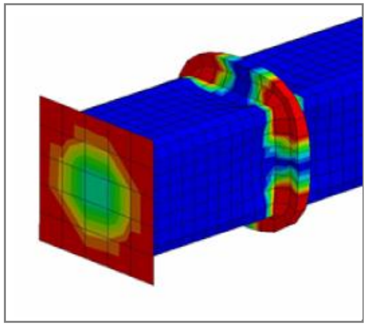


13. LS-DYNA® Forum

6 – 8 Oktober 2014, Bamberg, Germany

Workshop

Kontakte in LS-DYNA®

N. Karajan, T. Graf, F. Andrade
{nik, tg, fia}@dynamore,de

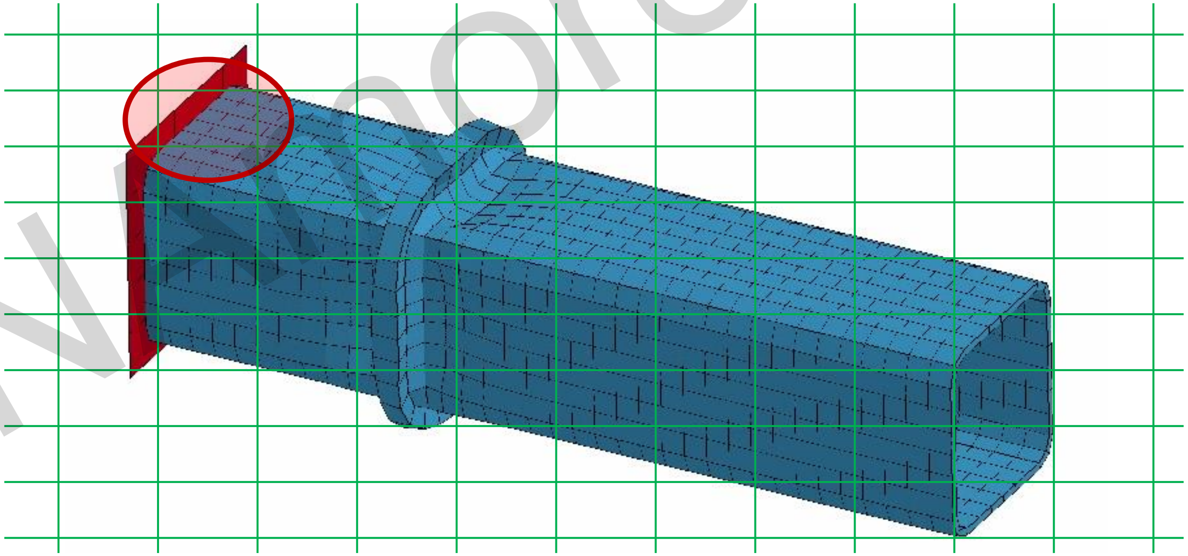
DYNAmore GmbH, Stuttgart (Germany)

Overview

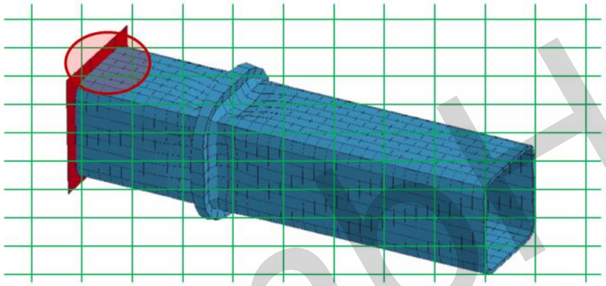
- Contact Search and Contact Treatment
- Defining Contacts in LS-DYNA
- Definition of Friction
- Contact Thickness
- Contact Stiffness
- Initial Penetrations
- Tied contacts
- Contact Output
- Summary
- Recommendations

Contact Search and Contact Treatment

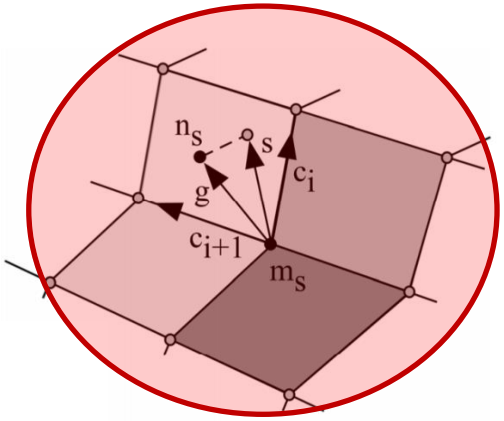
- Penetration of nodes into the opposite segment has to be checked
- Two steps for better performance during the search for contact partners:
  - Global search: Searching nodes and segments of possible contact candidates
  - Local search: Check for penetration between nodes and segments
- Applying contact condition:
  - Penalty method
  - Kinematic constraint method

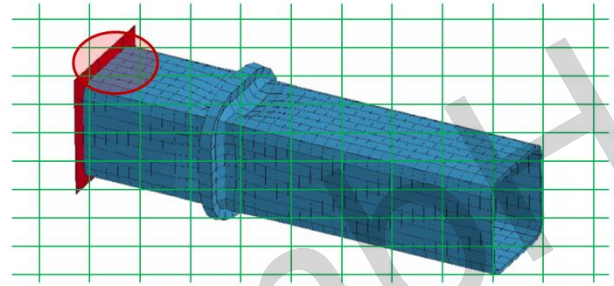


- Global and local contact search in more detail
  - Global search
    - Internal procedure
      1. Volume is partitioned into buckets
      2. Loop over all master segments:
         a. For each master segment, a list of crossed buckets is created
         b. Orthogonal distances of all nodes to the master segment are calculated
         c. Each node *k* stores the closest, second/third/… closest segment (for most contact types that‘s two segments per node)
      3. Each node stores the nearest nodes on these segments permanently



    - Local search with the segments and nodes found by the bucket sort
    - Execution of bucket sort every 10-200 cycles, (every cycle in case of implicit simulations)
    - Parameters of interest:
      SMP: BSORT, DEPTH (*CONTACT, optional card A)
      MPP: BCKT, NS2TRC (*CONTACT, MPP card)





