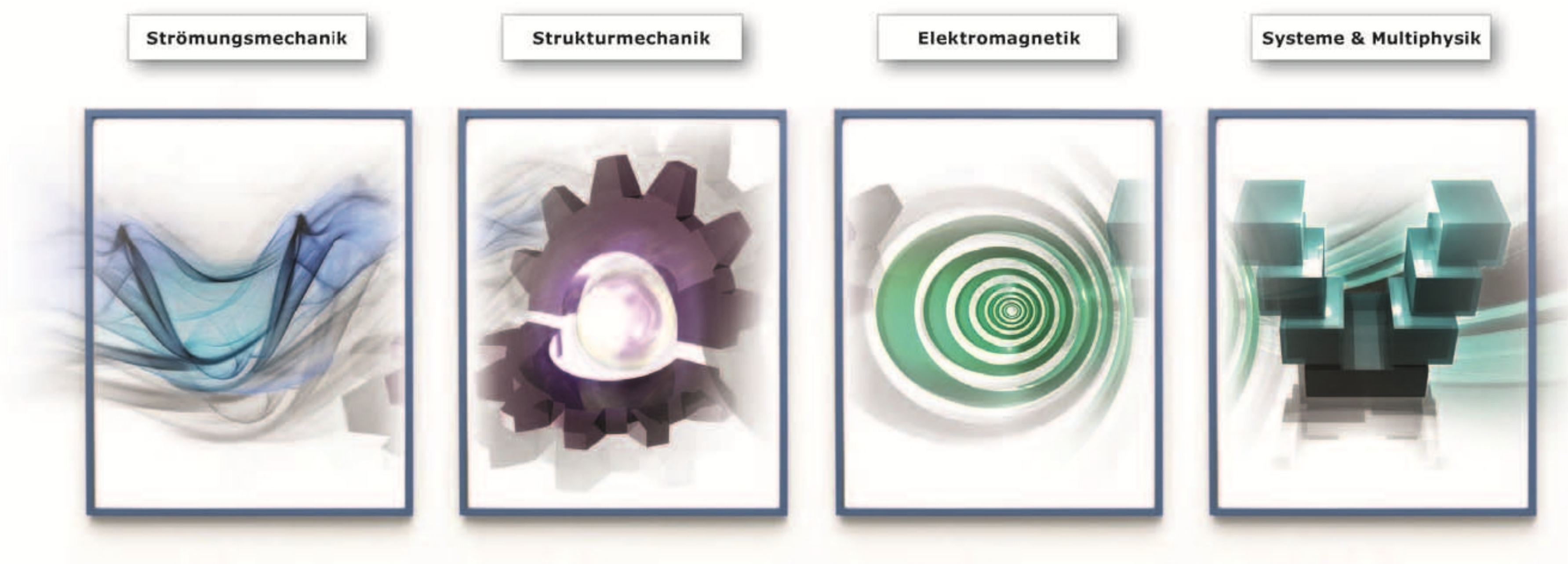
ANSYS CONFERENCE &
30. CADFEM USERS' MEETING



Robuste und effiziente Kontaktmodellierung in LS-DYNA: Wie gut sind die neuen Optionen?

Dr.-Ing. Ulrich Stelzmann
CADFEM Service

Introduction

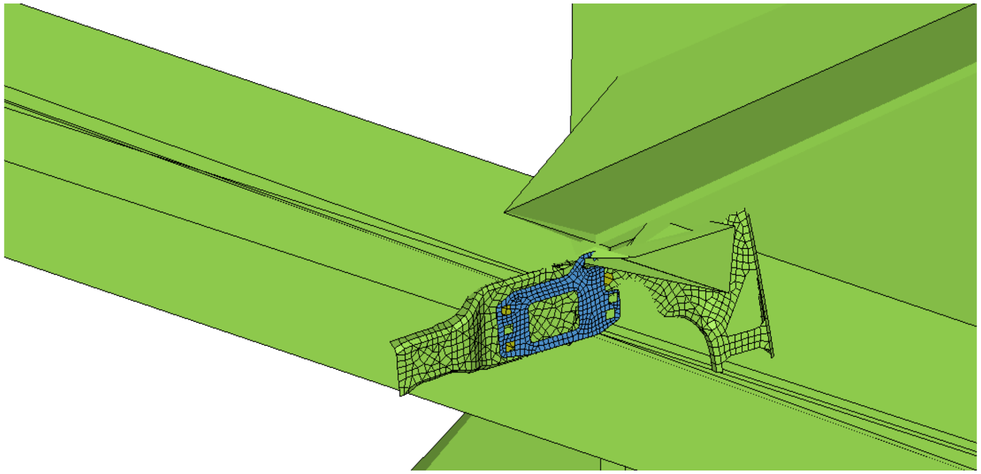
For implicit solution methods nonlinear contact is mostly a challenge and is avoided whenever it is possible.

For the explicit solution method nonlinear contact is the default.

Nevertheless, contact is very often the reason if an LS-DYNA simulation fails.

Contact is like an inexhaustible source of energy which can destroy each model completely if contact forces are acting in a wrong direction.

LS-DYNA has very powerful but also very complex contact options. For quality and efficient problem solution it is important to know the characteristics of the most important settings.



Introduction

Different types of contact available:

1) Stonewalls (*RIGIDWALL)

2) Sliding Interfaces
   3d contact (*CONTACT_...)
   2d contact (*CONTACT_2D…)

3) Geometric Contact entities (*CONTACT_ENTITY)

4) Guided cable (*CONTACT_GUIDED_CABLE)

5) Interior contact for solid elements (*CONTACT_INTERIOR)

Sliding interfaces

***CONTACT_OPTION1_{OPTION2}_{OPTION3}_{OPTION4}_{OPTION5}_{OPTION6}**

Contact types

LS-DYNA offers a huge number of different contact types, most of them also have some additional options.

a13	*CONTACT_AIRBAG_SINGLE_SURFACE	5	*CONTACT_NODES_TO_SURFACE
	*CONTACT_AUTOMATIC_BEAMS_TO_SURFACE	5	*CONTACT_NODES_TO_SURFACE_INTERFERENCE
26	*CONTACT_AUTOMATIC_GENERAL	10	*CONTACT_ONE_WAY_SURFACE_TO_SURFACE
i26	*CONTACT_AUTOMATIC_GENERAL_INTERIOR	10	*CONTACT_ONE_WAY_SURFACE_TO_SURFACE_INTERFERENCE
a 5	*CONTACT_AUTOMATIC_NODES_TO_SURFACE	20	*CONTACT_RIGID_NODES_TO_RIGID_BODY
a10	*CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE	21	*CONTACT_RIGID_BODY_ONE_WAY_TO_RIGID_BODY
t10	*CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE_TIEBREAK	19	*CONTACT_RIGID_BODY_TWO_WAY_TO_RIGID_BODY
13	***CONTACT_AUTOMATIC_SINGLE_SURFACE**	22	*CONTACT_SINGLE_EDGE
13	*AUTOMATIC_SINGLE_SURFACE_MORTAR	4	*CONTACT_SINGLE_SURFACE
13	*AUTOMATIC_SINGLE_SURFACE_SMOOTH	1	*CONTACT_SLIDING_ONLY
13	*AUTOMATIC_SINGLE_SURFACE_TIED	p 1	*CONTACT_SLIDING_ONLY_PENALTY
a 3	*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE	7	*CONTACT_SPOTWELD
a 3	*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_MORTAR	s 7	*CONTACT_SPOTWELD_WITH_TORSION
a 3	*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_MORTAR_TIED	3	*CONTACT_SURFACE_TO_SURFACE
t 3	*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_TIEBRAK	3	*CONTACT_SURFACE_TO_SURFACE_INTERFERENCE
a 3	*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_SMOOTH	8	*CONTACT_TIEBREAK_NODES_TO_SURFACE
18	*CONTACT_CONSTRAINT_NODES_TO_SURFACE	8	*CONTACT_TIEBREAK_NODES_ONLY
17	*CONTACT_CONSTRAINT_SURFACE_TO_SURFACE	9	*CONTACT_TIEBREAK_SURFACE_TO_SURFACE
23	*CONTACT_DRAWBEAD	6	*CONTACT_TIED_NODES_TO_SURFACE
23	*CONTACT_DRAWBEAD_INITIALIZE	o 6	*CONTACT_TIED_NODES_TO_SURFACE_OFFSET
16	*CONTACT_ERODING_NODES_TO_SURFACE	7	*CONTACT_TIED_SHELL_EDGE_TO_SURFACE
14	*CONTACT_ERODING_SINGLE_SURFACE	o 7	*CONTACT_TIED_SHELL_EDGE_TO_SURFACE_OFFSET
15	*CONTACT_ERODING_SURFACE_TO_SURFACE	b 7	*CONTACT_TIED_SHELL_EDGE_TO_SURFACE_BEAM_OFFSET
27	*CONTACT_FORCE_TRANSDUCER_CONSTRAINT	c 7	*CONTACT_TIED_SHELL_EDGE_TO_SURFACE_CONSTRAINED_OFFSET
25	*CONTACT_FORCE_TRANSDUCER_PENALTY	2	*CONTACT_TIED_SURFACE_TO_SURFACE
m 5	*CONTACT_FORMING_NODES_TO_SURFACE	f 2	*CONTACT_TIED_SURFACE_TO_SURFACE_FAILURE
m10	*CONTACT_FORMING_ONE_WAY_SURFACE_TO_SURFACE	o 2	*CONTACT_TIED_SURFACE_TO_SURFACE_OFFSET
m 3	*CONTACT_FORMING_SURFACE_TO_SURFACE		

Contact options

Option2: THERMAL, THERMAL_FRICTION

- important for coupled structural-thermal analysis

Option3: ID

- always recommended if more than one contact to exist

Option4: OFFSET, BEAM_OFFSET, CONSTRAINED_OFFSET

- important for TIED based contact if a gap exists

Option5: MPP

- rarely important for distributed parallel version

Option6: ORTHO_FRICTION

- new method to describe friction dependent on pressure and direction

Characterization

Purpose of Contact

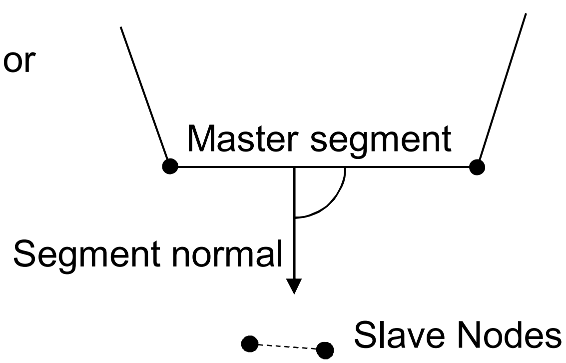
- Allows unmerged Lagrangian elements to interact with each other
  - Impacting, pushing, sliding
  - Permanently or temporarily tied together (_TIED, _TIEBREAK)

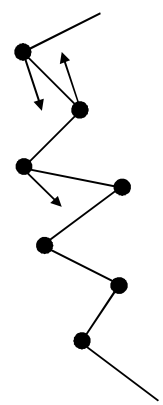
- Contact will not look for
  - Material in multi-material Eulerian elements (ALE)
  - Corpuscular particles (CPM, see *AIRBAG_PARTICLE)
  - Discrete particles (*ELEMENT_DISCRETE_SPHERE)
  - SPH elements with each other

Contact Input

- The contact algorithm prevents the penetration of nodes into element (contact) segments.
- Contact segments can be element faces of solid elements or the element area of shell elements.
- For a contact definition, parts of the model coming in contact must be described as so-called master and slave side. If it is not possible to describe two contacting model parts, a single surface contact can be used instead and only a slave side has to be defined.
- The contact partners can be defined by direct input of the nodes and segments or simply by a list of PART numbers. Internally LS-DYNA uses always nodes and segments.
- In some contact types the normal direction of the contact plane is important. By solid elements the normal is always outward directed, by shell elements the element normal is used as the first assumption.

contact type
nodes_to_surface or
surface_to_surface





contact type
single_surface

Input example: Single surface contact, soft=2

Lines printed in bold are mandatory

```
*CONTACT_AUTOMATIC_SINGLE_SURFACE_MPP_ID
$#     cid    title
         1Self contact
$#  ignore    bucket  lcbucket  ns2track  inititer    parmax    unused    cparm8
         0       200         0         3         2  1.000500         0         0
$#  unused   chksegs     pensf   grpable
&        0         0  1.000000         0
$#    ssid      msid     sstyp     mstyp    sboxid    mboxid       spr       mpr
         1         0         2         0         0         0         0         0
$#      fs        fd        dc        vc       vdc    penchk        bt        dt
     0.100     0.000     0.000    0.000     10.000         0     0.0001.0000E+20
$#     sfs       sfm       sst       mst      sfst      sfmt       fsf       vsf
  1.000000  1.000000     0.000     0.000  1.000000  1.000000  1.000000  1.000000
$# Card A
$#    soft    sofscl    lcidab    maxpar     sbopt     depth     bsort    frcfrq
         2  0.100000         0  1.025000  5.000000         5         0         1
$# Card B
$#  penmax    thkopt    shlthk     snlog      isym     i2d3d    sldthk    sldstf
     0.000         0         0         1         1         0     0.000     0.000
$# Card C
$#    igap    ignore    dprfac    dtstif    unused      unused    flangl   cid_rcf
         1         0     0.000     0.000         0         0     0.000
$# Card D
$#   q2tri    dtpchk     sfnbr    fnlscl    dnlscl      tcso    tiedid    shledg
         4     0.000     0.000         0         0         0         0         1
$# Card E
$#  sharec
         0
```

Input example: nodes to surface contact, soft=1

Lines printed in bold are mandatory

```
*CONTACT_NODES_TO_SURFACE_ID
$#     cid      title
         2
$#    ssid      msid     sstyp     mstyp    sboxid    mboxid       spr       mpr
         7         9         4         0         0         0         0         1
$#      fs        fd        dc        vc       vdc    penchk        bt        dt
     0.300     0.000     0.000     0.000      10.0               0.0001.0000E+20
$#     sfs       sfm       sst       mst      sfst      sfmt       fsf       vsf
  1.000000  1.000000     0.000     0.000
$#    soft    sofscl    lcidab    maxpar     sbopt     depth     bsort    frcfrq
         1  0.100000
$#  penmax    thkopt    shlthk     snlog      isym     i2d3d    sldthk    sldstf
     0.000         0         0         1         1         0     0.000     0.000
$#    igap    ignore    dprfac    dtstif   unused    unused    flangl   cid_rcf
         1         0
$#   q2tri    dtpchk     sfnbr    fnlscl    dnlscl      tcso    tiedid    shledg
         0
$#  sharec
         0
*SET_SEGMENT
$     ssid
         9
...
*SET_NODE
$     nsid
         7
...
```

Input example: recommended control settings

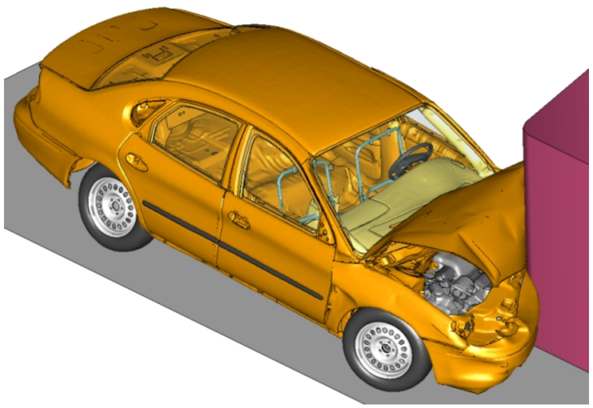
```
*CONTROL_CONTACT
$# Card 1
$ non_AUTOMATIC only: shell thickness on, automatic reorientation
$#  slsfac    rwpnal    islchk    shlthk    penopt    thkchg     orien    enmass
                                       1                             2
$# Card 2
$#  usrstr    usrfrc     nsbcs    interm     xpene     ssthk      ecdt   tiedprj

$# Card 3
$#   sfric     dfric       edc       vfc        th     th_sf    pen_sf

$# Card 4
$ Ignore initial penetrations, error if spotwelds not connected
$ remove spotwelds if segments fail, reduced contact thickness near spotwelds
$#  ignore    frceng   skiprwg    outseg   spotstp   spotdel   spothin
         2                                       1         1  0.500000
$# Card 5
$#    isym    nserod    rwgaps    rwgdth     rwksf      icov    swradf    ithoff

$# Card 6
$ square shell edge, nodal masses in soft2 stiffness without mass scaling
$ use all segments for force transducers
$#  shledg    pstiff    ithcnt    tdcnof     ftall    unused    shltrw
         1         0                             1
$
```

Contact definition

It is possible to use only one contact definition for a whole model, with one
***contact_automatic_single_surface**
(please don't use the *contact_automatic_general for that)

Advantages:

- ✓ Easy to define (simply leave slave and master side blank)
- ✓ Can't forget anything
- ✓ No redundant contact definitions
- ✓ Better performance in parallelization



Disadvantages:

- ➢ More expensive in CPU time
- ➢ Less options to define local contact parameters like
  - Thickness offset
  - Friction
- ➢ No output of contact forces
- ➢ Always symmetric contact behavior
- ➢ Shell and Beam thickness is included (use always AUTOMATIC option for single surface contact)

Disadvantages of one large Single Surface Contact:

- More expensive in CPU time
- **Less options to define local contact parameters like**
  - **Thickness offset**
  - Friction
- No output of contact forces
- Always symmetric contact behavior
- Shell and Beam thickness is included (use always AUTOMATIC option for single surface contact)

*PART_CONTACT

With *PART_CONTACT, some contact parameters can be defined locally:

- ➢ Friction (does not make sense!)
- ✓ Contact thickness as optional value (OPTT) or as scale factor (SFT)
- ✓ Contact stiffness as scale factor (SSF)

Available for:

*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE
*CONTACT_SINGLE_SURFACE
*CONTACT_AUTOMATIC_NODES_TO_SURFACE
*CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE
*CONTACT_AUTOMATIC_SINGLE_SURFACE
*CONTACT_AIRBAG_SINGLE_SURFACE
*CONTACT_ERODING_SINGLE_SURFACE
*CONTACT_AUTOMATIC_GENERAL

Disadvantages of one large Single Surface Contact:

- More expensive in CPU time
- **Less options to define local contact parameters like**
  - Thickness offset
  - **Friction**
- No output of contact forces
- Always symmetric contact behavior
- Shell and Beam thickness is included (use always AUTOMATIC option for single surface contact)

Contact Friction

Static and dynamic Coulomb friction coefficient is defined in ***CONTACT**: **FS** and **FD**

Transition to dynamic friction coefficient requires that a decay coefficient **DC** be input

$$\mu = FD + (FS - FD) \cdot e^{-DC \cdot |v|}$$

Frictional energy is optionally computed and output into the binary contact interface database (INTFOR) for visualization as “Surface Energy Density”.

- Set **FRCENG=1** in ***control_contact**
- Regions of high friction (wear) can be quickly identified.
- Frictional energy creates heat in a coupled thermal/structural analysis, friction energy is also reported in the ASCII file sleout

Coulomb friction coefficient is function of relative velocity and optionally, of interface pressure and direction. Since ls971 R6, the ***CONTACT_..._ORTHO_FRICTION** is available for most general friction definition.

Contact Friction

Before version 971 the user was only allowed to define a different friction coefficient for each part via *PART_CONTACT. But in reality, a friction coefficient is not a material property, it depends on the 2 materials in contact.

*DEFINE_FRICTION

- ✓ Define friction coefficients between parts for use in the contact options:
  - AUTOMATIC_SINGLE_SURFACE, incl. AUTOMATIC_GENERAL
  - AUTOMATIC_NODES_TO_SURFACE
  - AUTOMATIC_SURFACE_TO_SUFACE
  - AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE
  - ERODING_SINGLE_SURFACE
- ✓ One *DEFINE_FRICTION input permitted
- ✓ Friction values are given for each pair of parts, if n parts exist in the model, then up to n(n+1)/2 unique pairs are possible
- ✓ Default friction constants are used if a pair of contacting parts have no defined friction values
- ✓ Every contact segment has an associated part ID

Contact Friction

Since ls971 R6, the ***CONTACT_..._ORTHO_FRICTION** is available for most general friction definition.