- Local search:
  - Applied in each time step!
  - Accurate search for interpenetration only for the contact candidates found in global search
  - Bucket sort: Nearest master node(s) is/are found for slave node
  - Identifying master segments for slave node
  - Compute the orthogonal distance (normal projection) of the slave node to the master segments by solving the following system of equations

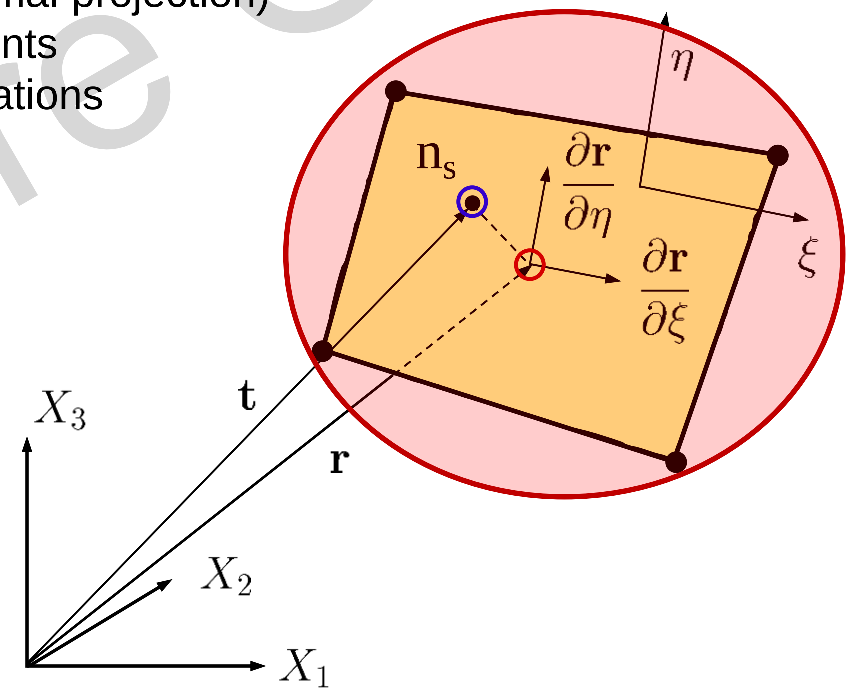
$$\frac{\partial \mathbf{r}(\xi_c,\, \eta_c)}{\partial \xi} \cdot [\mathbf{t} - \mathbf{r}(\xi_c,\, \eta_c)] \;=\; 0$$

$$\frac{\partial \mathbf{r}(\xi_c,\, \eta_c)}{\partial \eta} \cdot [\mathbf{t} - \mathbf{r}(\xi_c,\, \eta_c)] \;=\; 0$$

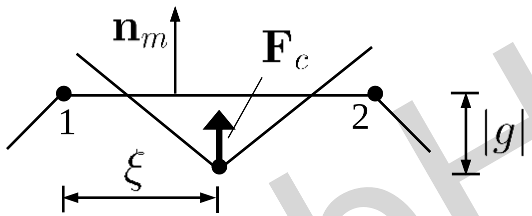
  - Then, check for penetrations

$$g = \mathbf{n}_m \cdot [\mathbf{t} - \mathbf{r}(\xi_c,\, \eta_c)]$$

$g \leq 0$ → penetration



- Penalty method to treat the contact
  - Contact treatment is internally represented by linear springs between the slave nodes and the nearest master segment
  - Resulting forces proportional to the penetration are applied to resist, and ultimately eliminate, the penetration
  - Contact stiffness calculation
    - Penalty-based approach
      - » Segments on solid/shell elements:
      - » K : slave/master bulk modulus
      - » Default: Min. value of slave/master is used



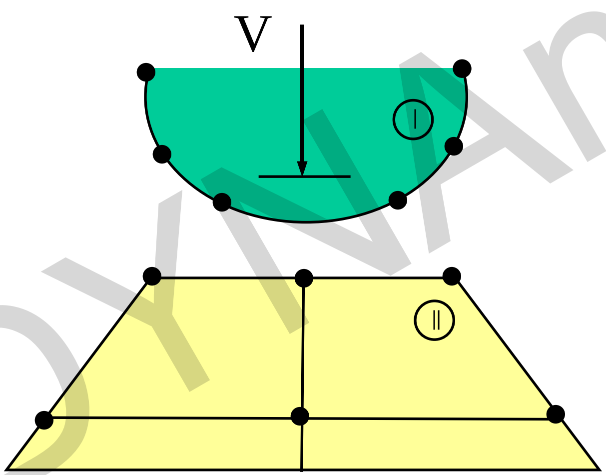
$$\mathbf{F}_c = k\, g\, \mathbf{n}_m$$

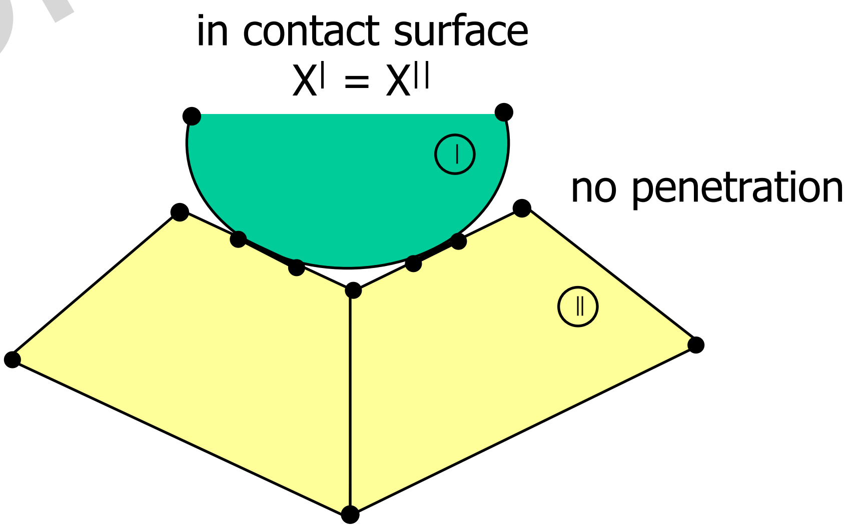
$$\mathbf{F}_{1c} = (1-\xi)\, k\, \mathbf{g}, \quad \mathbf{F}_{2c} = \xi\, k\, \mathbf{g}$$

$$k = \frac{f_S\, A^2\, K}{V} \quad \text{and} \quad k = \frac{f_S\, A\, K}{d_{\min}}$$

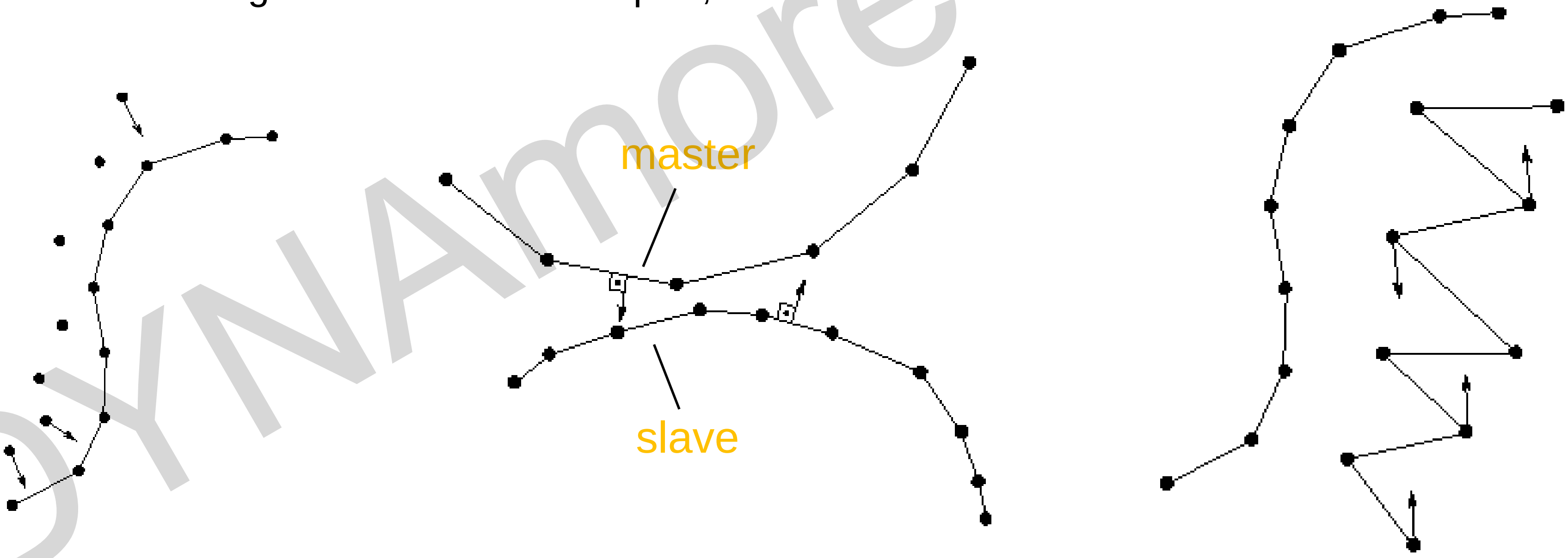
- **Contact stiffness affects critical time step of simulation**
  - » Critical time steps of contact springs are generally not taken into account (it is not certain that the contact springs are really activated)
  - » Estimation of contact time step is printed on the d3hsp: "The LS-DYNA time step should not exceed …"
  - » If necessary, scale time step using TSSFAC or the contact stiffness
  - » Contact time step can be considered using eroding contact types, ECDT (*CONTROL_CONTACT)

- Kinematic constraint method to treat the contact
  - Impact and release conditions of Hughes et al. [1976]
  - Constraints are imposed on the global equations
    - Transformation of the nodal displacement components of the slave nodes along the contact interface
    - Eliminating the normal degree of freedom of the slave nodes
    - Interpenetrating nodes are moved back to surface
  - **Problems:**
    - Rigid bodies cannot be handled correctly (multiple constraints)
    - Either energy or momentum is preserved, never both

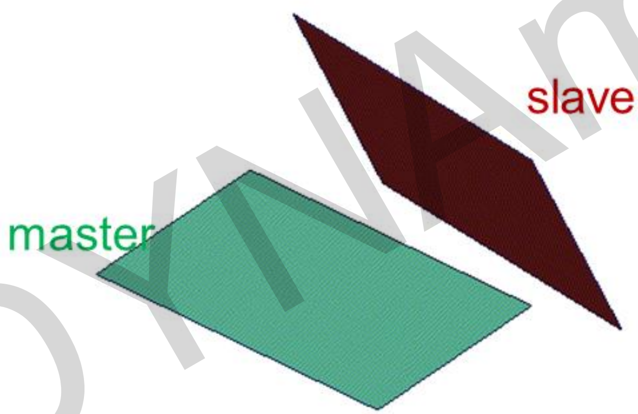




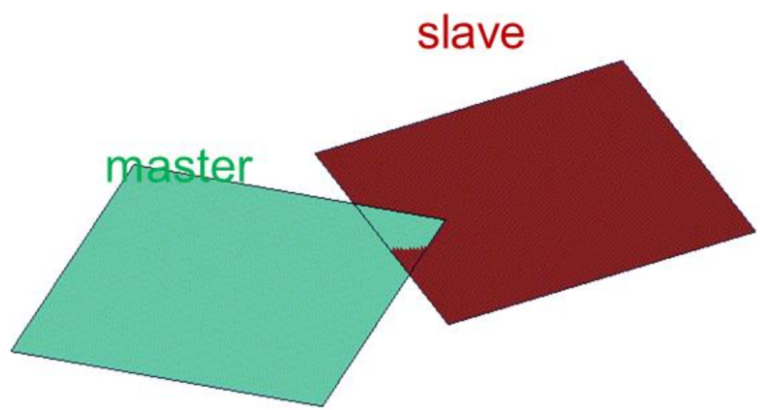
- Different treatment of sliding interface contact
  - One-way treatment
    - Slave nodes are checked for penetration through master segments
    - Slave side needs to be the finer mesh
  - Two-way treatment
    - Slave nodes are checked for penetration through master segments and master nodes are checked for penetration through slave segments
  - Single surface contact
    - Contact is considered between all parts in the slave list including self-contact of each part, no master surface is defined