Coulomb friction coefficient can be a function of relative velocity and optionally, of interface pressure and direction:

A two dimensional table *DEFINE_TABLE_2D giving the friction coefficient in the local direction as a function of the relative velocity. Each curve in the TABLE definition defines the coefficient of friction versus the interface pressure.

Currently this feature is limited to contact types

AUTOMATIC_SURFACE_TO_SUFACE

AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE

and the contact must be defined by segment sets.

So: this can not be used in one large Single Surface Contact!

Disadvantages of one large Single Surface Contact:

- More expensive in CPU time
- Less options to define local contact parameters like
  - Thickness offset
  - Friction
- **No output of contact forces**
- Always symmetric contact behavior
- Shell and Beam thickness is included (use always AUTOMATIC option for single surface contact)

Force Transducers

Every contact definition of type nodes_to_surface and surface_to_surface writes out resulting contact forces on slave and master side to the ASCII file **rcforc**. But a single surface definition can't write a resultant contact force.

To always get force output on arbitrary contact surfaces, force transducers can be used. In contrast to many single contact definitions, force transducer don't waste CPU time and don't bother the parallelization.

*CONTACT_FORCE_TRANSDUCER_*option*_ID

- Provides a convenient means of contact force retrieval at select locations
- Slave side defines where contact forces are retrieved (by parts or segments)
- The Master side can be blank (see next page)
- Cards 2 and 3 are blank

Force Transducers

*CONTACT_FORCE_TRANSDUCER_*option*_ID

- No contact forces are created by force transducers
  - Transducers only measure forces from non-transducer contact types
  - Measured contact forces retrieved via *database_rcforc
  - Contact segments also written to the interface force file INTFOR
- Two *options* for *CONTACT_FORCE_TRANSDUCER
  - _PENALTY (default, measures forces from penalty-based contacts)
  - _CONSTRAINT (measures forces from constraint-based contacts)

Force transducer may sum up if more than one contact applies forces to one surface.

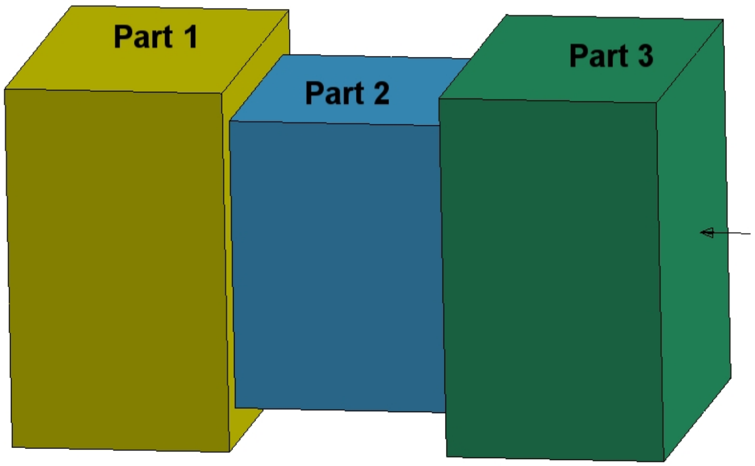
Force Transducers

Example:

If slave side contains Part 2 and master side is empty, then the force from the force transducer may be zero.

The input of a master side can be used to limit the source of contact forces, for example to Part 3.

Attention: the option FTALL=1 on *CONTROL_CONTACT should be set for this!



If the slave side is described with segment sets (*set_segment), then we are much more flexible to select local areas.

Of course, a segment set may not help if the example above was modeled with shells.

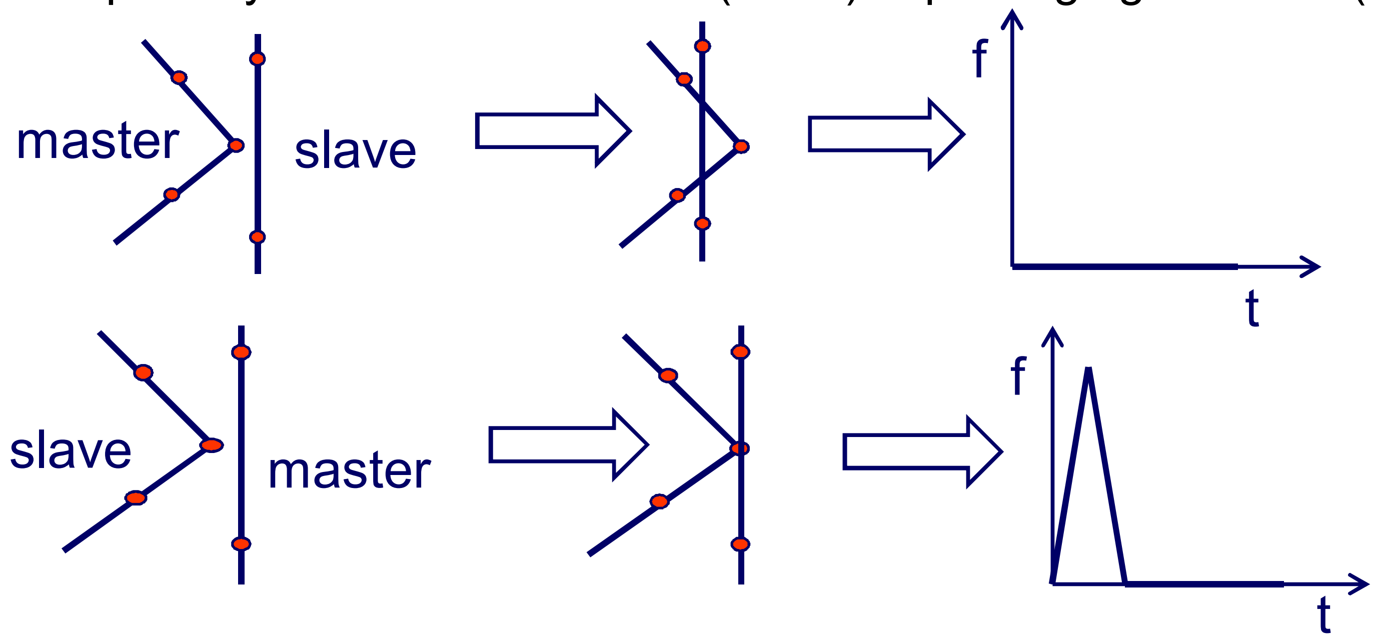
Disadvantages of one large Single Surface Contact:

- More expensive in CPU time
- Less options to define local contact parameters like
  - Thickness offset
  - Friction
- No output of contact forces
- **Always symmetric contact behavior**
- Shell and Beam thickness is included (use always AUTOMATIC option for single surface contact)

Symmetric vs. Non-symmetric

Non-symmetric contact: _one_way_surface_to_surface, _nodes_to_surface

- Generally, coarser side should be master
- Computationally efficient
- Especially well-suited to nodes (slave) impacting rigid bodies (master)



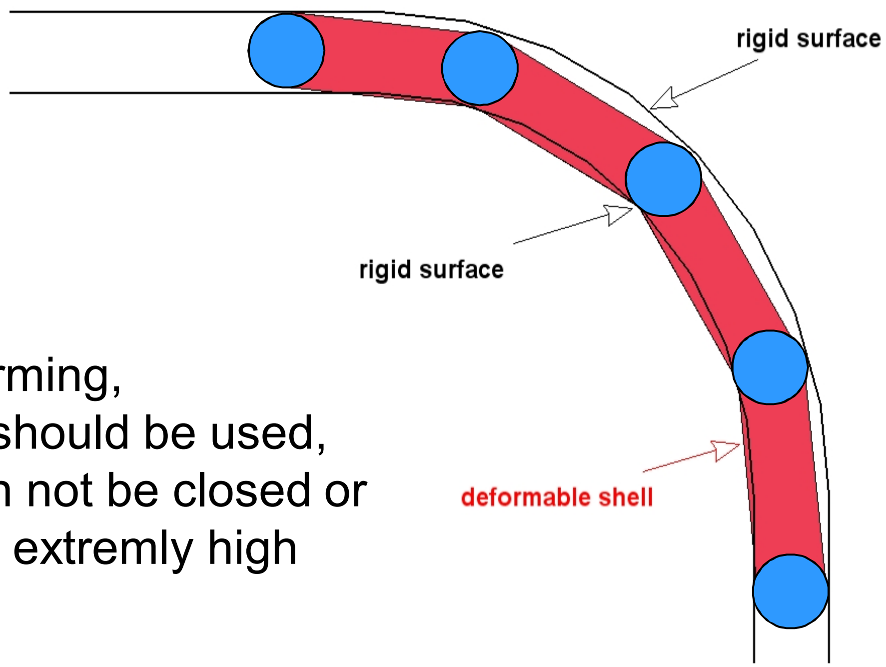
Symmetric contact: _surface_to_surface

- Two times a one-way-contact with interchanged master and slave

Symmetric vs. Non-symmetric

- Symmetric contact (_surface_to_surface) will not always give the best result. Coarse mesh may force penetrations in curved regions.
- Single surface contact will have the same problem.
- One way contact has no penetration.

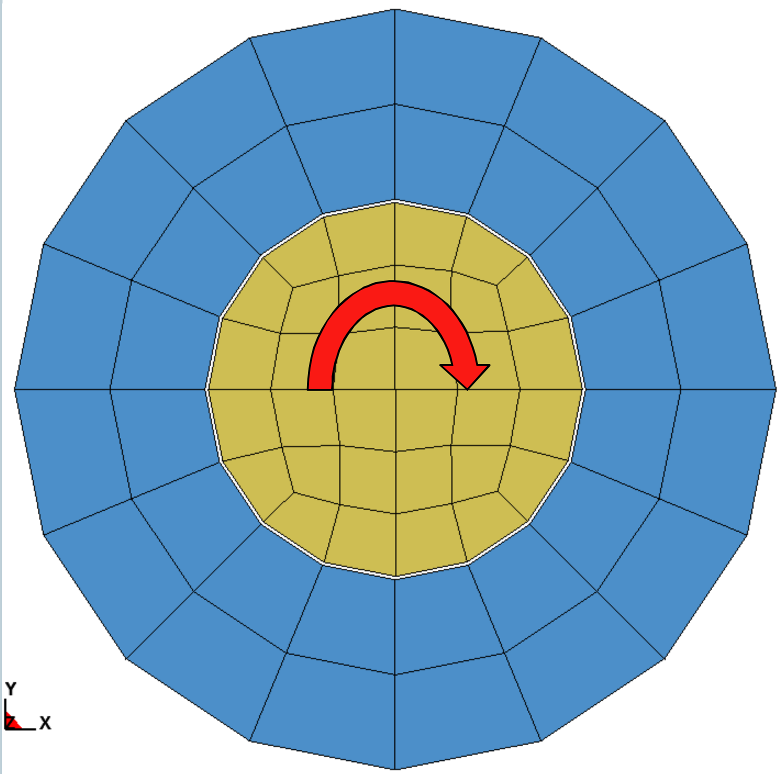
Please note: Element edges are always straight in contact, except if _SMOOTH is used!

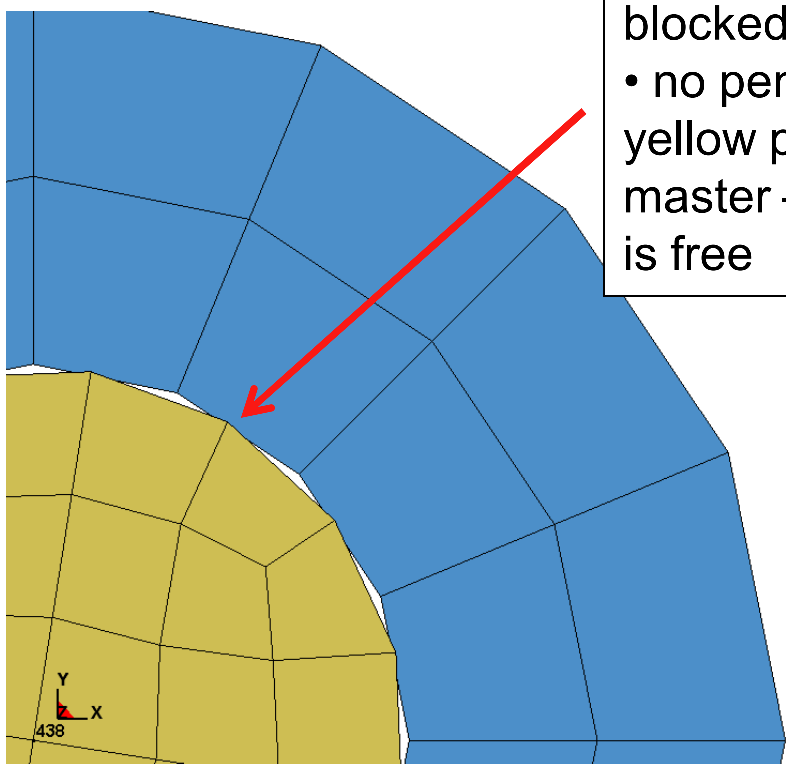


For sheet metal forming, one_way_contact should be used, otherwise tools can not be closed or punch force will be extremly high

Symmetric vs. Non-symmetric

- Symmetric contact (_surface_to_surface) will not always give the best result. Coarse mesh may force penetrations in curved regions.





- penetration if yellow part is slave – rotation is blocked
- no penetration if yellow part is master – rotation is free

Disadvantages of one large Single Surface Contact:

- More expensive in CPU time
- Less options to define local contact parameters like
  - Thickness offset
  - Friction
- No output of contact forces
- Always symmetric contact behavior
- **Shell and Beam thickness is included** (use always AUTOMATIC option for single surface contact)

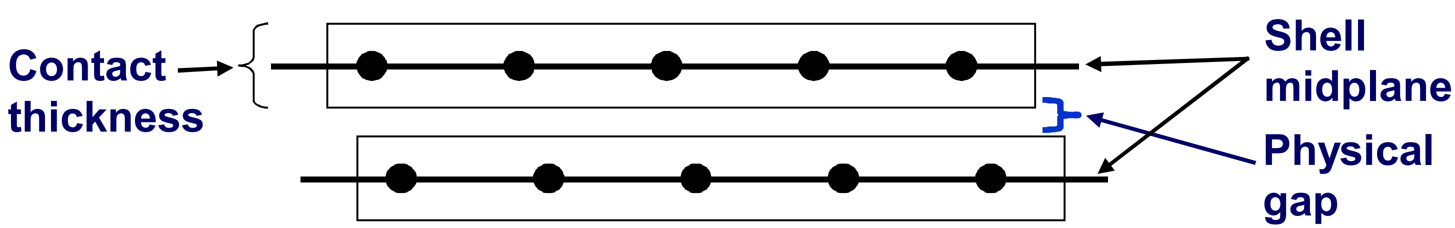
Thickness Offset

Contact Thickness

- ✓ Determines thickness offsets in contact, doesn't affect stiffness or mass
- ✓ Default contact thickness = shell thickness, beam diameter, zero for solids
- ✓ All contact types of AUTOMATIC_ and ERODING_ will always include thickness offset, the same is true if SOFT=2 is set (segment based contact). Contact thickness should **not** be very small for these contact types.
- ✓ The only way to neglect thickness offset is:
  - use non-AUTOMATIC contact type
  - set *CONTROL_CONTACT, SHLTHK=0 or 1 (global setting)
  - or set *CONTACT, Card B: THKOPT=1, SHLTHK=0 or 1

  (SHLTHK=1 is the nice option to neglect thickness for rigid bodies only, useful for metal forming)

Never use the non-AUTOMATIC Single Surface contact: *contact_single_surface !



Summary *CONTACT

Non symmetric Contact:

_one_way_surface_to_surface (slave and master must consist of faces)

_nodes_to_surface (allows only nodes on slave side, like SPH elements)

Symmetric Contact:

_surface_to_surface (slave and master must consist of faces)

Single Surface Contact:

_automatic_single_surface

_eroding_single_surface

_automatic_general

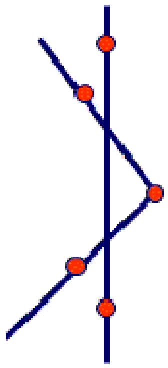
_airbag_single_surface (most expensive contact search)

Search for Penetrations

Search for penetrations

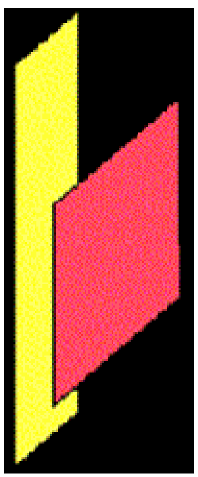
How contact penetration is found:

- Nodes to Segment contact search



  - For each node in a slave node list it is checked if it penetrates a segment from a master segment list (for Single Surface contact, both lists are identical)
  - Common approach for most contact types and for all TIED and TIEBREAK types
  - The segment is flat, only the SMOOTH-Option supports curved segments

- Segment to Segment contact search

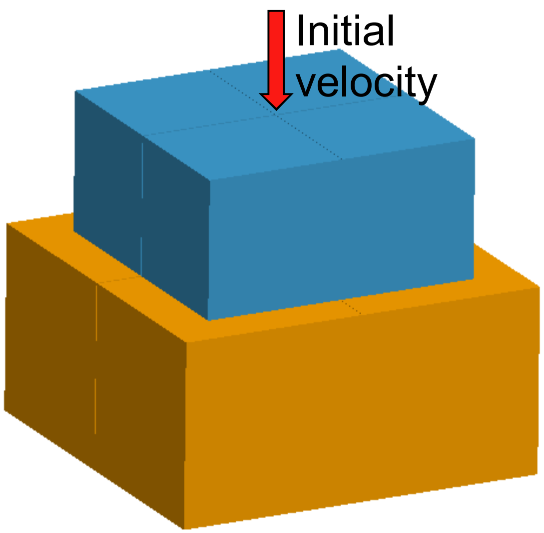


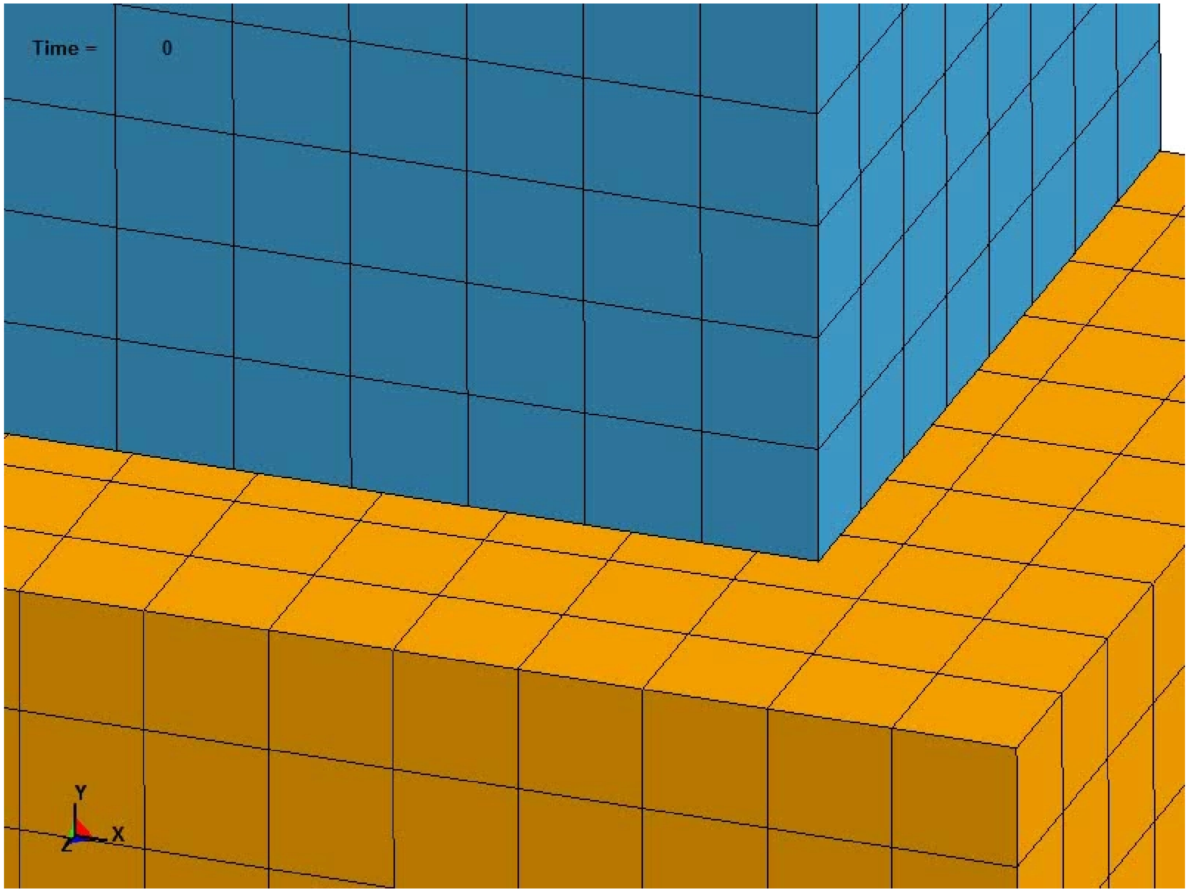
  - Penetration between two segments
  - The segments are always flat, a quad is may be split in two triangles
  - Only SOFT=2 and MORTAR activates segment to segment search

Note: If a contact is defined with Segment sets on both master and slave side, this does **not** mean that a Segment to Segment search will be done!