- One-way contact types
  - Only the user-specified slave nodes are checked for penetration through master segments
  - Applications:
    - Appropriate when master side is rigid, e.g., a punch in metal stamping
    - Appropriate for deformable bodies when a relative fine mesh (slave) encounters a relatively smooth, coarse mesh (master)
  - Examples:
    - *CONTACT_AUTOMATIC_NODES_TO_SURFACE (type a5)
    - *CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE (type a10)

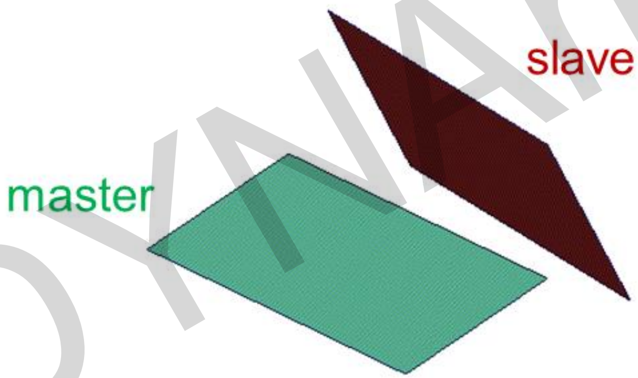


contact recognized

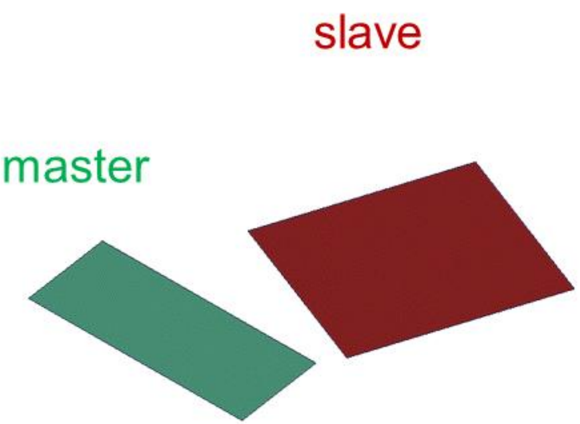


contact not recognized

- Two-way contact types
  - Slave nodes are checked for penetration through master segments and master nodes are checked for penetration through slave segments
  - i.e., the treatment is symmetric and the definition of the slave and master surface is arbitrary
  - Increased cost of approximately a factor two
  - Examples:
    - – *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE (type a3)
    - – *CONTACT_FORMING_SURFACE_TO_SURFACE (type m3)

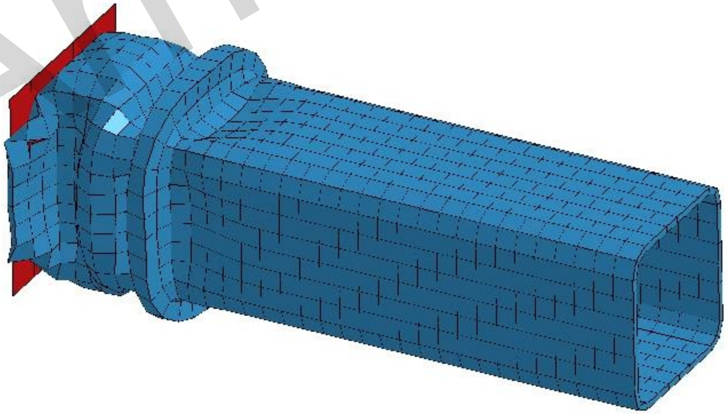


contact recognized



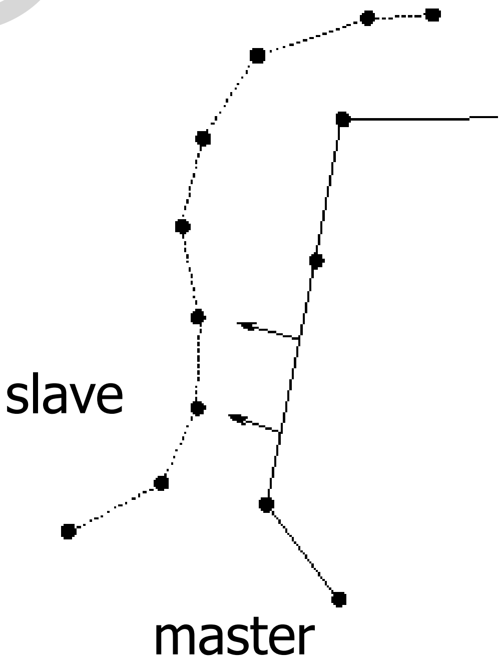
contact recognized

- Single surface contact types
  - Contact is considered between all parts in the slave list including self-contact of each part and no master surface is defined
  - Very reliable and accurate contact type, if the model is accurately defined
  - If several interpenetrations exist in the initial conditions, energy balances show either a growth or decay of energy as calculation proceeds
  - Examples:
    - *CONTACT_AUTOMATIC_SINGLE_SURFACE (type 13)
    - *CONTACT_AUTOMATIC_GENERAL_{INTERIOR} (type 26)
      → shell edge-to-edge and beam-to-beam contact is treated automatically



- **Defining Contacts in LS-DYNA**
  - Card ordering for *CONTACT_OPTION….
    - Card for ID Option → CID, heading
    - **Card 1 (mandatory)** → slave/master, box, print flags
    - **Card 2 (mandatory)** → friction, viscous damping, birth- death-time
    - **Card 3 (mandatory)** → penalty stiffness, optional thickness, friction scaling
    - Card 4 only mandatory for the following contact types:

      *CONTACT_CONSTRAINED_type
      *CONTACT_DRAWBEAD
      *CONTACT_ERODING_type
      *CONTACT_..._INTERFERENCE
      *CONTACT_RIGID_type
      *CONTACT_TIEBREAK_type
    - Card for THERMAL option
    - Optional card A; soft constraint, MAXPAR, ...
    - Optional card B; PENMAX, optional solid thickness, ...
    - Optional card C; IGNORE flag