Search for penetrations

Sharp corners are difficult to handle in a nodal based contact:

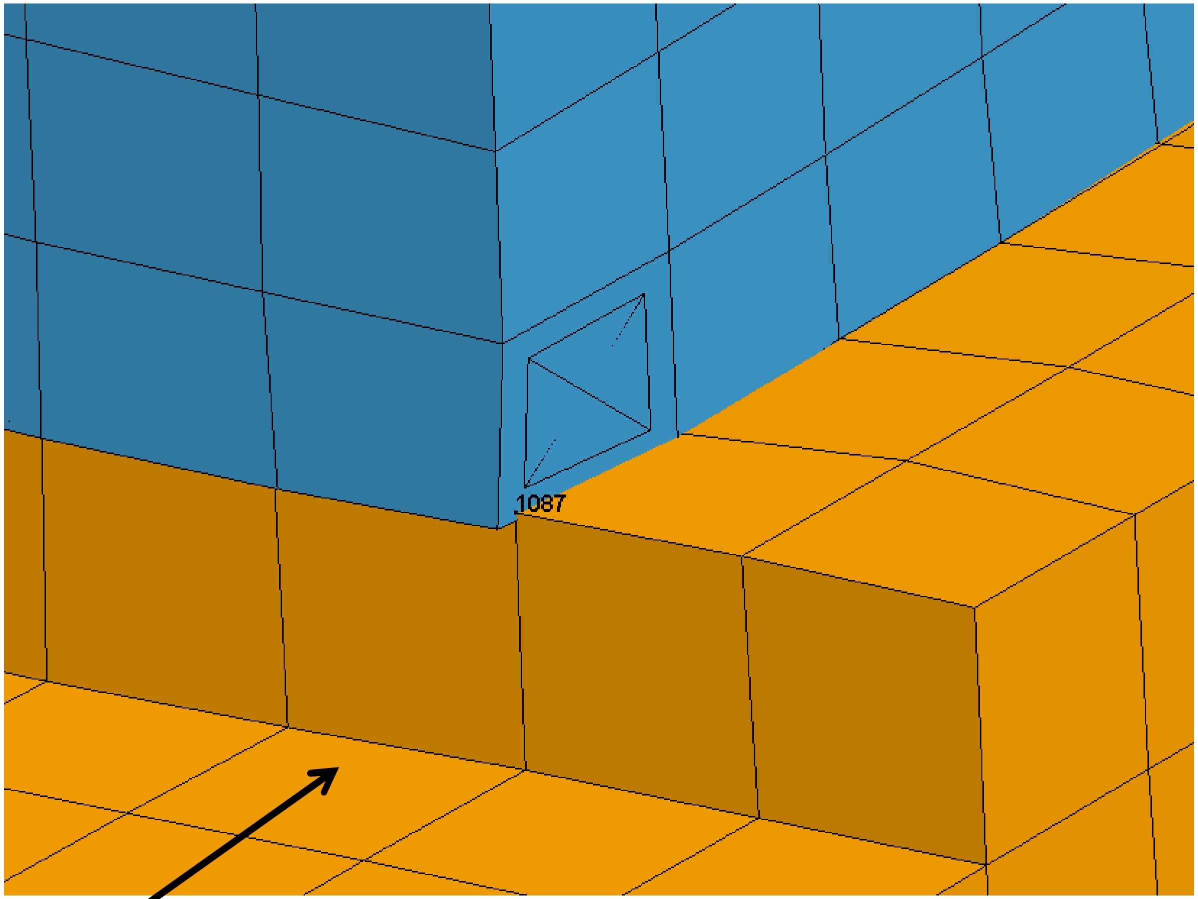




Search for penetrations

In a nodal based contact search, the node 1087 will penetrate the signed segment if the blue part slides on the surface of the yellow part.

Because of penalty contact method, there is always a small penetration.

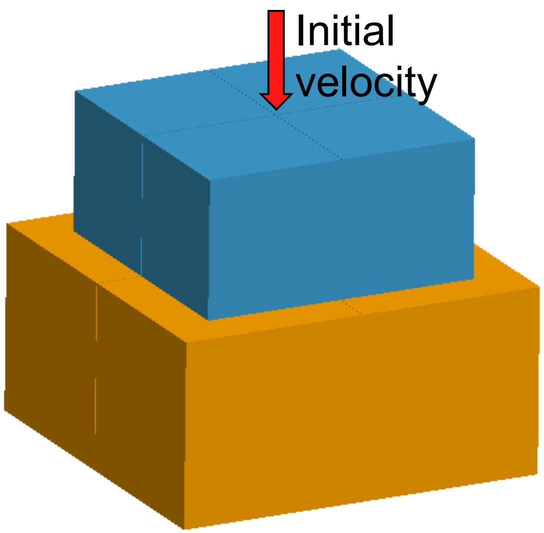


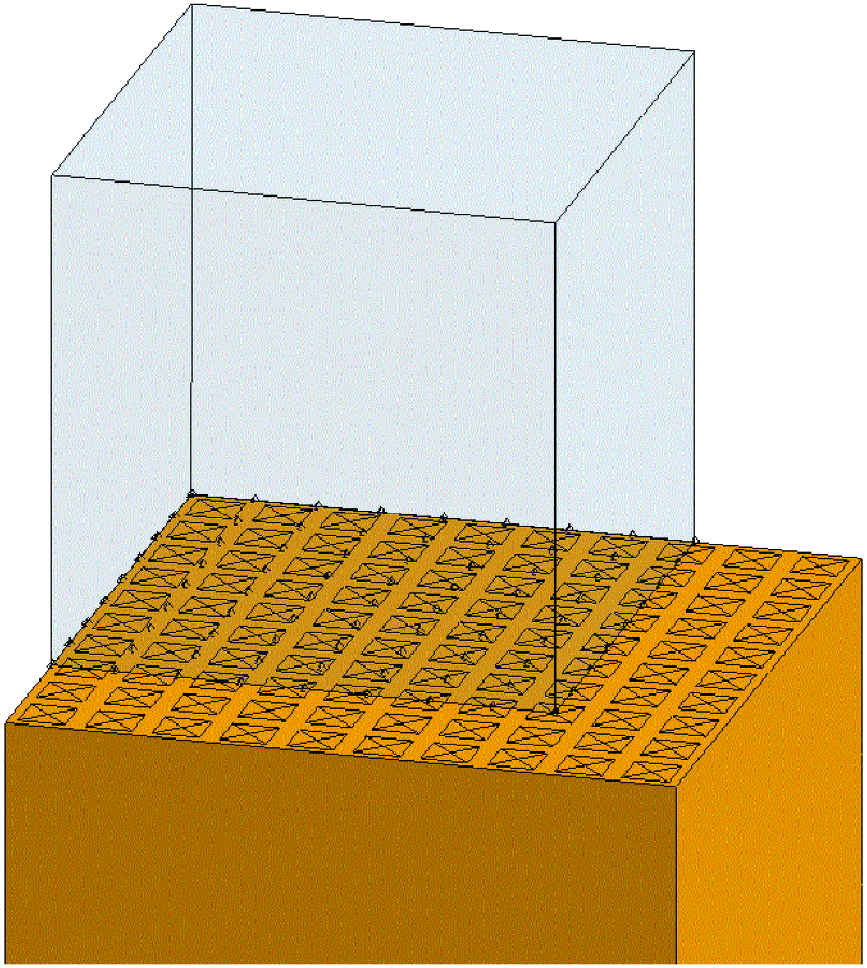
elements blanked
from the display

Search for penetrations

Sharp corners are difficult to handle in a nodal based contact:

One possible solution:
Use a surface_to_surface contact and define slave and master side by segments, only on the active contact surface (manual input).

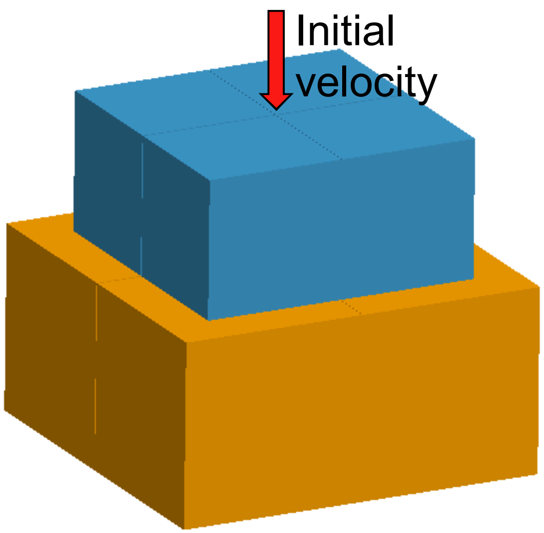


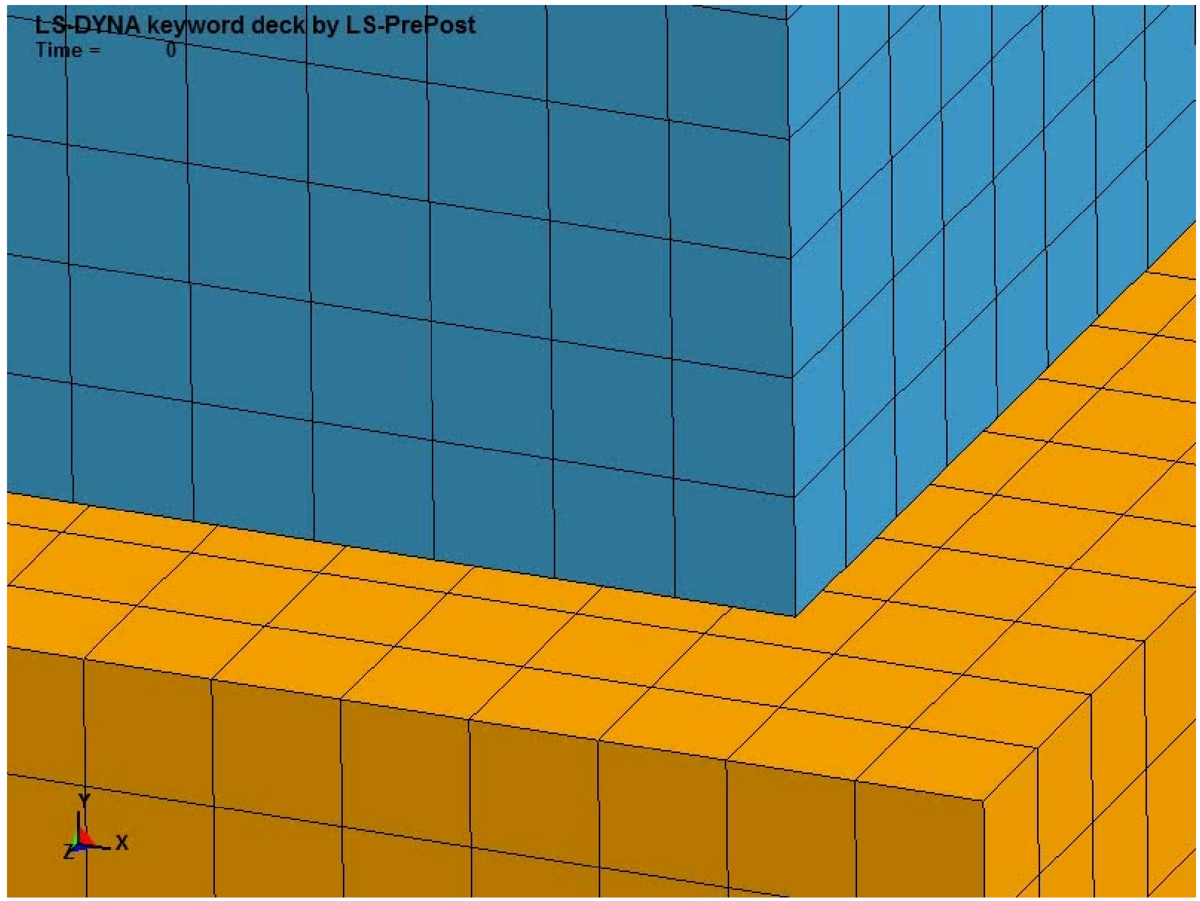


Search for penetrations

Best solution: Segment based contact search (like SOFT=2 or MORTAR)

But edge to edge checking must be turned on





The „old“

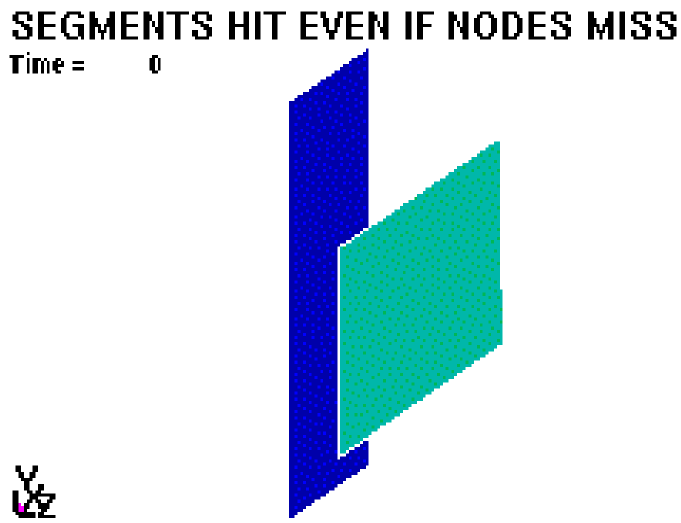
segment based contact

SOFT=2

Segment Based: SOFT=2

Segment-Based Contact (SOFT=2):

- Is an alternative, penalty-based contact algorithm for shells, solids, and thick shells. Beams are ignored.
- Searches for penetration in a unique way:
  - SOFT=2 contact checks for segment penetration, not node penetration.
  - Segment penetration is detected even if nodes don't penetrate.
  - Segment-based contact is a good option if geometry has sharp corners or edges



Segment Based: SOFT=2

Segment-Based Contact is implemented for ***contact_**...

- **surface_to_surface**, including **_automatic** and **_eroding**
- **single_surface**, including **_automatic** and **_eroding**
- **one_way_surface_to_surface**, including **_automatic**
- **airbag_single_surface**

Segment Based: SOFT=2

Additional Options for SOFT=2 Contact

- SBOPT on Opt. Card A
  - 2 (default): assumes planar segments
  - 3: takes into consideration segment warpage
  - 4: sliding option
  - 5: use options 3 and 4
- DEPTH on Opt. Card A
  - 2 (default): checks for surface penetrations
  - 3: same as 2 but depth of penetration is also checked at segment edges
  - 13: same as 3 but optimized to conserve energy
  - 23: same as 3 but optimized in terms of robustness
  - 5: same as 2 but adds check for edge-to-edge penetration
  - 15: since R5: same as 5 in R4.2.1

With default settings this contact type is nearly as expensive as the other one. With the most expensive settings SBOPT=3 and DEPTH=5 it takes approx. two times more (contact calculation only)

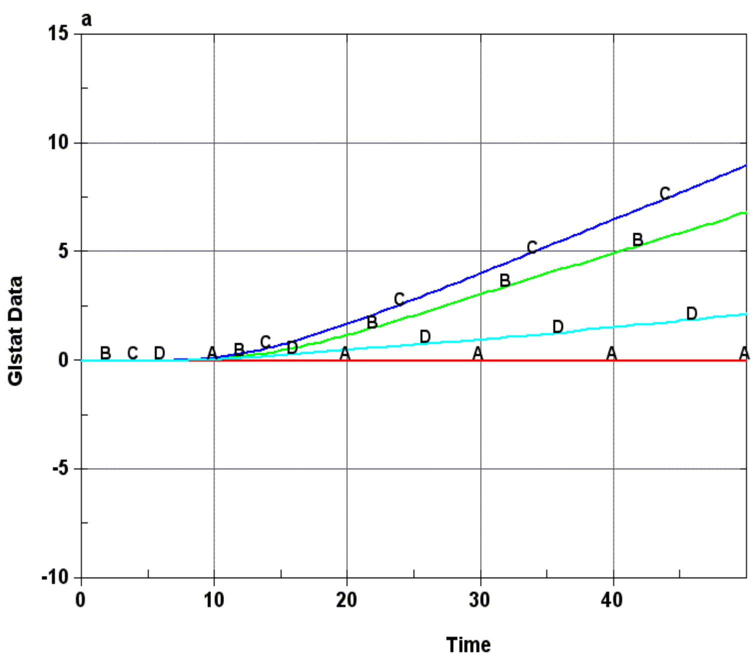
***** Recommended setting: SBOPT=3, DEPTH=5 (since R5: DEPTH=15)

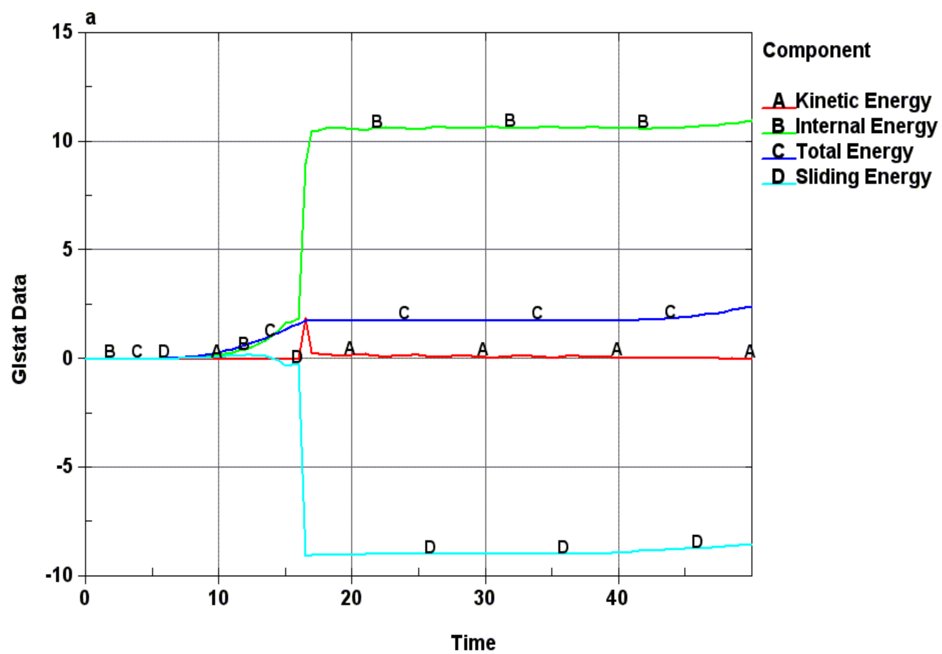
***** If sliding is important: SBOPT=5, DEPTH=5 (since R5: DEPTH=15)

Segment Based: SOFT=2

Starting with R5.1.1 the contact with SOFT=2 becomes more often unstable:

Example:





R4.2.1: DEPTH=5
R5.1.1 / R6.1.0: DEPTH=**15**

R5.1.1 / R6.1.0: DEPTH=5

Segment Based: Initial Penetration

When there are initial penetrations…

- For SOFT=2, initial penetration for each segment pair is stored and subtracted from the current penetration before calculating penalty forces.
  - Similar to setting IGNORE=1 for SOFT=0 or 1 contact.
  - Geometry is not perturbed.
- Initial penetration is both:
  - Penetration in the geometry at time zero
  - The first penetration value if a contact closes at time >0
- Formerly, SOFT=2 gives no penetration warning messages. This will be added in R4 version by setting IGNORE=2
- Because of each first penetration is ignored, there is a danger that visible penetrations become deeper and deeper during simulation. The parameter DPRFAC (*contact, optional card C) can be used to prevent this.

Segment Based: Initial Penetration

For SOFT=2, initial penetration for each segment pair is stored and subtracted from the current penetration before calculating penalty forces. Initial penetration is both:

- Penetration in the geometry at time zero
- The first penetration value if a contact closes at time >0

Because of each first penetration is ignored, there is a danger that visible penetrations become deeper and deeper during simulation. The parameter DPRFAC (*contact, optional card C) can be used to prevent this.

Contact force is calculated by the formula:

$$F = k \cdot (\delta - \delta_i)$$

k – contact stiffness
δ - current penetration
δ_i - initial penetration

If DPRFAC >0 (*contact, optional card C) then δ_i will be reduced by a value of δ_i * DPRFAC in each time step. Setting DPRFAC=0.01 will reduce δ_i by 1% each cycle. Values between 0.001 and 0.01 are recommended.

DPRFAC does not change δ_i if it was measured at time=0 (initial penetration in the geometry).

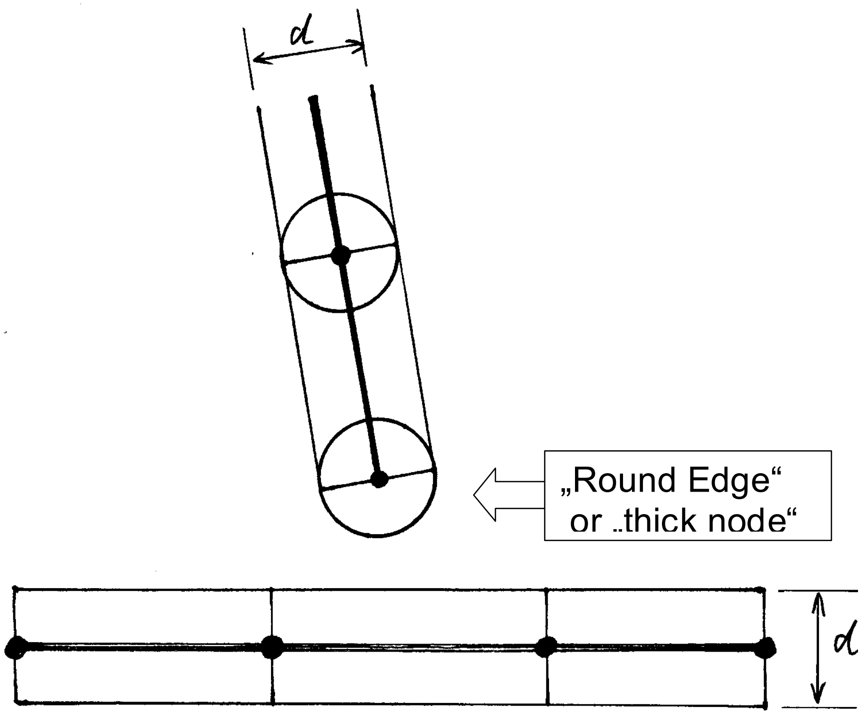
DPRFAC has a tendency to produce negative sliding energy. Use it with care!

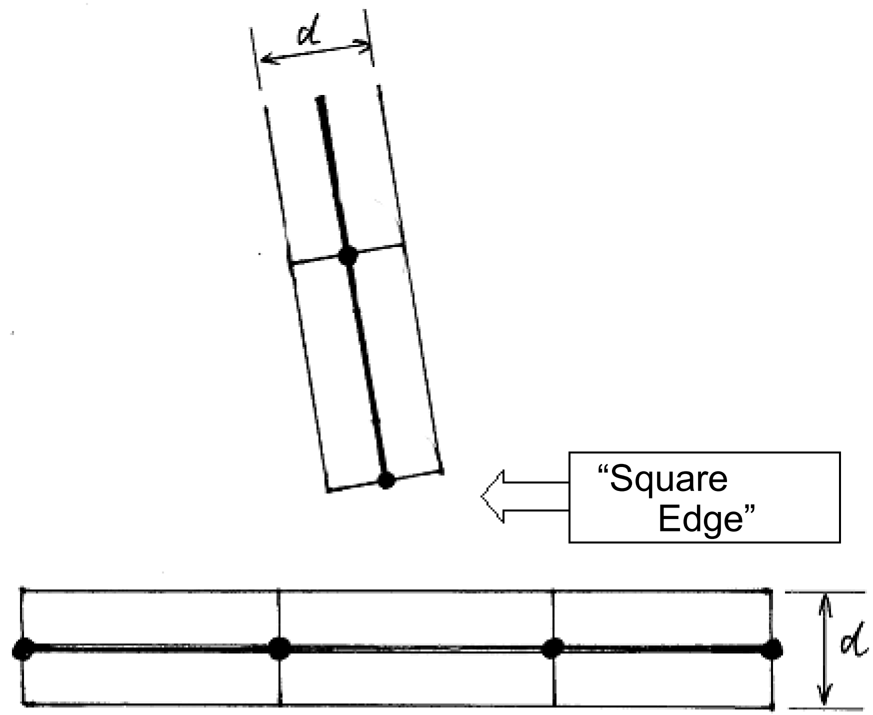
Segment Based: Square Shell Edges

For the segment based contact type (SOFT=2) a square shell edge option is supported since version ls971 R3:

*CONTROL_CONTACT, shledg=1

or *CONTACT, card D, field 8: shledg=1 (since R5.1)





The „new“

segment based contact

MORTAR

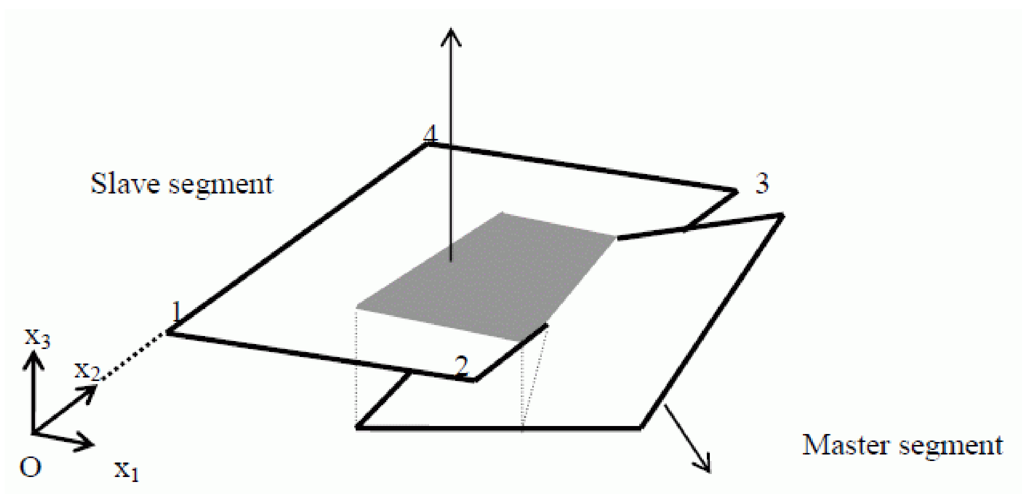
MORTAR contact

Mortar is a **segment** to **segment penalty** based contact.

For overlapping and penetrating segments a consistent nodal force assembly is performed, taking into account the individual shape functions of the segments.

In this respect the results with this contact may be more accurate, especially when considering contact with elements of higher order.

This contact is intended for implicit analysis in particular but is nevertheless supported for explicit analysis as well.



CADFEM

MORTAR contact

Available contact types:

***CONTACT_AUTOMATIC_SINGLE_SURFACE_MORTAR**
***CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_MORTAR**
***CONTACT_FORMING_SURFACE_TO_SURFACE_MORTAR**

***CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_MORTAR_TIED**
***CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_TIEBREAK_MORTAR**

- ✓ Provides contact tractions that are consistent with finite element theory, for trias/quads/tet4/pentas/hexa/tet10
- ✓ Intended for implicit analysis but works for explicit
- ✓ MPP and SMP
- ❖ Currently may be not stable enough as a large general single surface contact

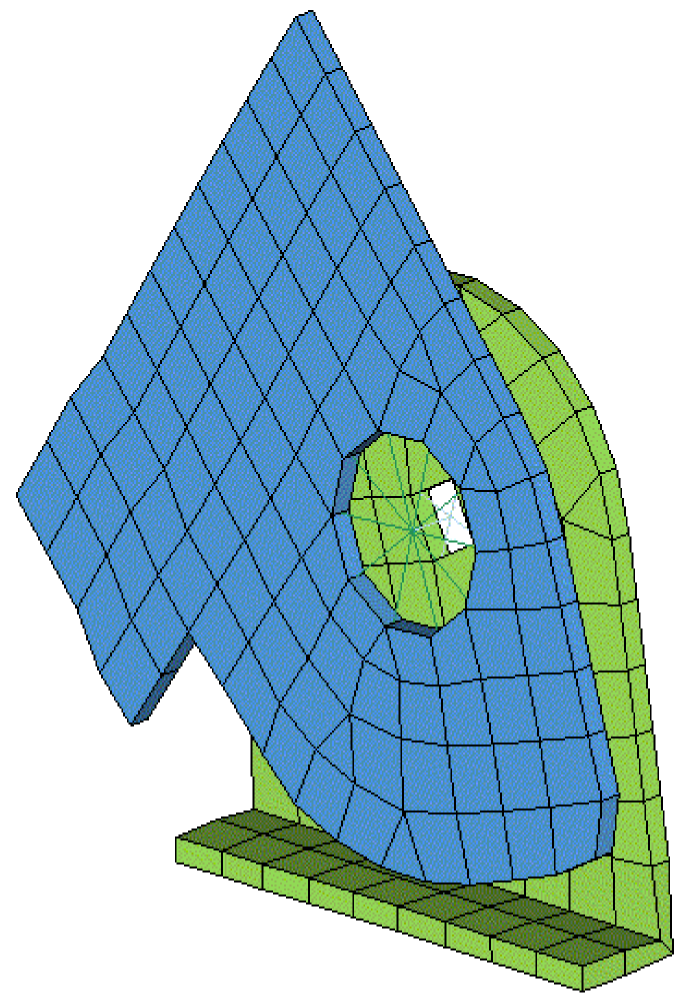
MORTAR contact

Mortar contact features

- ✓ Smooth force transition by sliding
- ✓ Edge to edge contact is supported with no extra option for shells ans solids.
- ✓ Well situated for sharp corners on solid elements.
- ✓ ”Square edge” assumed on shell edges (*control_contact, shledg=1 is always turned on!)

Disadvantages:

- ➢ May need double precision also for the explicit solver (improvement is on the way)
- ➢ Much more CPU time: a factor of **5-10** (in rare cases factor of 50!) in contact search compared to SOFT=2



CADFEM

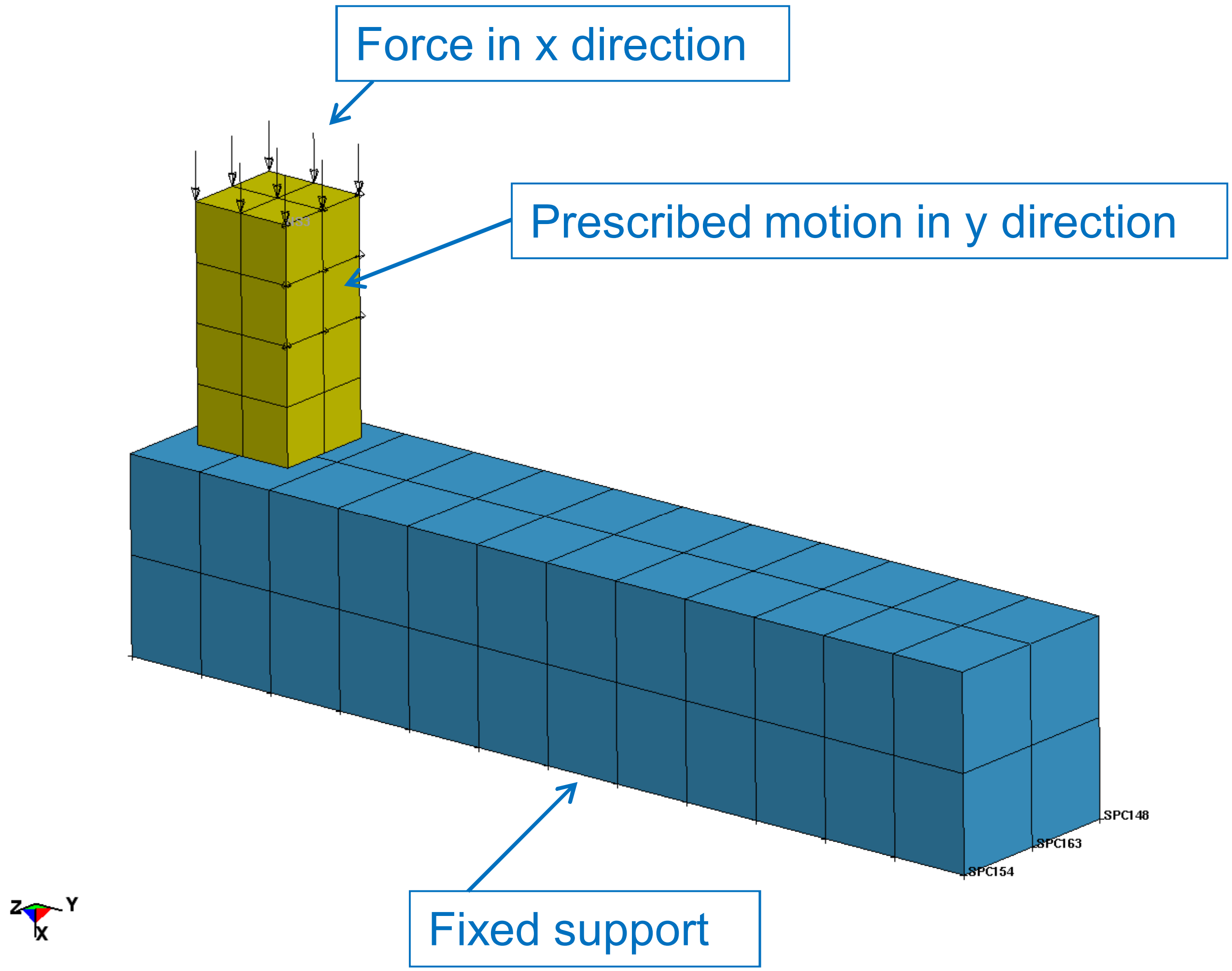
MORTAR contact

<u>Example:</u> Sliding

Compare MORTAR

and

Segment based (SOFT=2)

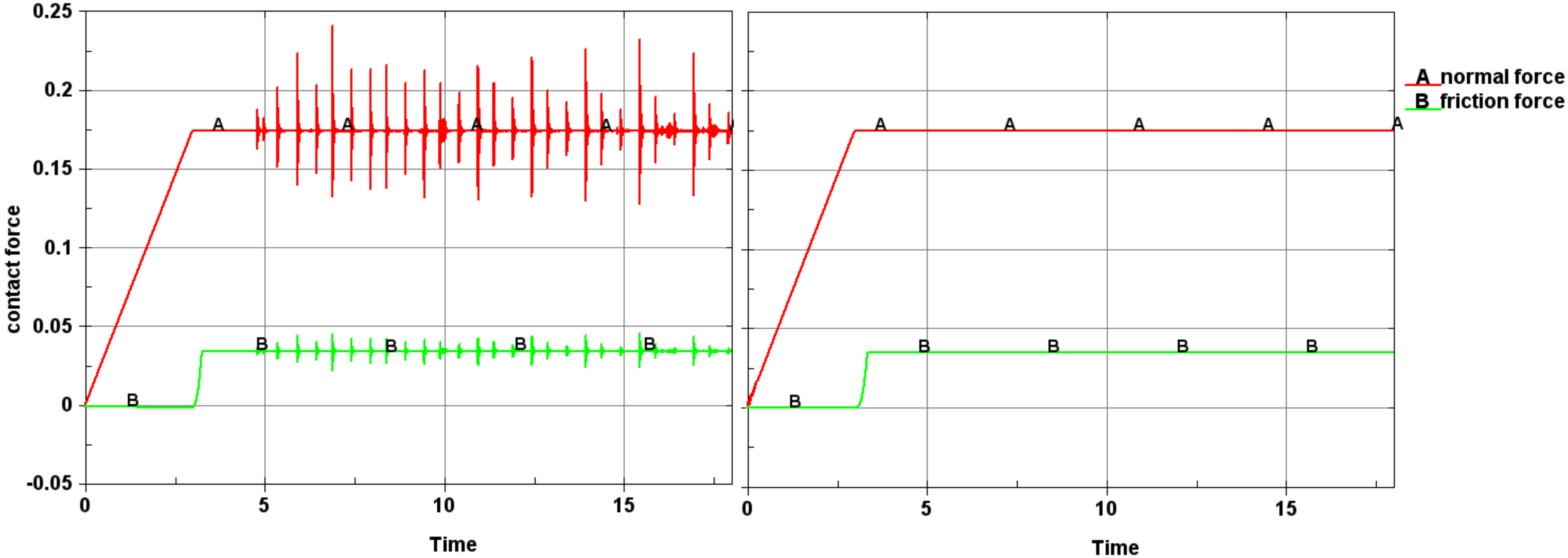


CADFEM

MORTAR contact

Example: Sliding

Compare Segment based (SOFT=2) and MORTAR



CADFEM

Characterization

How contact forces are applied:

- Penalty-based
  - By far the most common approach
  - Uses a finite contact stiffness and thus some (small) penetration occurs between surfaces in contact

- Constraint-based
  - No penetrations because of “infinite contact stiffness”
  - Tend to be less stable than penalty-based contacts
  - Some problems with rigid bodies and double contact definition
  - May be conflict with other Constrained types and Boundary conditions
  - Tied contact usually constraint-based but sometimes penalty-based

contact stiffness

Penalty Contact

Elastic, compression-only springs in normal direction to resist penetration

Very stable and tends NOT to excite mesh hourglassing (good!)

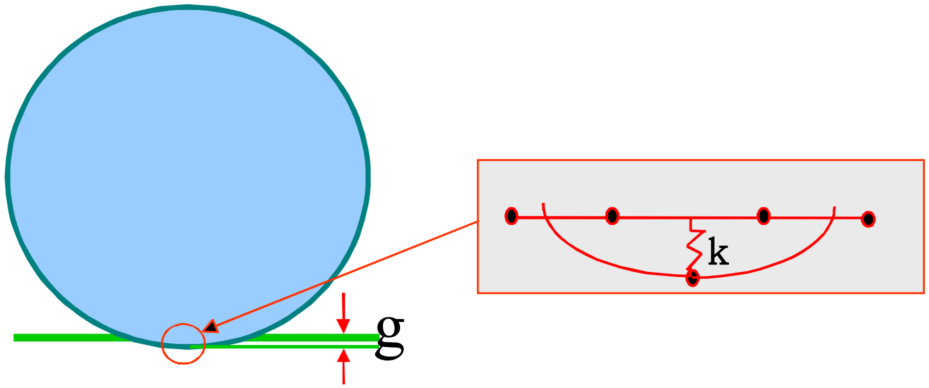
Applicable to deformable bodies and to rigid bodies

penalty force:

$$F = k \cdot g$$

with k - contact stiffness

g - penetration depth



the biggest disadvantage of the penalty method is, that a contact stiffness has to be defined, which might be not optimal for all cases:

→ if the stiffness is too low, the penetration will be too high

→ if the stiffness is too high, high frequency vibrations are activated and the explicit time integration procedure may become unstable

Penalty Contact

Standard method (SOFT=0): Contact stiffness is calculated based on element length and material property

For shell elements the contact stiffness is determined by:

$$k = slsfac \cdot sf \cdot K \cdot A / d$$

with
- *slsfac* - global scale factor given in *CONTROL_CONTACT
- *sf* - local scale factor given in *CONTACT_, Card 3 (sfs,sfm)
- *K* - bulk modulus
- *A* - element area
- *d* - thickness or shortest diagonal

and for solid elements:

$$k = slsfac \cdot sf \cdot K \cdot A^2 / V$$

with
- *slsfac* - global scale factor given in *CONTROL_CONTACT
- *sf* - local scale factor given in *CONTACT_, Card 3 (sfs,sfm)
- *K* - bulk modulus
- *A* - segment area
- *V* - volume of element

Penalty Contact – Soft Constraint

"Soft Constraint" is activated by soft=1 in *CONTACT, optional card A. It is only a different formula to calculate contact stiffness for the penalty method.