- Recommended contact for most analyses: *CONTACT_AUTOMATIC_SINGLE_SURFACE

```
*CONTACT_AUTOMATIC_SINGLE_SURFACE_TITLE
$     CID|                                                                    NAME|
        1 Global Contact - all parts against all parts
$    SSID|     MSID|    SSTYP|    MSTYP|   SBOXID|   MBOXID|       SPR|       MPR|

$      FS|       FD|       DC|       VC|      VDC|   PENCHK|        BT|        DT|
      0.2       0.2       1.0                 20.0
$     SFS|      SFM|      SST|      MST|     SFST|     SFMT|       FSF|       VSF|
                          0.0       0.0
$    SOFT|   SOFSCL|   LCIDAB|   MAXPAR|   PENTOL|    DEPTH|     BSORT|    FRCFRQ|
        1
$  PENMAX|   THKOPT|   SHLTHK|    SNLOG|     ISYM|    I2D3D|    SLDTHK|    SLDSTF|

$    IGAP|   IGNORE|   DPRFAC|   DTSTIF|                        FLANGL|   CID_RCF|
                  1
$   Q2TRI|   DTPCHK|    SFNBR|   FNLSCL|   DNLSCL|     TCSO|    TIEDID|    SHLEDG|

$  SHAREC|   CPARM8|   IPBACK|    SRNDE|
```

Cards 1 – 3 are mandatory for each *CONTACT Definition

Card 4 is missing for this contact type!

Optional Cards A - E

- Keyword input format: *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE

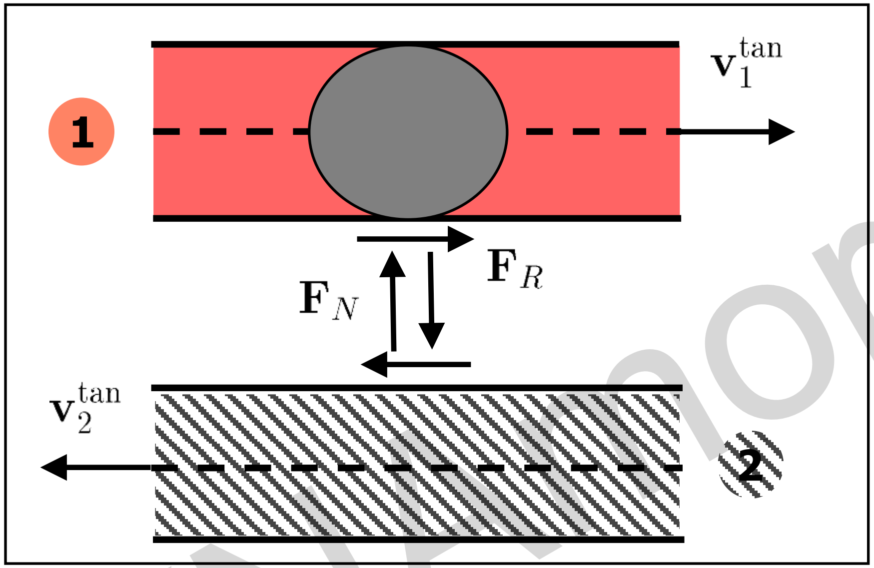
```
*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE
$     SSID|     MSID|    SSTYP|    MSTYP|   SBOXID|   MBOXID|      SPR|      MPR|
         2         3         3         2
$       FS|       FD|       DC|       VC|      VDC|   PENCHK|       BT|       DT|
       0.2       0.1         1                20.0
$      SFS|      SFM|      SST|      MST|     SFST|     SFMT|      FSF|      VSF|
```

```
*PART
Definition of a Part
$      PID|      SID|      MID|    EOSID|      HGID|
         2         1         2                   1
```

```
*SET_PART_LIST_TITLE
$                                                                         TITLE|
SpotShell elements
$   Set_ID|
         3
$    Part1|    Part2|    Part3|
         1         3         4
```

Definition of Friction

- Node of element 1 is checked against segment of element 2
- Mode of operation



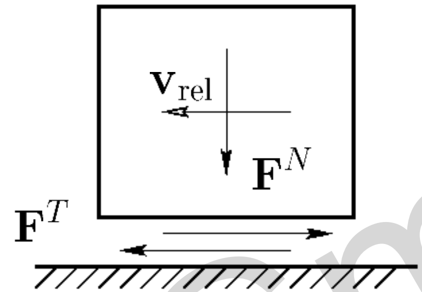
$$\mathbf{v}_{\text{rel}} = \mathbf{v}_1^{\tan} - \mathbf{v}_2^{\tan}$$

$$\mathbf{F}_N \leq 0$$ (no tensile traction)

$\mathbf{F}_N$:	Normal contact force
$\mathbf{F}_R$:	Frictional force
$\mathbf{v}_{\text{rel}}$:	Relative velocity between the two parts (fric. force)
$\mathbf{v}^{\tan}$:	Tangential velocity

- Compression loads are transferred between the slave nodes and the master segments
- Tangential loads are transferred when contact friction is active

- Friction
  - rigid walls
    - frictionless sliding after contact
    - no sliding after contact
    - *Coulomb* friction



  - other contact surfaces
    - *Coulomb* friction $\boxed{\mathbf{F}^T \leq \mu \, \mathbf{F}^N}$ where $\mu = \mu_d + (\mu_s - \mu_d)\exp(-d_c \, |\mathbf{v}_{\mathrm{rel}}|)$

with: μ_s - static coefficient of friction

μ_d - dynamic coefficient of friction

d_c - exponential decay factor (keep units in mind)

    - viscous friction to limit friction force (yielding of materials) $\boxed{\mathbf{F}_{\mathrm{lim}} = \mu_v \, \mathbf{A}_c}$

with: μ_v - coefficient for viscous friction

$\mathbf{A}_c$ - area of the segment contacted

- *Coulomb* friction

$$\boxed{\mathbf{F}^T \leq \mu\, \mathbf{F}^N} \qquad \text{where} \qquad \mu = \mu_d + (\mu_s - \mu_d)\exp(-d_c\,|\mathbf{v}_{\text{rel}}|)$$

```
*CONTACT_AUTOMATIC_SINGLE_SURFACE_TITLE
$     CID|                                                                       NAME|
         1Global Contact
$    SSID|     MSID|    SSTYP|    MSTYP|   SBOXID|   MBOXID|      SPR|      MPR|
         5        0        6        0        0        0        0        0
$      FS|       FD|       DC|       VC|      VDC|   PENCHK|       BT|       DT|
       0.2      0.0        0               20.0
$     SFS|      SFM|      SST|      MST|     SFST|     SFMT|      FSF|      VSF|
                         0.0      0.0
```