SOFT=1 is usually recommended for contact involving soft materials, e.g. foams, and steel or for contact between parts of dissimilar mesh densities.

If soft contsraint formulation is used, contact stiffness is maximum of …

- the SOFT=0 stiffness (see previous page), and
- a stiffness calculated based on stability of a spring-mass system

$$k_1 = slsfac \cdot sf \cdot K \cdot A / d \quad \text{(for shells)}$$

$$k_1 = slsfac \cdot sf \cdot K \cdot A^2 / V \quad \text{(for solids)}$$

$$k_2 = \frac{1}{2} \cdot sofscl \cdot \frac{m_n}{\Delta t^2}$$

m_n is nodal mass without mass scaling and mass elements !

Δt is time step size

$$\boxed{k = \max(k_1, k_2)}$$

The scale factor *sofscl* (*CONTACT, optional card A) is default sofscl=0.1

The maximum allowed value is sofscl=1.0 , which yields to a very stiff contact for high surface pressure.

Penalty Contact – Segment-Based SOFT=2

"Soft Constraint" is activated by soft=2 in *CONTACT, optional card A. It is a completely new method to search for contact by checking the penetration of segments to segments instead of nodes to segments.

It computes contact stiffness in a manner similar to SOFT=1:

$$k = slsfac \cdot sfs \cdot \frac{m_1 \cdot m_2}{(m_1 + m_2) \cdot \Delta t_c^2}$$

m_i is segment mass

Δt_c is dtstif, if dtstif > 0 (*CONTACT, optional card C) – new method
or |maxpar|, if maxpar < 0 (*CONTACT, optional card A) – old method
or 1.05*max (Δt_0 , Δt) with Δt_0 - first time step size at t=0
Δt - current time step size

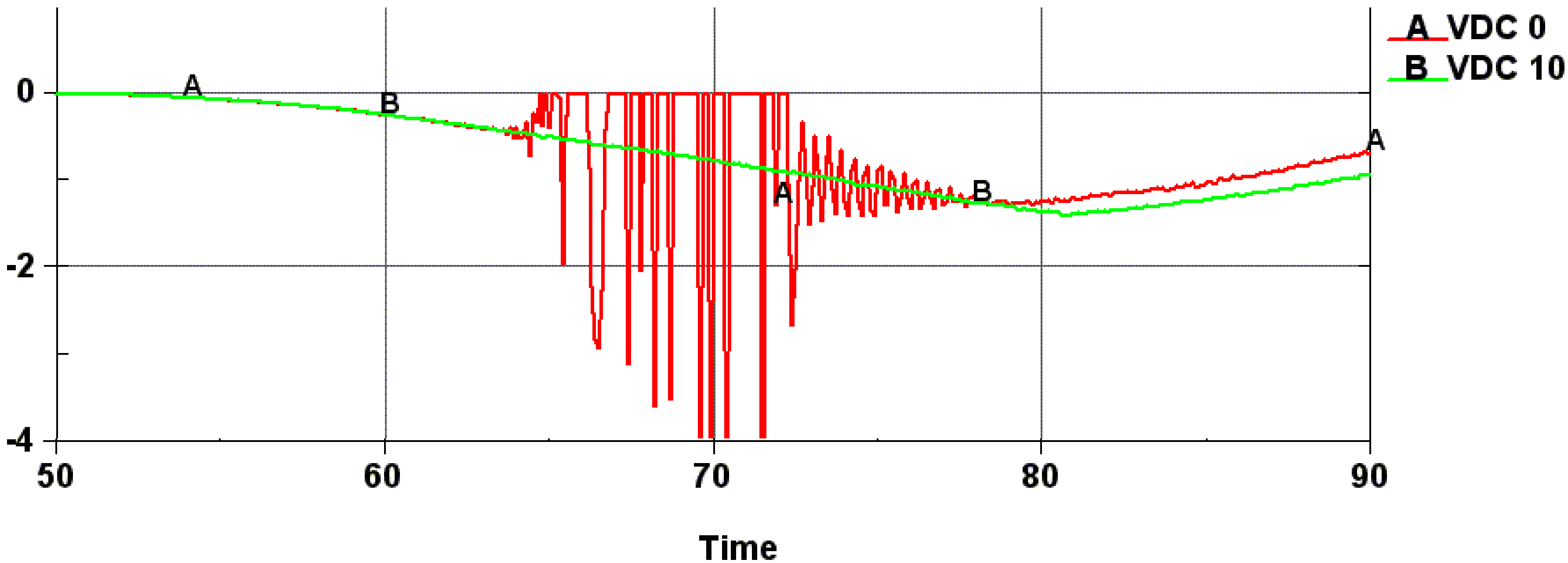
The segment mass is without mass scaling and without *element_mass !
Starting with R4.2 the segment mass can include both by setting *CONTROL_CONTACT, PSTIFF=1 (not generally recommended)

In tendency SOFT=2 is softer than SOFT=1. Sometimes setting **SFS=4** may be recommended for SOFT=2.

Contact Damping

For all penalty based contact types, a discrete damper can be defined in parallel to the normal contact stiffness. The **V**iscous **D**amping **C**oefficient is defined as parameter **VDC** in ***CONTACT**, the value is given in percent. We recommend damping between **VDC=10** or **VDC=20** for every contact definition.

Without contact damping, the contact force may be more noisy.



Constraint Contact

Only a few contact types are based on constraint method:

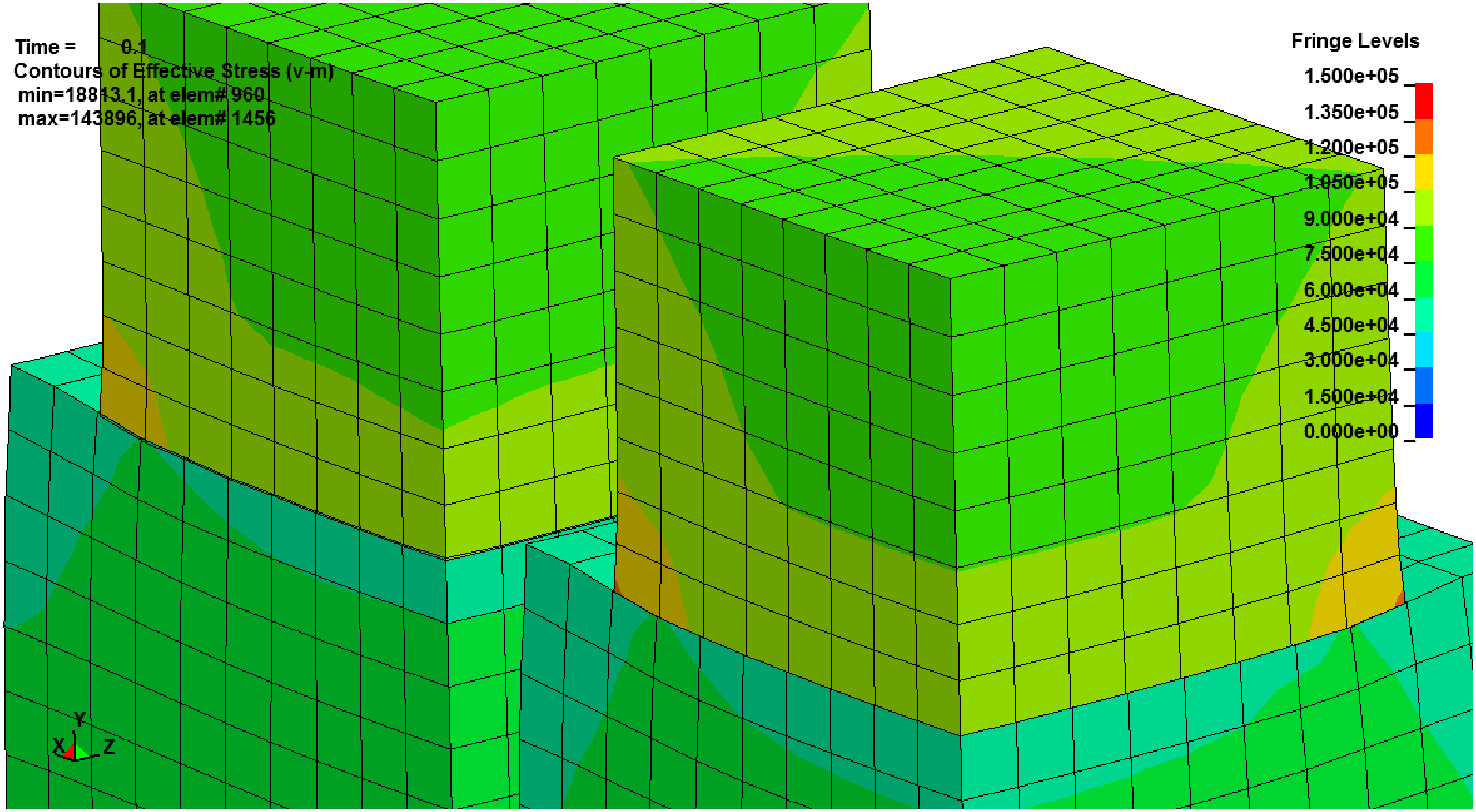
*contact_sliding_only	- in MPP penalty
*contact_constraint_nodes_to_surface	- in MPP since R5
*contact_constraint_surface_to_surface	- in MPP since R5
*contact_forming_nodes_to_surface with SOFT=4	- in MPP since R4
*contact_forming_one_way_surface_to_surface with SOFT=4	- in MPP since R4

These contact types don't apply contact forces to rigid bodies. This means they can only be used for fixed or driven rigid bodies. Because of this, sliding energy is always negative.
One node or one segment can only be used once with such contact types.

*contact_tied_...	(except the option _offset is used)
*contact_spotweld_...	(except the option _offset is used)

These contact types can not be used with rigid bodies or other constraints.

Constraint Contact



SOFT=2, SFS=4
(with TSSFAC=0.5 much less penetration)

CONSTRAINT

Constraint Contact

New since R5.1: ***CONTACT_CONSTRAINT** suported in MPP

- ➢CONSTRAINT is not SINGLE_SURFACE !
- ➢CONSTRAINT does not support SOFT=2 ! (it is always nodal based)

*CONTACT_CONSTRAINT_NODES_TO_SURFACE is recommended

-KPF is meaningless
-Feiner mesh on slave side
-Rigid bodies on master side
-no contact force is applied to rigid bodies:
→Don‘t use it with freely moving rigid bodies
→Sliding energy is will become negative (instead of external work)

*CONTACT_CONSTRAINT_SURFACE_TO_SURFACE is sometimes unstable

-KPF determines between one_way or symmetric contact

Initial Penetration

Initial Penetration

When there are initial penetrations…

- Default behavior (for SOFT=0 or 1) is to move each initially penetrating slave node back to the master surface.
  - Perturbs geometry. May initiate buckling.
  - No guarantee that all initial penetrations will be removed using this approach
- By setting IGNORE=1 (via *control_contact or *contact), initial penetrations are NOT removed for SOFT=0 or 1 contact. Rather, the contact thickness is reduced according to the penetration. The contact thickness will increase (up to a maximum of the full contact thickness) as the penetration decreases.
- Initial penetration is both:
  - Penetration in the geometry at time zero
  - The first penetration value if a contact closes at time >0
- IGNORE=1 works silent, IGNORE=2 is the same but prints penetration warnings
- IGNORE=2 is always recommended, especially for MPP

Initial Penetration – CONTACT_INTERFERENCE

The contact types - _NODES_TO_SURFACE,
- _ONE_WAY_SURFACE_TO_SURFACE
- _SURFACE_TO_SURFACE

and all segment based (SOFT=2) contact types (new since R4) are supporting the option _INTERFERENCE.
This means, than initial penetrations are not corrected and not tracked during the initialization. Instead of that, during the simulation the contact stiffness scale is continuously increased from zero to one in a reasonable time. With this procedure an initial overlapping contact can be removed by calculating the real contact forces. A load curve for contact stiffness over time is necessary, typically it consists of three lines:

```
*DEFINE_CURVE
   {id}
          0.0          0.0
          time         1.0
          endtime      1.0
```

time must be large enough to prevent unphysical vibrations.

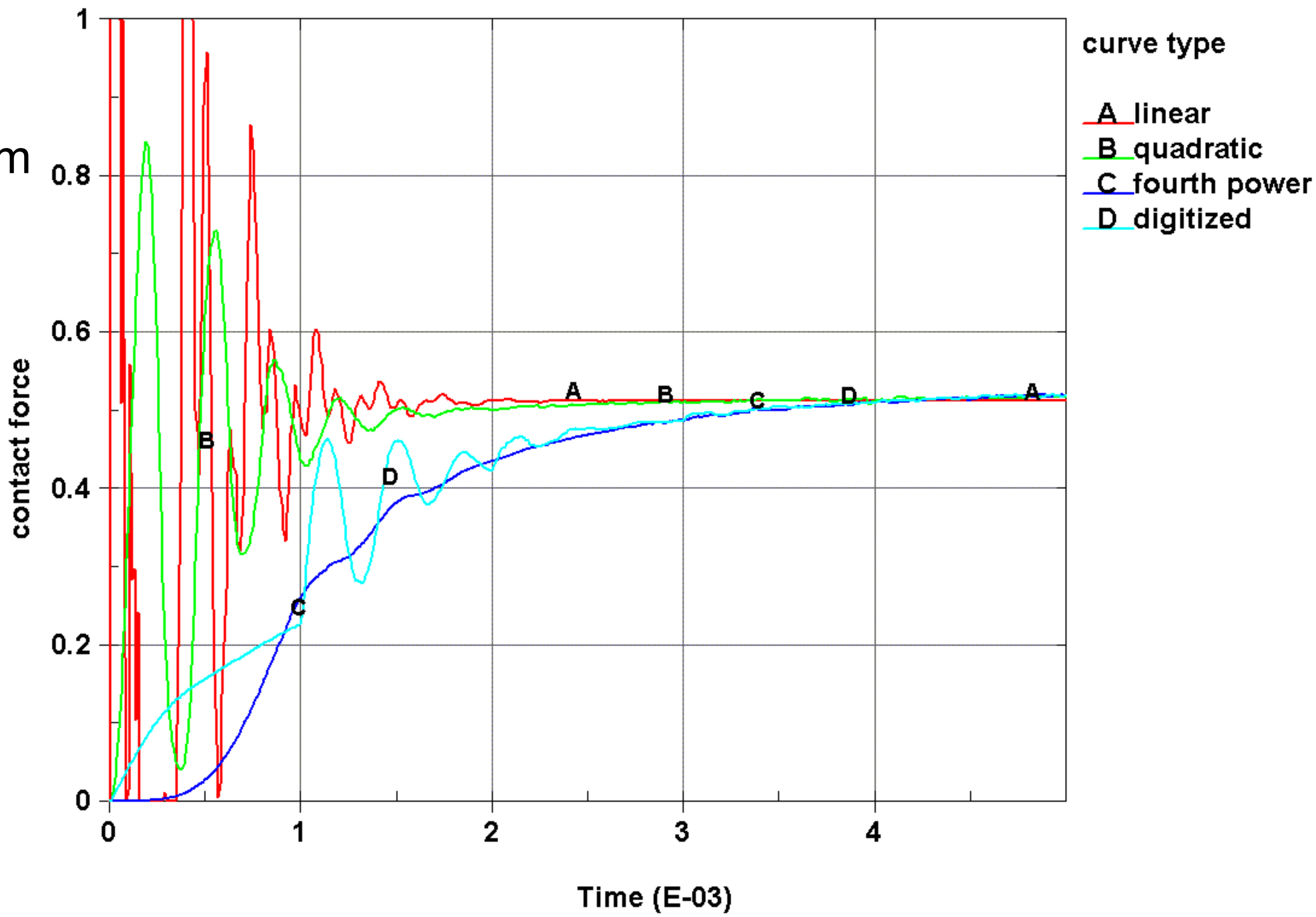
INTERFERENCE with SOFT=2

Also, relatively deep penetrations can be solved by this way.
But care must be taken not to increase the contact stiffness to much

In all four lines the contact stiffness scale factor s is increased from zero to one in 0.01 sec.

Red: s=t/0.01
Green: $s=(t/0.01)^2$
Blue: $s=(t/0.01)^4$
(all as functions)
Cyan: $s=(t/0.01)^4$
(as tabulated curve)

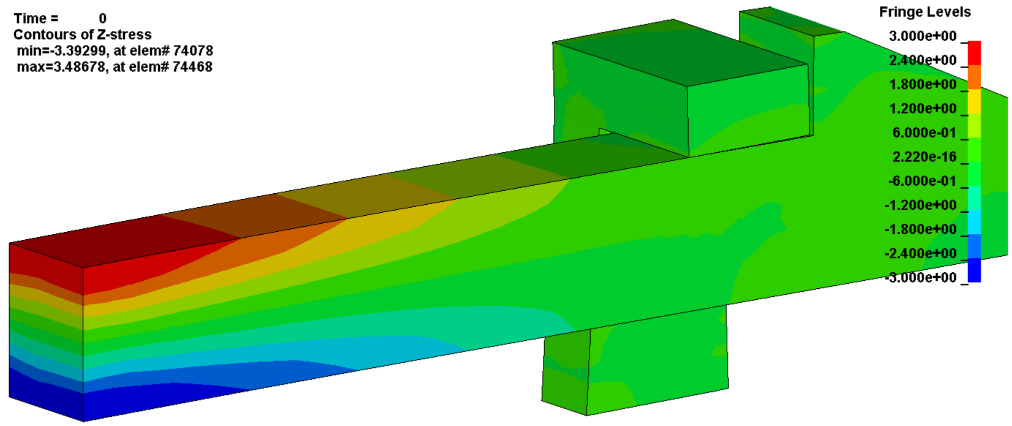
This is a nice application for the *define_curve_function.



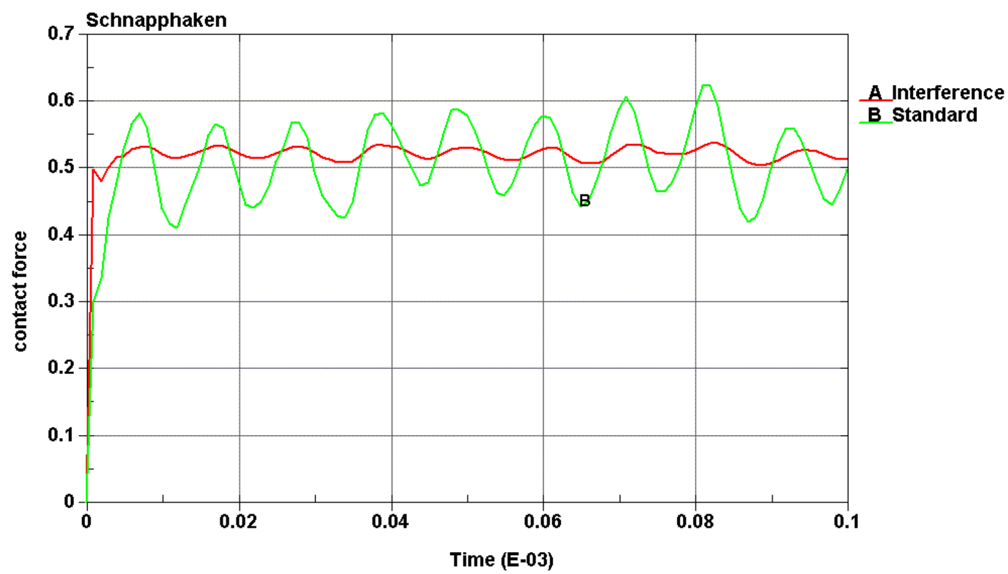
CADFEM

INTERFERENCE with SOFT=2

The problem:
Start with a prestressed model.
Because of penalty contact there will be some small initial penetrations which reduces the prestress and initiates vibrations.



The INTERFERENCE option can here be used to initialize the contact in a model, which was prestressed in a former analysis and has now initial stresses and initial penetrations.
With INTERFERENCE the penetrations are remaining. The contact stiffness has to be increased from zero to one in a very short time (one or two cycles). Then a smoother contact force will occur.



CADFEM

SMOOTH contact

*CONTACT_..._SMOOTH

- For some contact types a _SMOOTH option is available:

  *contact_nodes_to_surface_smooth - incl. _automatic and _forming
  *contact_one_way_surface_to_surface_smooth - incl. _automatic and _forming
  *contact_surface_to_surface_smooth - incl. _automatic
  *contact_automatic_single_surface_smooth - (may have problems in R3)

- A smooth curve fitted surface is used to represent the master surface. This will reduce contact noise and produce smoother results with coarse mesh.

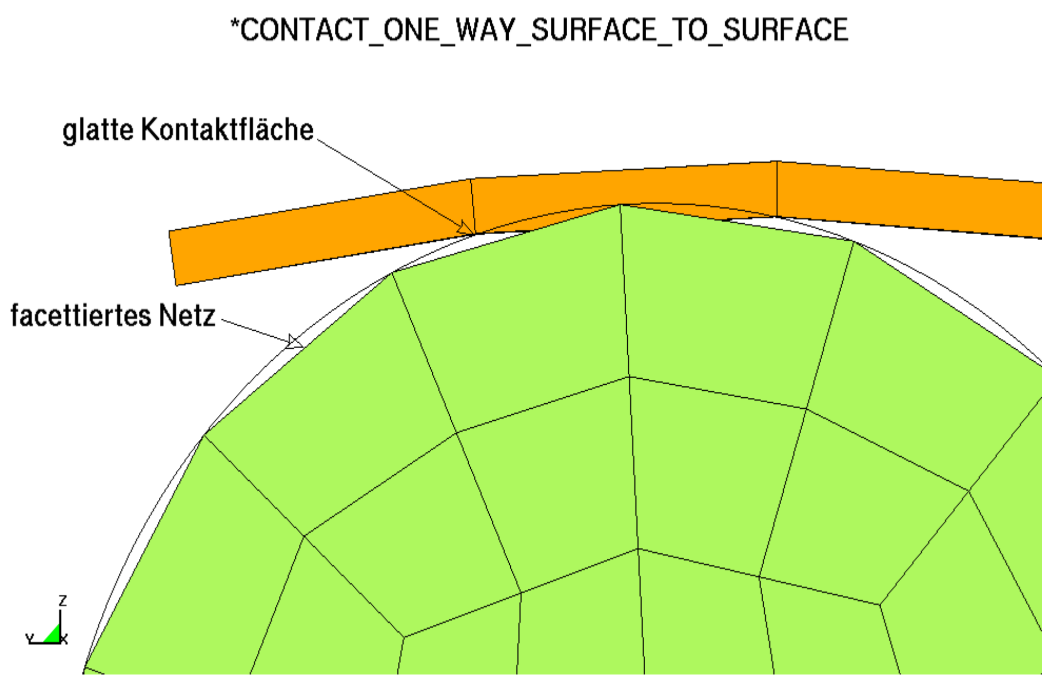
- For **SMP** only one contact type is available:
  ***CONTACT_FORMING_ONE_WAY_SURFACE_TO_SURFACE_SMOOTH**

- For **MPP** all contact types above support the _**SMOOTH**, including AUTOMATIC and FORMING option.

- Segment based contact, SOFT=2, does not support _SMOOTH.

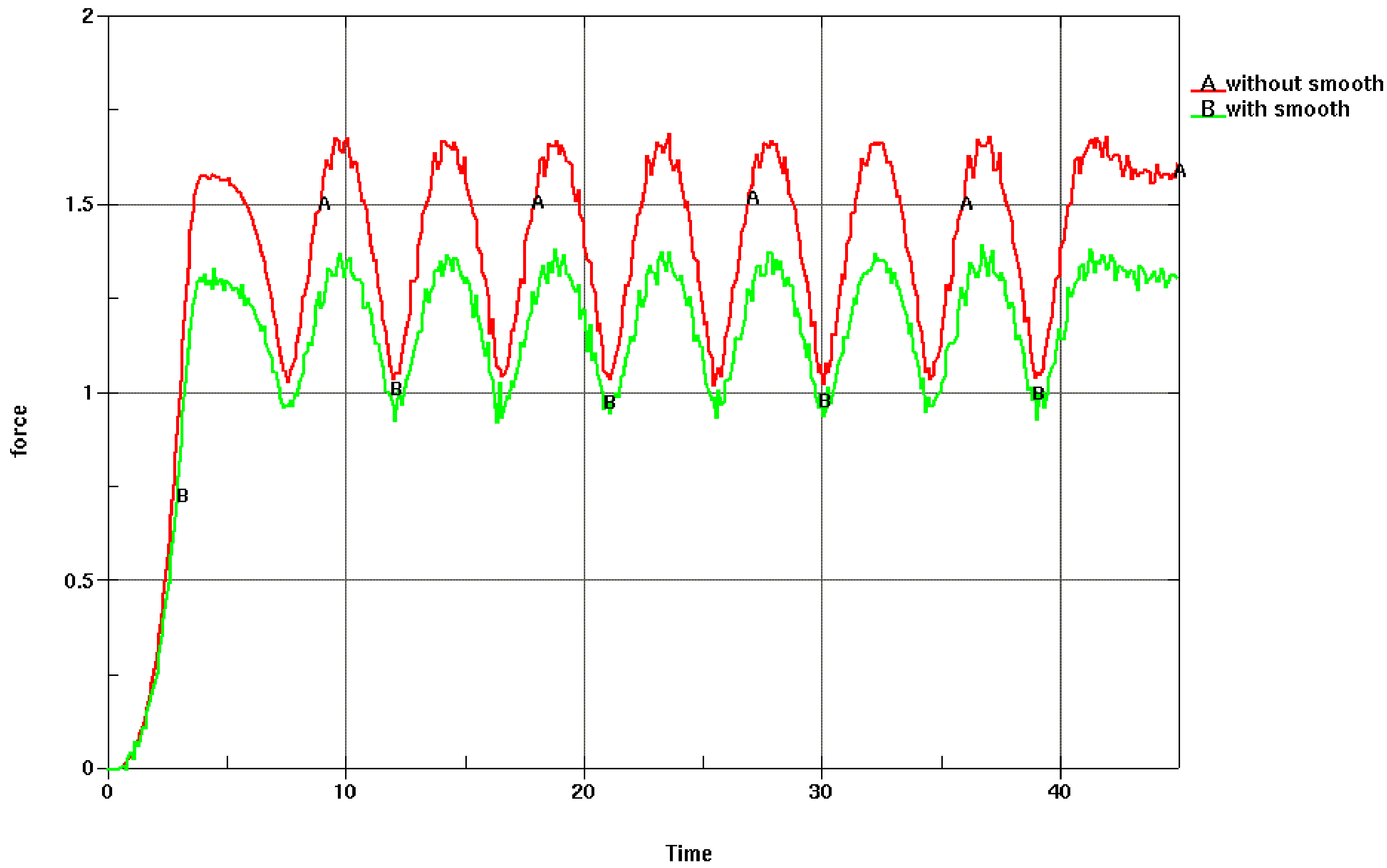
*CONTACT_..._SMOOTH

- The ONE_WAY_SURFACE_TO_SURFACE gives the smoothest result because only the master surface is smoothed.
- The SINGLE_SURFACE and SURFACE_TO_SURFACE contact types are also supported for MPP but because only single nodes are in contact the response is not as smooth as expected.
- The parameter FLANGL on optional contact card C controls an angle between two contact segments which will not be smoothed.
- The SMOOTH option increases the computation time for contact by a factor 5-20 , except for FORMING contact.
- Memory is not increased.



*CONTACT_..._SMOOTH

*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_SMOOTH



CADFEM

Contact with beams

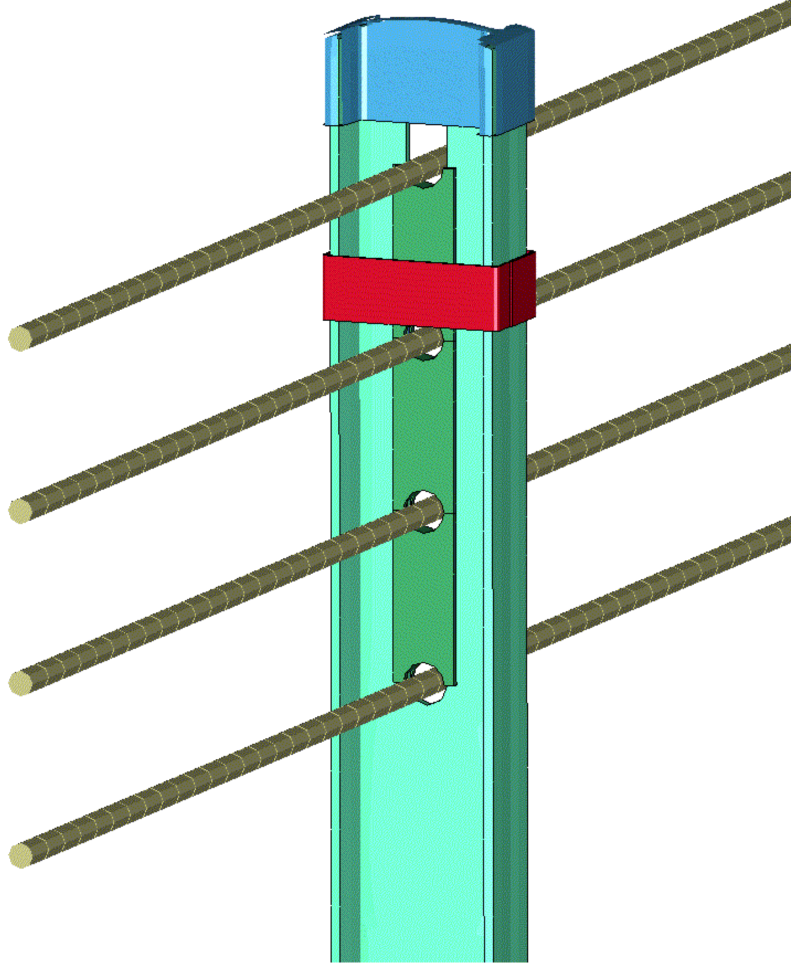
Beams to Surface Contact

In standard contact, beams represented only by their nodes. For the SOFT=2 contact beams are ignored.

Only the old *CONTACT_AUTOMATIC_GENERAL was able to take beams as cylindrical contact surfaces.

The need for simple and efficient beam to surface contact:

- Seatbelts or beams contacting shell edges
- Cables adjacent to a structural surface
- Human body modeling of muscles and tendons interacting with skeleton
- Interaction of woven fabrics



Beams to Surface Contact

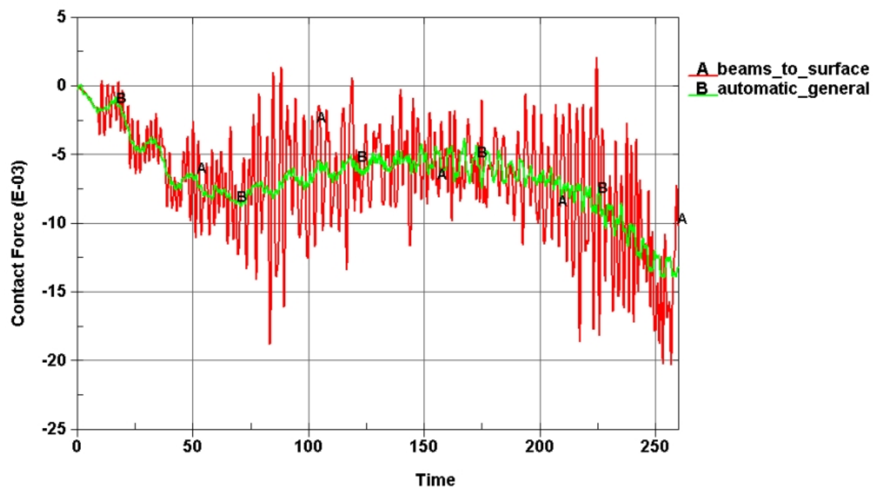
*CONTACT_AUTOMATIC_BEAMS_TO_SURFACE

- Accuracy over node to surface contact types: Provides continuous force distribution due to beam contact
- Should be faster than AUTOMATIC_GENERAL
- Avoids beam to beam contact checking of the AUTOMATIC_GENERAL type

  but this is always possible with CPARM8=1 in *CONTACT_..._MPP

Current experience:

- AUTOMATIC_BEAMS_TO_SURFACE is more noisy than AUTOMATIC_GENERAL
- Not always stable yet !?



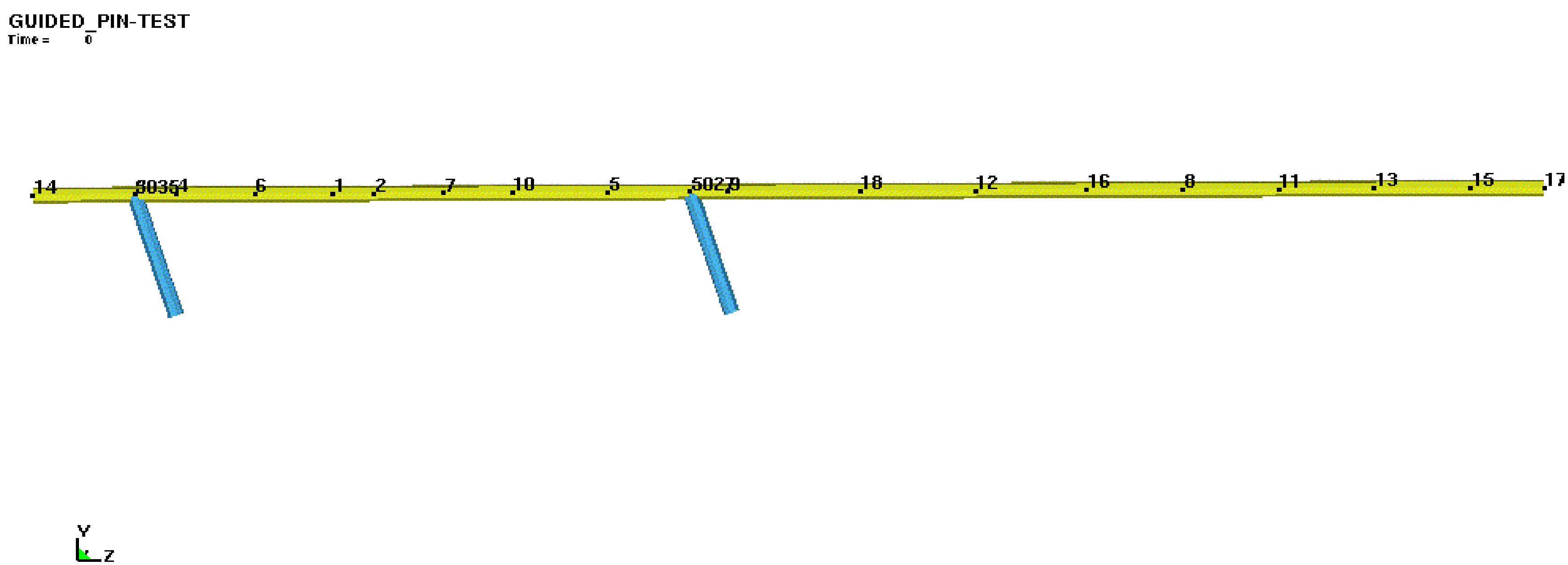
CONTACT_GUIDED_CABLE

A very simple but sometimes very helpful contact type was added:

***CONTACT_GUIDED_CABLE**

This is not directly a contact, it is more like a cylindrical joint with deformable parts. It is also comparable with a slipring (*element_seatbelt_slipring).

For some applications it may be favoured to replace translational or cylindrical joints because it describes the physical behaviour much better and is stable also for arbitrary large relative displacements.



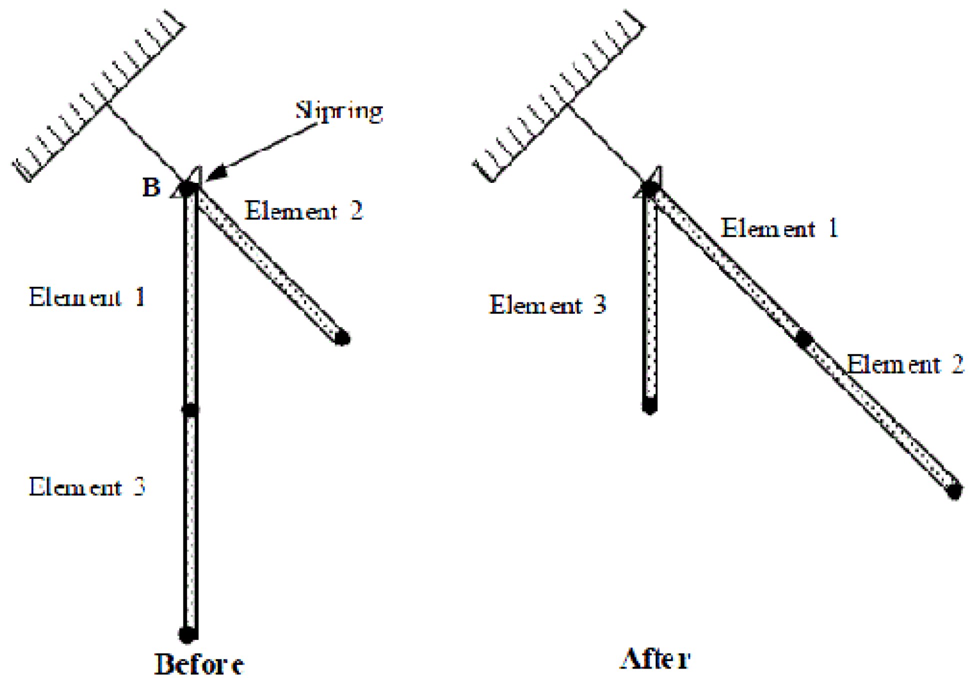
CONTACT_GUIDED_CABLE

The *CONTACT_GUIDED_CABLE

Is not really good for cables, especially when the cable is guided through an eyelet at an acute angle.