- FS: Static coefficient of friction
- FD: Dynamic coefficient of friction
- DC: Exponential decay coefficient
- VC: Coefficient for viscous friction
- FSF: Coulomb friction scale factor $\rightarrow \quad \mu_{SC} = \text{FSF}\,\mu_s$
- VSF: Viscous friction scale factor $\rightarrow \quad \mathbf{F}_{\text{lim}} = \text{VC}\;\;\text{VSF}\,\mathbf{A}_c$

□ *PART_CONTACT (*CONTACT-Keyword: FS=-1)

```
*PART_CONTACT
 Steel_Part
$      pid|     secid|       mid|     eosid|      hgid|      grav|    adpopt|      tmid|
        18          3        140          0          4
$      fs|        fd|        dc|        vc|      optt|       sft|       ssf|
       0.2       0.1       1.0       0.0        4.9       0.0       0.0
```

– available for the following contact formulations:

» *CONTACT_AUTOMATIC_NODES_TO_SURFACE

» *CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE

» *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE

» *CONTACT_AUTOMATIC_SINGLE_SURFACE

» *CONTACT_AIRBAG_SINGLE_SURFACE

» *CONTACT_AUTOMATIC_GENERAL

» *CONTACT_SINGLE_SURFACE

» *CONTACT_ERODING_SINGLE_SURFACE

□ *DEFINE_FRICTION (*CONTACT-Keyword: FS=-2)

```
*DEFINE_FRICTION
$       ID|     FS_D|     FD_D|     DC_D|     VC_D|
         1         0         0         0         0
$    PID_I|    PID_J|    FS_IJ|    FD_IJ|    DC_IJ|    VC_IJ|   PTYPEI|   PTYPEJ|
         1         2       0.2       0.1         1                 PSET
$    PID_I|    PID_J|    FS_IJ|    FD_IJ|    DC_IJ|    VC_IJ|
         3         4       0.2       0.1         1
...next "*" - Card terminates the friction definition
```

default friction values

friction values for a single pair of PIDs or PSETs

– if more than one DEFINE_FRICTION-card is defined, FD (*CONTACT_) references ID

– available for the following contact formulations:

» *CONTACT_AUTOMATIC_NODES_TO_SURFACE

» *CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE

» *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE

» *CONTACT_AUTOMATIC_SINGLE_SURFACE

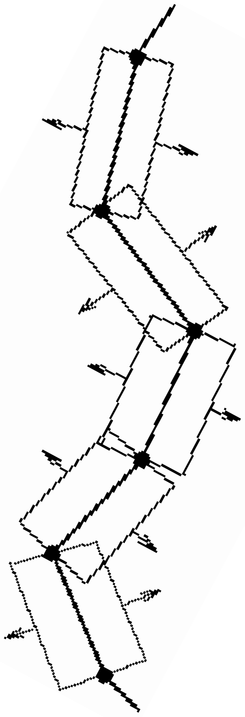
» *CONTACT_AUTOMATIC_GENERAL

» *CONTACT_SINGLE_SURFACE

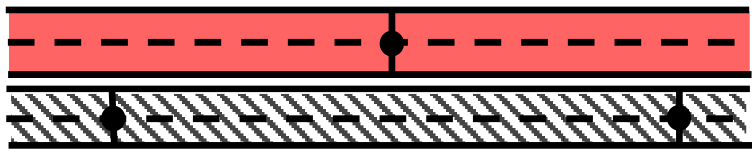
» *CONTACT_ERODING_SINGLE_SURFACE

• Automatic Contacts and Contact Thickness

- *CONTACT_AUTOMATIC_...
  - Often large deformations with changing contact situations
  - Predetermination of contact is difficult to impossible
  - Non-oriented AUTOMATIC contact types are recommended
    with penetration detection coming from either side of the shell elements
  - To avoid instabilities slave nodes that penetrate "too far" are eliminated/released
    - e.g. *CONTACT_AUTOMATIC_SINGLE_SURFACE with shell to shell contact
      d = 0.4 x (master thickness + slave thickness)
    - Very high forces due to large penetrations are not applied
    - Release criteria might be controlled by PENMAX and XPENE
  - Thickness offset: Segment based projection
  - **Advantages**:
    - Simultaneous contact on both sides of shell surfaces is possible
    - Segment orientation is meaningless
    - Easy to use
  - **Disadvantages:**
    - If shell thickness to small, nodes may penetrate and be released
      → contact thickness must be re-defined



- Contact thickness



- Contact thickness is considered and generally taken as the shell thickness using
  - Single surface
  - Constraint method and
  - Automatic node-to-surface or surface-to-surface contact types
- Shell thickness is considered for non-automatic node/surface to surface contact types if SHLTHK=1 or 2 (*CONTROL_CONTACT)
- Contact thickness of shell elements can be modified using
  - SST and MST (Card 3 in contact definition)
    → absolute value; overrides true thickness; definition holds for whole contact
  - SFST and SMST (Card 3 in contact definition)
    → scaling factor; scales true thickness; definition holds for whole contact
  - OPTT (*PART_CONTACT)
    absolute value; overrides thickness modifications in contact definition; definition holds for every individual part; applies to shell and beams