Here a much more realistic behavior can be modeled with seatbelt elements and sliprings:

*element_seatbelt
*element_seatbelt_slipring



general tips

General Tips

- ***CONTACT_AUTOMATIC_SINGLE_SURFACE with SOFT=1 is recommended for most explicit impact simulations**
  - Perhaps the most efficient and reliable contact
  - One ‘global’ contact is not significantly more expensive than several small ones, especially for MPP
  - Use *contact_force_transducer to monitor forces
  - SOFT=2 is now stable enough to become the new recommendation, it’s more accurate but also more expensive
- **Use *CONTACT_AUTOMATIC_GENERAL sparingly where needed**
  - recommended for shell edge-to-beam and beam-to-beam contact
  - If there are interior shell edges in contact, try *contact_automatic_general_interior as an alternative to adding null beams to shell edges
- **Non-automatic contacts are generally reliable for simple geometries where contact orientation can be established reliably from the outset**
  - Shell thickness consideration is <u>not</u> mandatory
  - Correct contact orientation is critical (check it)
  - Often used (and necessary) for implicit simulations

General Tips

- If contact breaks down for very thin shells, increase the contact thickness (to no less than 0.5 mm) – AUTOMATIC types only
- If a one-way contact is used: Make coarser mesh the master side,
  make the more stiff part as the master side
  make the larger part as the master side
- Avoid redundant contact specification
- Default contact stiffness may have to be changed for contact between disparate meshes or materials
  - Modify penalty scale factor on Card 3 of *contact
  - Set SOFT=1 on optional card A in *contact
- Use segment-based contact (SOFT=2) if geometry contains angular surfaces, i.e., sharp corners and to consider square shell edges:
  - SOFT=2 contact may work in situations where other contacts fail.
  - Set DEPTH=5 (or 15) if SOFT=2 contact must also treat edge contact.
- Use always a small contact damping: VDC=10.0 (i.e. 10%)

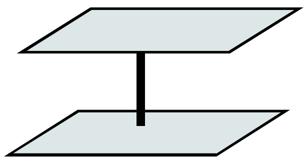
General Tips

- Set IGNORE=2 to get report of initial penetrations
  - Crossed shell midplanes never OK and IGNORE may not help here
- Default bucket sorting interval is generally OK. For the most contacts the sort is performed every 100 cycles. This can be changed using *contact or *control_contact. High velocity impacts may require a more frequent bucket sort.
- Rigid parts should have reasonable mesh refinement to adequately distribute contact forces (and to give accurate mass properties)
- Specify TIED contacts with node / segment sets to control what should really be tied.
- In case of trouble:
  - try smaller time step size (tssfac<0.9 in *control_timestep)
  - try smaller contact stiffness (depending on SOFT=0,1,2)
  - use finer mesh (a closed contact should include several elements)

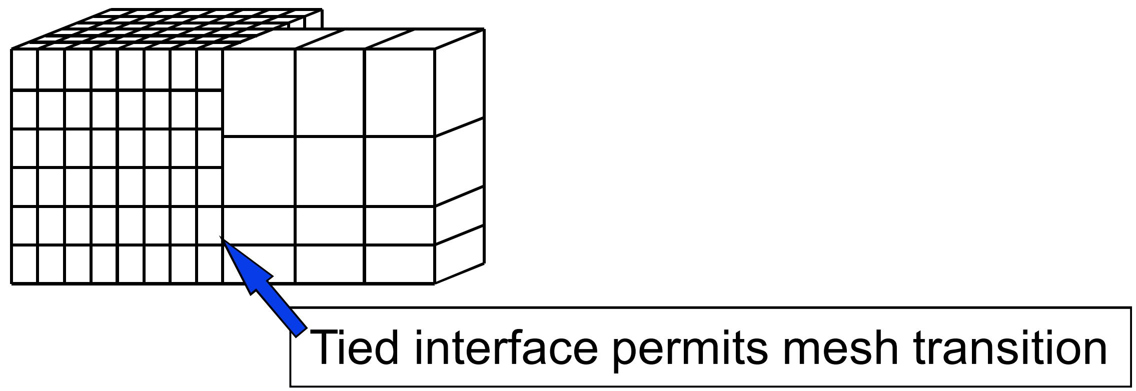
permanent connection

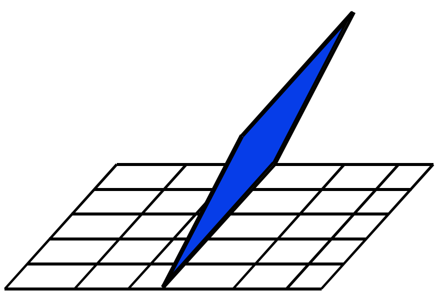
*CONTACT_TIED

Tied Contact Applications

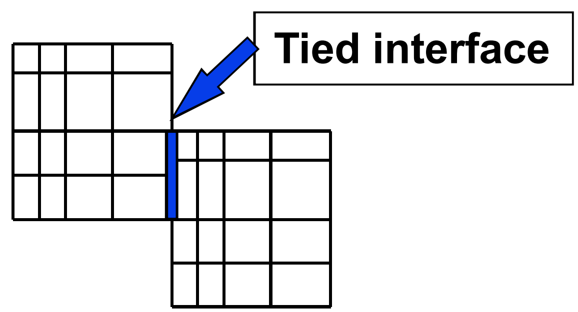


Spotweld to shell surface





Shell edge to surface



Tied connections

Contact types for permanent connection:

- *CONTACT_TIED_NODES_TO_SURFACE
- *CONTACT_TIED_NODES_TO_SURFACE_OFFSET
- *CONTACT_TIED_NODES_TO_SURFACE_CONSTRAINED_OFFSET

- *CONTACT_TIED_SURFACE_TO_SURFACE
- *CONTACT_TIED_SURFACE_TO_SURFACE_OFFSET
- *CONTACT_TIED_SURFACE_TO_SURFACE_CONSTRAINED_OFFSET

- *CONTACT_TIED_SHELL_EDGE_TO_SURFACE
- *CONTACT_TIED_SHELL_EDGE_TO_SURFACE_BEAM_OFFSET
- *CONTACT_TIED_SHELL_EDGE_TO_SURFACE_CONSTRAINED_OFFSET

The TIED_SHELL_EDGE family transfers nodal displacements and rotations. The others only transfer displacements.

Tied connections

New Contact types for permanent connection:

- *CONTACT_AUTOMATIC_SINGLE_SURFACE_TIED

  Mainly developed for linear implicit analysis like modal analysis, but can also be used in explicit as a penalty based TIED contact to connect many different parts.

- *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_MORTAR_TIED

  Mainly developed for nonlinear implicit analysis to improve convergence, but can also be used in explicit as a penalty based TIED contact to get a smoother contact stress distribution for dissimilar meshes.
  This contact is still under development and is currently very expensive in explicit.

These two new contact types only transfer displacements.

Tied connections

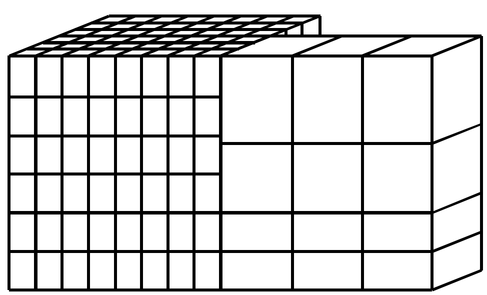
- Good for tying parts with disparate meshes. This may make model generation much easier. But please note: A consistent mesh is more accurate than a tied connection.
- In each case, the nodes defined on the slave side are connected with a segment (an element face) on the master surface

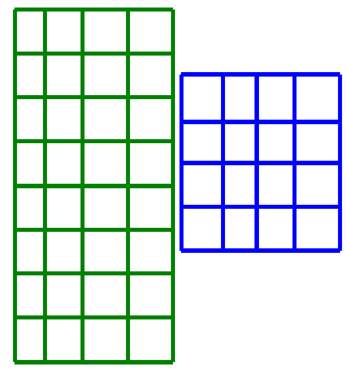
Note: There is no symmetric contact option, tied_surface_to_surface should be read as tied_one_way_surface_to_surface.

This means, it is very important who is the master and who is the slave.

The same criteria can be used as for sliding interfaces:

- The finer mesh should be the slave.
- In case of overlapping parts, the larger should be the master.





Tied connections

- Most tied contacts impose kinematic constraints
  - Constraint-based tied contacts are only for deformable bodies

- The options **_OFFSET** or **_BEAM_OFFSET** invoke penalty-based treatment
  - OK for deformable bodies and rigid bodies

  Note: - **CONTRAINT_OFFSET** is of course constraint-based

  - **BEAM_OFFSET** is penalty based, but includes rigid beams. In some cases it may have problems with rigid bodies.

- Penalty-based TIED contacts have a contact stiffness similar to normal contact. Stiffness can be scaled with the known parameters (slsfac, sfs, sfm). SOFT=1 is not supported for TIED contact.
  Contact damping is recommended here (VDC=10)
- Constraint-based contacts are totally rigid. VDC and stiffness scaling has no meaning.

Tied connections

- Constrained-based TIED contacts are recommended because they transfer displacements and forces without a distortion based on a finite contact stiffness.

  There are two disadvantages of Constrained-based TIED contacts:

  1) no connection with rigid bodies is possible

  2) one node or one segment can not be used in two TIED contacts

For 2) it is clear that one node can only be used one times on a slave side of a TIED contact. This means if one node is listed on the slave side of a TIED contact, he is not allowed to be listed on another slave side of a TIED. LS-DYNA does not check, if a node is really used.

But it should be allowed to list a node on more than one master sides. Unfortunately this is only true for the MPP version of LS-DYNA. The SMP version will print an error message:

*** Warning tied node #268 on the master side of surface #2 type # 6 also belongs to interface #1

TIED CONSTRAINT IS RELEASED.

Tied connections

- Criteria for tying:

  For each slave node a master segment is searched that

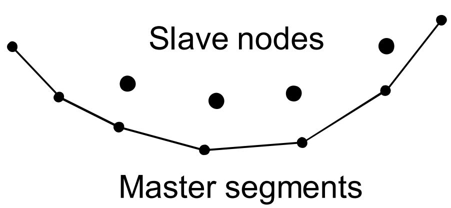
  - an orthogonal projection from the node of the master segment is possible
  - the projected distance is within a tolerance

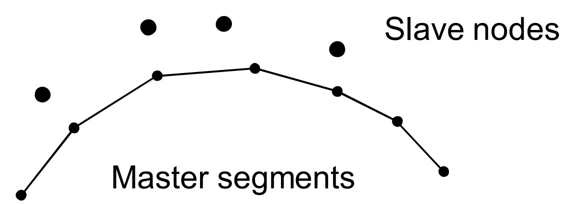
  For concave surfaces for one node may be more than one segment is possible.

  Here the first one is taken.

  For convex surfaces may be no segment can be found. A segment extension is done for such situations. The size of the segment extension can be controlled with *CONTACT, MAXPAR (optional card A). The default is MAXPAR=1.006. This means that the segment becomes 0.6% larger.

  Sometimes it would be necessary to increase this value to ensure that all nodes are connected. To large values of MAXPAR may lead to instabilities.





Tied connections

Search depth:

Slave nodes are only connected if they are near to the master segment, so the projected distance δ is important. Nodes are tied if:

δ_1 = 0.6*(master_thickness+slave_thickness)
δ_2 = 0.05*(smallest diagonal of master segment)

→ $\delta < \max(\delta_1, \delta_2)$

- for solid elements the thickness is zero
- in *CONTACT, SST und MST can be an optional slave or master thickness for shells
- if SST and MST are negative, than only δ_1 is used as the distance criterion.
- for each slave not which is not connected, a warning message is printed (see file messag)

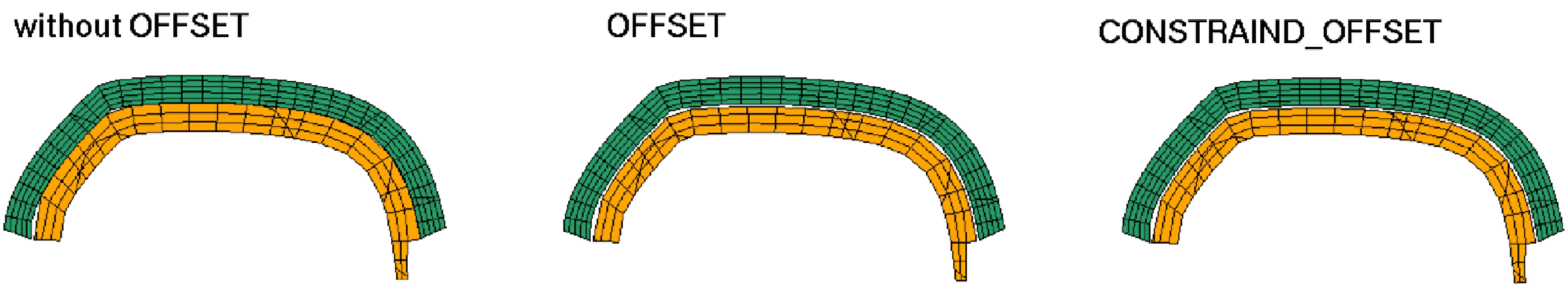
Tied connections with or without offset

<u>Connecting solid elements:</u>

*CONTACT_TIED (without _OFFSET) require that slave nodes are located exactly on the surface of the master segment. If there is any gap in the geometry, the contact will close this gap by moving the slave node in the initialization phase. This is a change in the geometry without strains and stresses. With this approach, the force and moment balance is always correct.

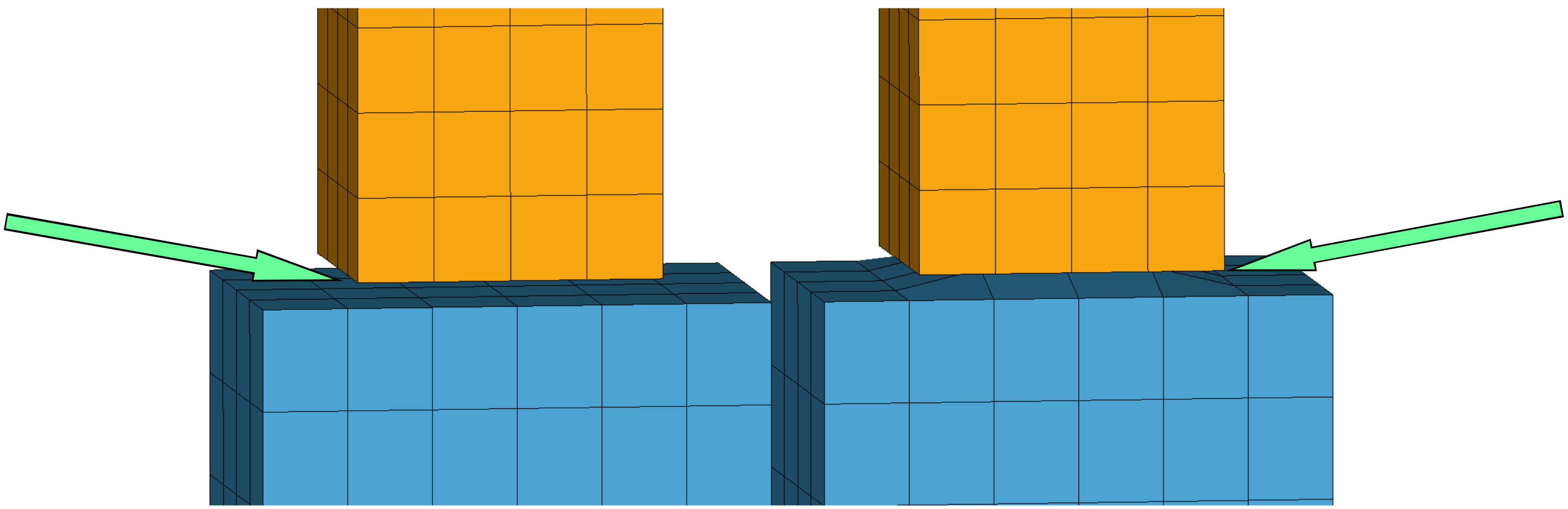
With the option _OFFSET existing gaps remain but the contact will only transfer forces, no moments. Depending on the size of the gaps and depending on how much the structure rotates the moment balance is wrong.

With the option _CONSTRAINED_OFFSET existing gaps remain and the contact will transfer forces and moments right.



Tied connections with or without offset

Connecting solid elements:



tied contact with _OFFSET or _CONSTRAINED_OFFSET ↳ nodes remain at original position	tied contact without _OFFSET ↳ nodes are moved onto master segment

Recommended:
***CONTACT_TIED_NODES_TO_SURFACE_CONSTRAINED_OFFSET**

Recommended:
***CONTACT_TIED_NODES_TO_SURFACE**

Tied connections with or without offset

Connecting shell and beam elements:

Similar to solids:
***CONTACT_TIED_SHELL_EDGE** (without _OFFSET) require that slave nodes are located exactly on the midsurface of the master shell segment. If there is any gap in the geometry, the contact will close this gap by moving the slave node in the initialization phase. This is a change in the geometry without strains and stresses. With this approach, the force and moment balance is always correct.

With the option _OFFSET existing gaps remain but the contact will only transfer forces and moments without consideration of that gap. Depending on the size of the gaps and depending on how much the structure rotates the moment balance is wrong.

With the option _BEAM_OFFSET existing gaps remain and the contact will transfer forces and moments right.

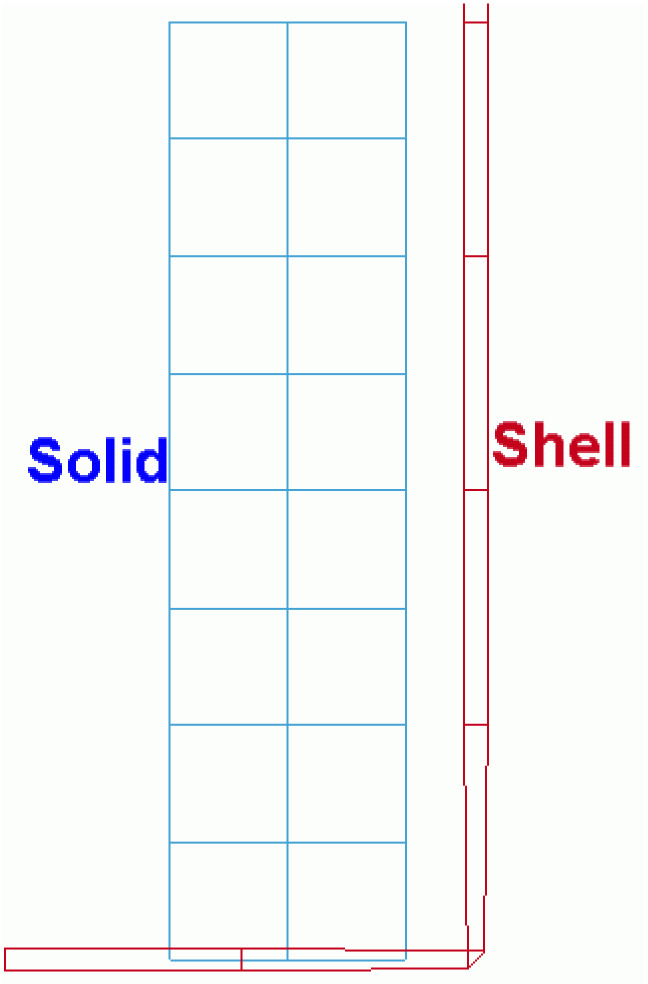
Recommended:
***CONTACT_TIED_SHELL_EDGE_TO_SURFACE_BEAM_OFFSET**

Tied connections with or without offset

Connecting shell and beam elements:

If slave nodes are tied on master segments from shell elements without an _OFFSET, then these nodes would be located deeply inside the shell volume.
All AUTOMATIC contact types will ignore such nodes which are located directly in the mid surface of the shell or in a range of $\pm 0.1*(d_{Master}+d_{Slave})$. This will not be seen as an initial penetration but a warning is printed.
LS-DYNA does not check if such nodes really tied!

For solids there is not such a logic. A _OFFSET option should be used here or a more detailed contact definition to exclude the penetrated nodes from other contacts.