- *CONTACT_... keyword card

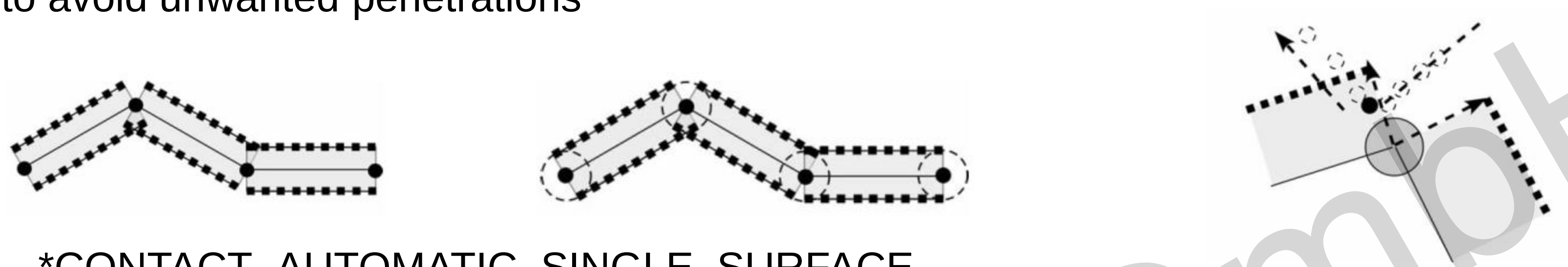
```
*CONTACT_AUTOMATIC_SINGLE_SURFACE
$     SSID|      MSID|     SSTYP|     MSTYP|    SBOXID|    MBOXID|       SPR|       MPR|

$       FS|        FD|        DC|        VC|       VDC|    PENCHK|        BT|        DT|
       0.2       0.1       1.0                 20.0
$      SFS|       SFM|       SST|       MST|      SFST|      SFMT|       FSF|       VSF|
                             4.2       4.2
```

- *PART_CONT

```
*PART_CONTACT
 Steel_Part
$      pid|     secid|       mid|     eosid|      hgid|      grav|    adpopt|      tmid|
        18         3       140                     4
$       fs|        fd|        dc|        vc|      optt|       sft|       ssf|
       0.2       0.1       1.0                   4.2
```

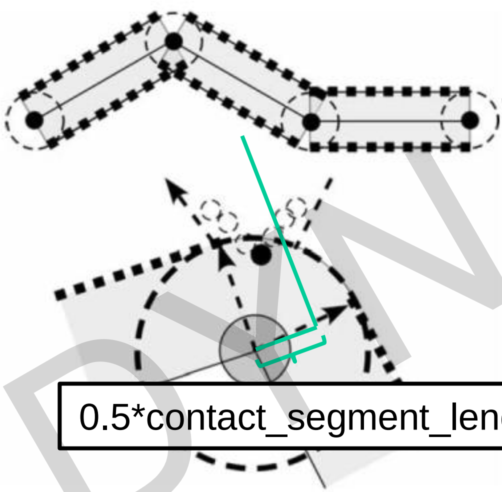
- SINGLE_SURFACE-contacts:
  - Default contact thickness is a function of SSTHK (*CONTROL_CONTACT)
  - SSTHK=0: $\mathrm{SST} = \min(t,\, 0.4\, l_{min}^{edge})$
  - SSTHK=1: $\mathrm{SST} = t$
- *CONTACT_AIRBAG_SINGLE_SURFACE: Contact thickness vs. time can be specified using LCIDAB (*CONTACT, optional card A)

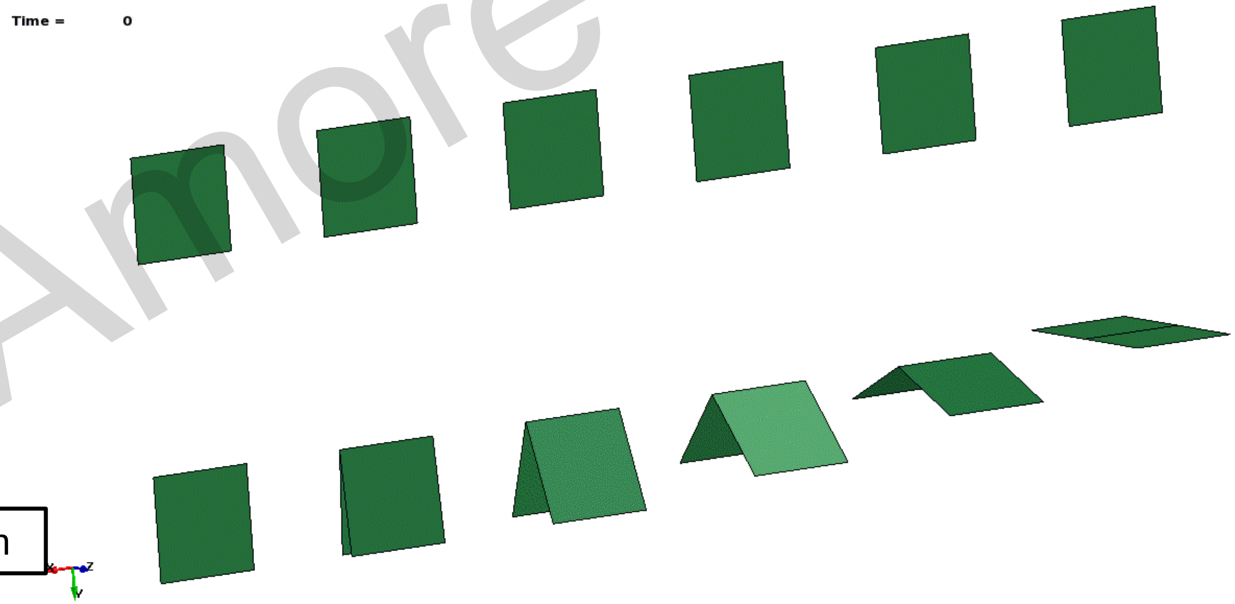
- A cylindrical surface is created in the gap between two shell segments to avoid unwanted penetrations



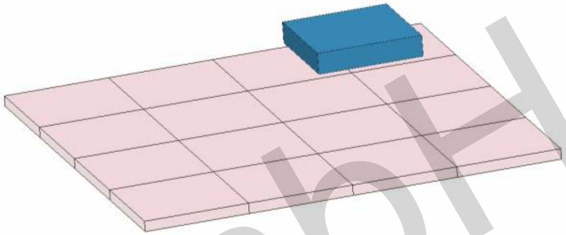
*CONTACT_AUTOMATIC_SINGLE_SURFACE

- ◻ Slave node is only considered in the contact algorithm, if normal projection on contact segment lies within the contact segment or within a surrounding area with a width of 0.5*contact_segment_length





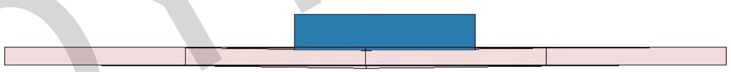
- Numerical example “Shell rebounds from plate”
  - Investigate influence of contact thickness