Practical Guidelines for Tied connections

- Specify the contact using node sets on slave side, containing only these nodes which really should be connected. Each node from the slave side which could not be connected produces a warning message. Master side can be part, part set or segment set.
- Use _TIED_SHELL_EDGE_TO_SURFACE… types when tying shells or spotweld beams, this includes tying of rotational DOF. All other TIED contact types will connect translational DOF only.
- If a physical offset between tied surfaces is desired, _CONSTRAINED_OFFSET or _BEAM_OFFSET are preferred as these will transfer moments in a beam-like manner
  - _constrained_offset is constraint-based and thus cannot be used with rigid bodies
  - _beam_offset is an option only with tied_shell_edge_to_surface
- do not use a simple _OFFSET option (_TIED_NODES_TO_SURFACE_OFFSET),especially if the offset distance is not negligible, because of moments are not transferred

Tied connections

Other methods for Tying Parts

- *CONSTRAINED_*option* offers alternatives to tied contacts in tying nodes to other nodes or surfaces
  - _SPOTWELD - rigid massless beam, two nodes only
  - _GENERALIZED_WELD… - rigid connection for more than two nodes
  - _NODAL_RIGID_BODY - general kind of rigid body by nodes
  - _INTERPOLATION - RBE3
  - _EXTRA_NODES… - connect deformable or massless nodes with existing rigid bodies
  - _POINTS - connect two shells
  - _TIE-BREAK - edge-to-edge tying of shells with failure
  - _TIED_NODES_FAILURE - node-to-node coupling with failure

tied contact with failure

*CONTACT_AUTOMATIC_..._TIEBREAK

Modeling of bonded joints

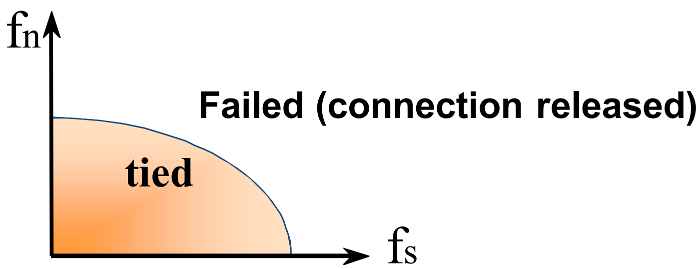
Two families of contact to model adhesive bond: **_TIEBREAK**

A) *CONTACT_TIEBREAK_NODES_TO_SURFACE
*CONTACT_TIEBREAK_NODES_ONLY
*CONTACT_TIEBREAK_SURFACE_TO_SURFACE

The *CONTACT_TIEBREAK can fail because of a combination of normal and shear stress. After failure it works like a *CONTACT_SURFACE_TO_SURFACE or _NODES_TO_SURFACE. The _NODES_ONLY will not longer exist after failure.

The TIEBREAK contact types use a Penalty method. Therefore they can also be used in combination with rigid bodies.
This family of *CONTACT_TIEBREAK pretty old and is not particularly robust.

→ generally not recommended



Modeling of bonded joints

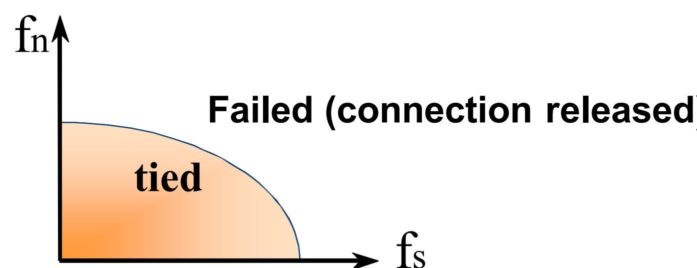
Two families of contact to model adhesive bond: **AUTOMATIC_…_TIEBREAK**

B) *CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE_TIEBREAK
*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_TIEBREAK

- These are the modern alternative to the old *CONTACT_TIEBREAK.
- The ONE_WAY option uses a different method to calculate failure stresses.
- After failure they are working like the same contact without _TIEBREAK.
- Segmentbased contact (SOFT=2) is not supported.
- Penalty method is used, therefore also rigid bodies are allowed.
- With parameter OPTION a lot of different failure models are available:

simplest case:

OPTION=2 : brittle failure with normal and shear stress



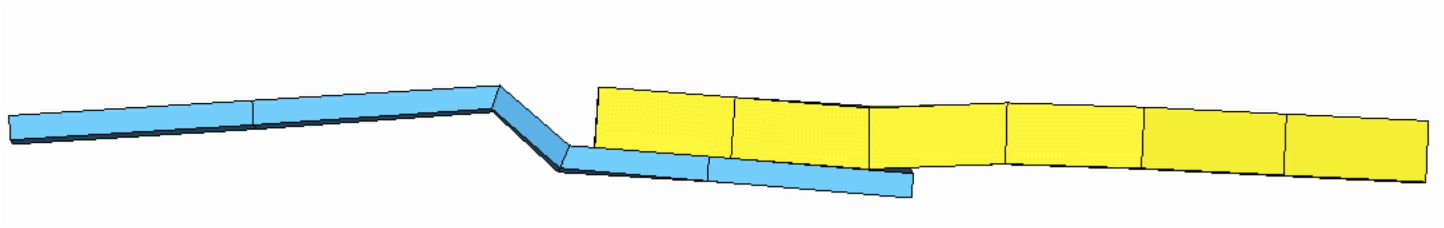
Modeling of bonded joints

B) *CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE_TIEBREAK
*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_TIEBREAK

The meaning of IGNORE:

- In *CONTROL_CONTACT only IGNORE=1 can be set, not IGNORE=2.
- In local *CONTACT definition, optional card C, IGNORE is completely supported
- IGNORE controls, if initial penetrations or initial gaps are corrected, the latter one only if the gap is not too large.
- If IGNORE=0 (or -1) then the connecting nodes moved directly onto the master segments, taking account of the shell thickness
- If IGNORE=1 then the nodes are conected where they are, if not to far away.

MPP-LSDYNA ignores currently all IGNORE settings and uses always IGNORE=1



Modeling of bonded joints

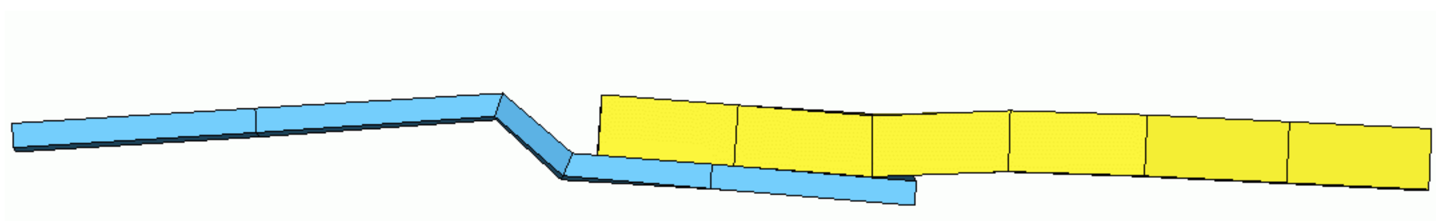
B) *CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE_TIEBREAK
*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_TIEBREAK

Up to what distance connection is made?

SMP: 0.9*(thickness_master+thickness_slave)

MPP: 0.5*(thickness_master+thickness_slave) + 0.01*master_segment_diagonal

Attention: These are different values than for *CONTACT_TIED.



Modeling of bonded joints

B) *CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE_TIEBREAK
*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_TIEBREAK

Contact thickness and OFFSET:

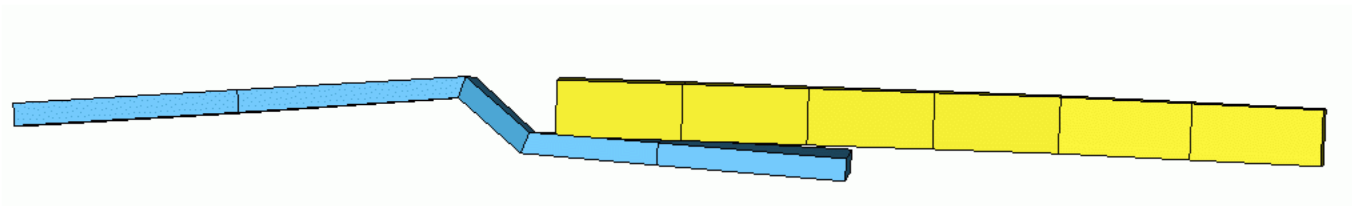
If shells are connected, they may have a distance according to the sheet thickness.
If solid elements are connected, they can only have a distance if IGNORE=1 is used.

The transfer of moments due to offsets is different depending on OPTION:

OPTION= 1,2,3,5,6,7,9 : Moments are not calculated.

OPTION= -1,-2,-3, 8,10,11 : Moments are not calculated and taken into account

OPTION= -1,-2 and -3 are currently not available n MPP-LSDYNA.



Modeling of bonded joints

All previously mentioned tiebreak contact definitions have in common, that their failure behavior is ideally brittle. I.e. they do not dissipate energy during the opening of the crack. In reality, the crack development as well as the crack propagation dissipates energy due to the creation of a new surface (recall the theory of fracture mechanics).

In order to account for this effect, LS-971 offers some new contact options which have the ability to dissipate energy during the crack occurrence. This is typically done by defining a linear force-crack-opening relation in the contact definition. Thereby the transmitted forces respectively stresses by the contact are successively reduced to zero until the maximum crack opening displacement defined by the user is reached. After this point the contact still works as a *contact_automatic_surface_to_surface in case that both contact pairs get in contact again.

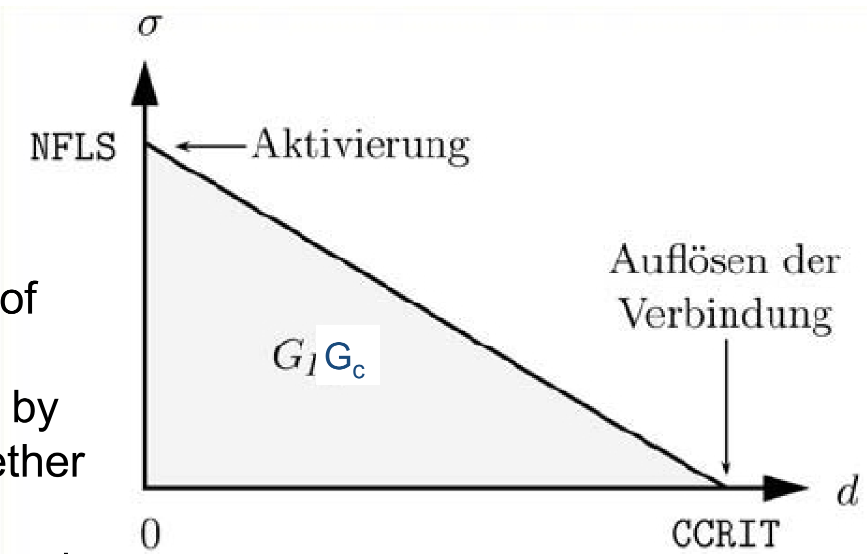
Contact option 5	→ plastic yield with damage function (for solids and shells)
Contact option 6 and 8	→ linear damage model (6 for solids, 8 for shells)
Contact option 7 and 10	→ DYCOS discrete crack model (7 for solids, 10 for shells)
Contact option 9 and 11	→ extended option 7: discrete crack model with damage, similar to *MAT_138: *MAT_COHESIVE_MIXED_MODE (9 for solids, 11 for shells)

Modeling of bonded joints

```
*CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE_TIEBREAK_ID
$#      CID        HEADING
          1        DCB
$#     SSID        MSID      SSTYP      MSTYP     SBOXID     MBOXID        SPR        MPR
          2           1          3          3                                1          1
$#       FS          FD         DC         VC        VDC     PENCHK         BT         DT
       0.10         0.0        1.0        2.0       10.0          0        0.01.0000E+20
$#      SFS         SFM        SST        MST       SFST       SFMT        FSF        VSF
        0.0         0.0        0.0        0.0        0.0        1.0        1.0        1.0
$#   OPTION        NFLS       SFLS       CCOD
          8         1.0        1.0       0.05
```

NFLS	normal stress at failure
SFLS	shear stress at failure
CCOD	critical crack opening distance

Parameters NFLS and SFLS control the onset of failure, I.e. the crack occurrence, while CCOD/CCRIT controls the energy G_C released by the crack formation. There is no distinction whether the cracking is in mode I (tensile) or II (shear). Consequently always the same energy is released.



Modeling of bonded joints

```
*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_TIEBREAK_ID
$#     cid    title
         1    DCB
$#    ssid       msid      sstyp      mstyp     sboxid     mboxid        spr        mpr
         2          1          0          3          0          0          1          1
$#      fs         fd         dc         vc        vdc     penchk         bt         dt
  0.100000      0.000   0.000000   0.000000  10.000000          0      0.0001.0000E+20
$#     sfs        sfm        sst        mst       sfst       sfmt        fsf        vsf
      0.1        0.1
$#  OPTION       NFLS       SFLS        PHI       GI_C      GII_C
         7  52.000000  38.000000   0.500000     10.2       12.3
```

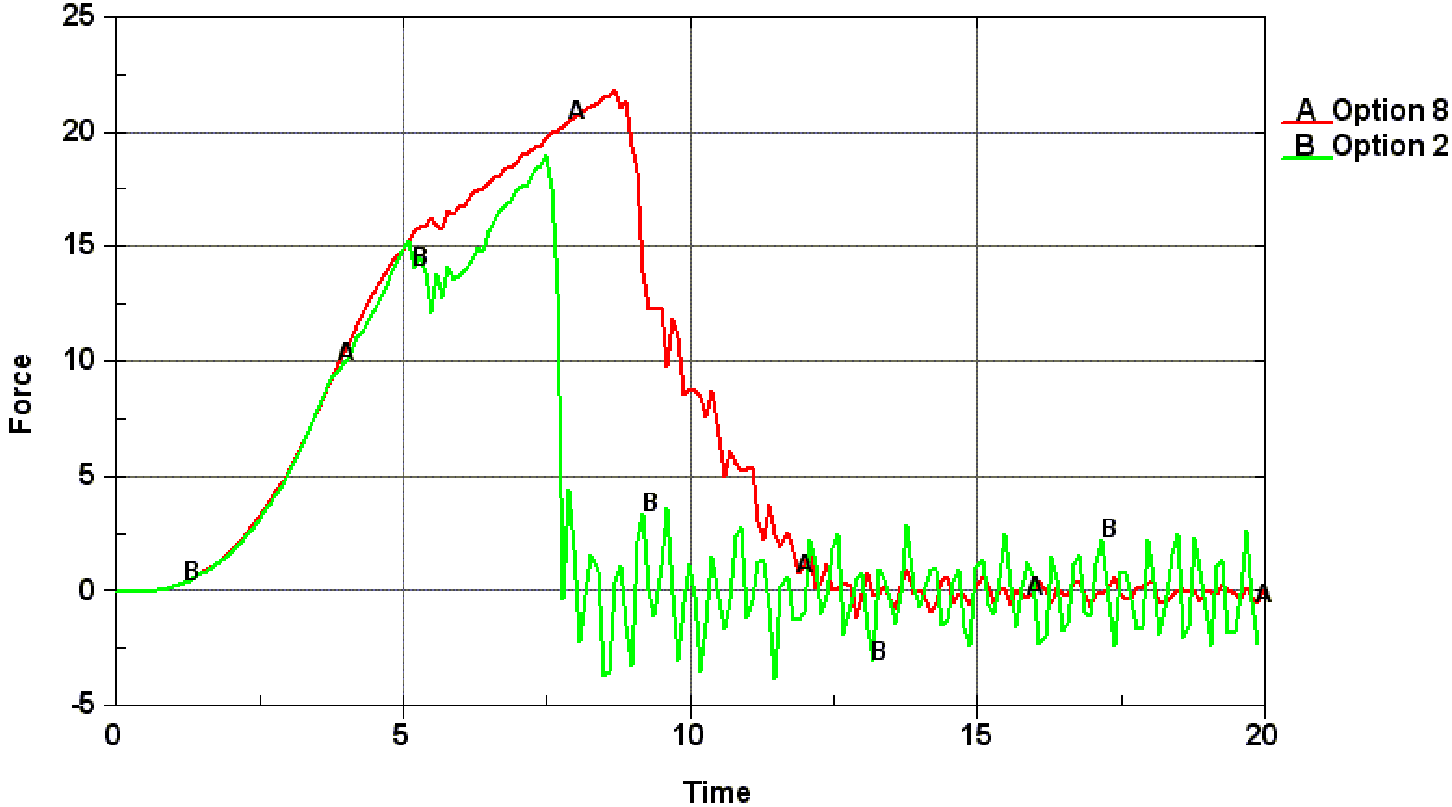
NFLS	normal stress at failure
SFLS	shear stress at failure
PHI	friction angle (degrees)
GI_C	fracture energy mode I (tensile)
GII_C	fracture energy mode II (shear)

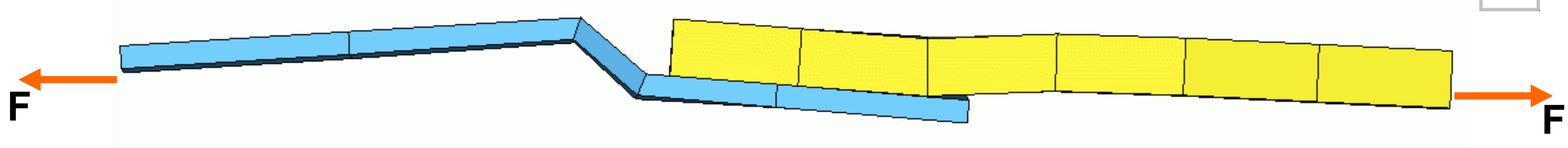
The DYCOS model is basically the same as the one previously described. It is based on an elastic-brittle damage formulation (one damage parameter), whereby the stiffness degradation is control by critical energy release rates. Additionally it has some important possibilities like:

1. Different energy release rates for mode I and II fracture
2. Lateral forces/stress transmission in case of normal compression due to friction between the two contact partners.

Modeling of bonded joints

*CONTACT_AUTOMATIC_..._TIEBREAK



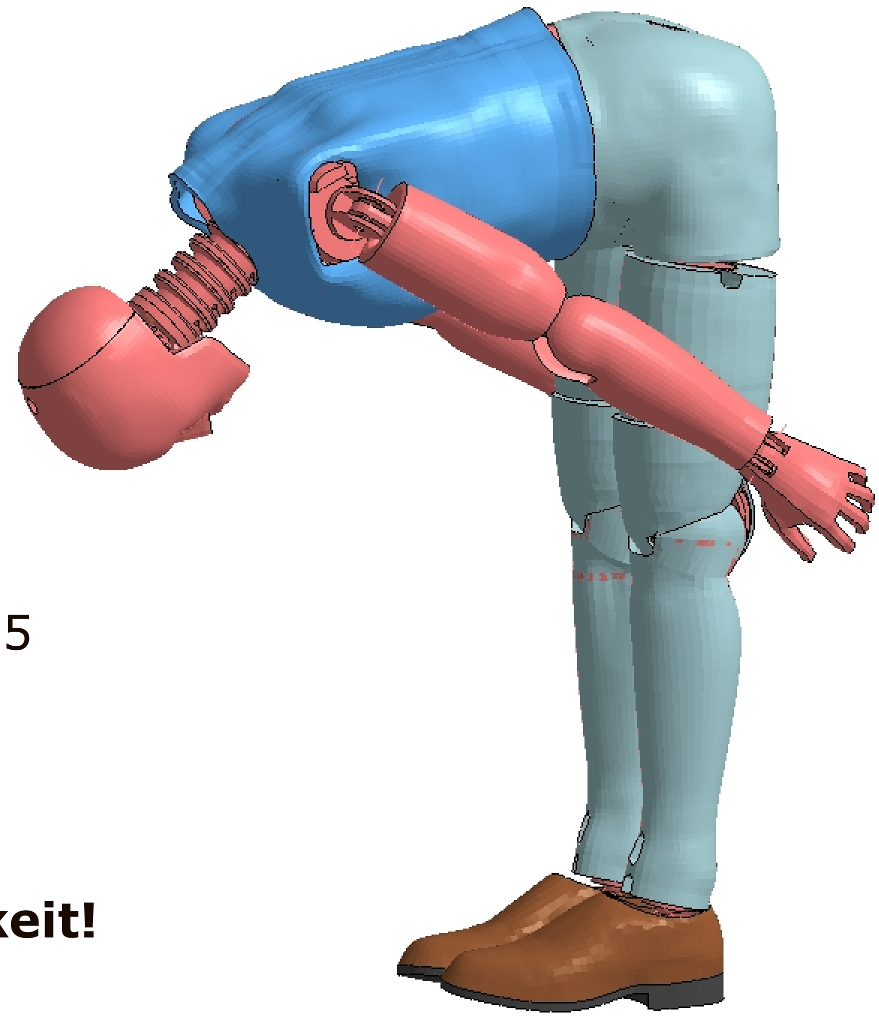


Kontakt

CAD-FEM GmbH
Marktplatz 2
DE-85567 Grafing b. München

Telefon +49 (0)8092-7005-0
Telefax +49 (0)8092-7005-77
E-Mail info@cadfem.de
www.cadfem.de

Support-Hotline: +49 (0)8092-7005-55
Support-Mail: support@cadfem.de



Vielen Dank für Ihre Aufmerksamkeit!