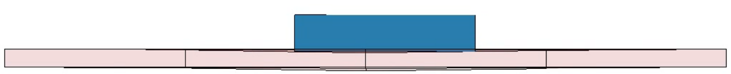
***CONTACT_SURFACE_TO_SURFACE**
***CONTROL_CONTACT:** ssthk=0

***CONTACT_AUTOMATIC_SURFACE_TO_SURFACE**

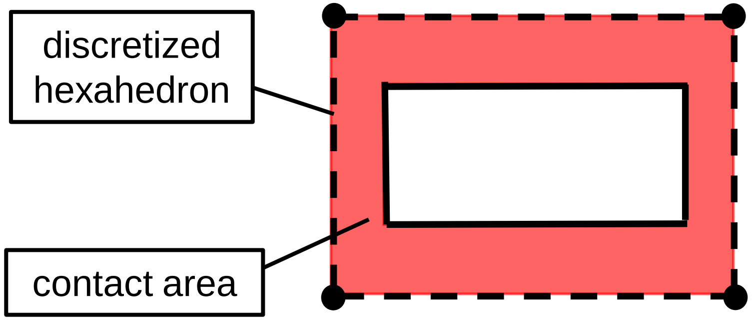
***CONTACT_SURFACE_TO_SURFACE**
***CONTROL_CONTACT:** ssthk=1

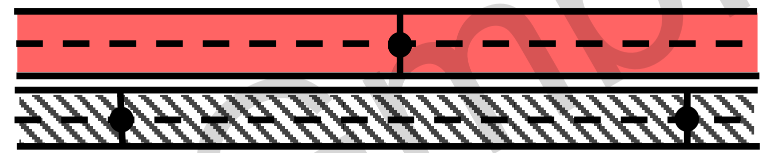


***CONTACT_SINGLE_SURFACE**
***CONTACT_AUTOMATIC_SINGLE_SURFACE**



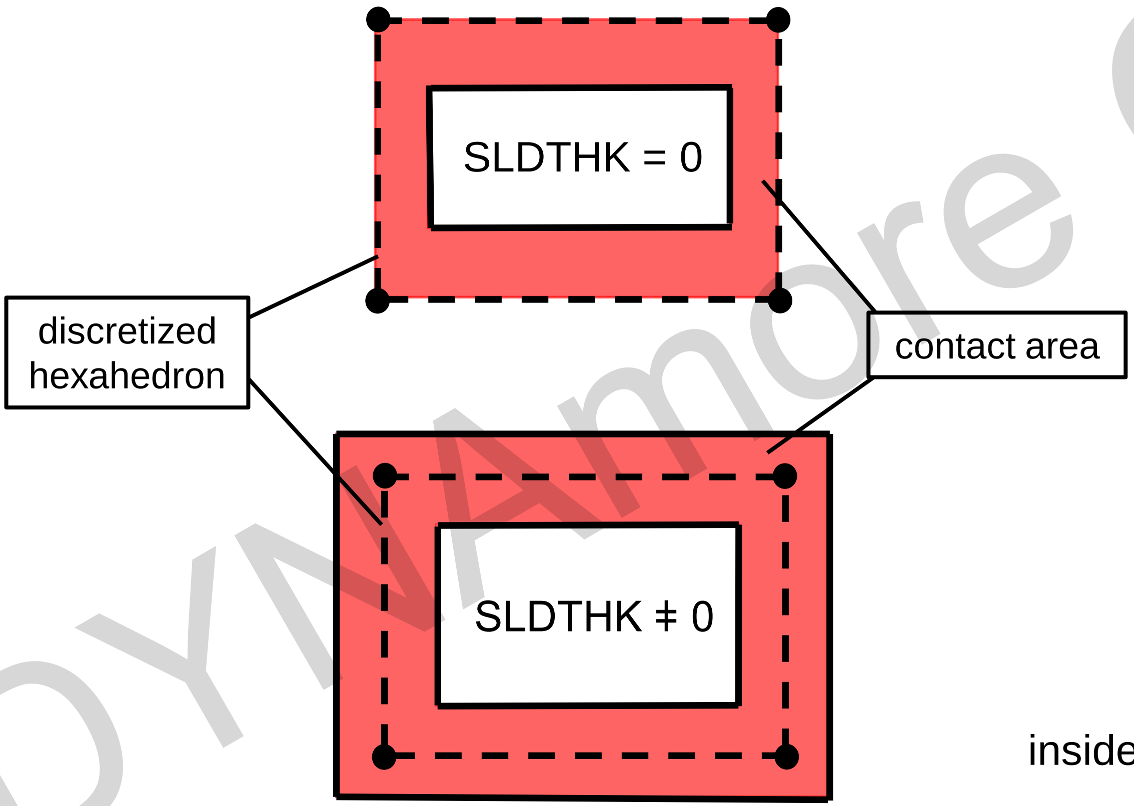
- Contact behavior of volume elements
  – By default, contact area “lies within” the volume element

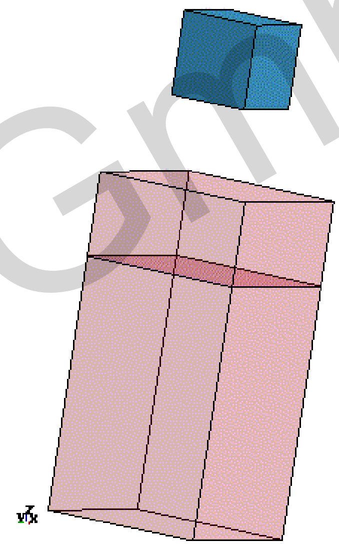




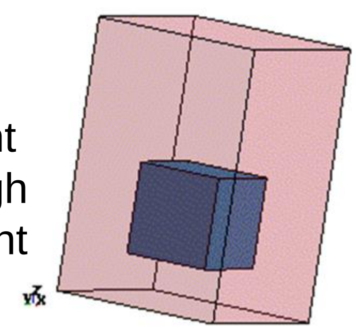
  – The usage of contact types with an automatic release of too far penetrated nodes, e.g., *CONTACT_AUTOMATIC_SINGLE_SURFACE, SOFT={0|1}, can result in poor contact behavior:
    » The computed contact thickness of volume elements (based on volume and area) is generally lower than the contact thickness of shell elements, e.g., *CONTACT_ASS: contact thickness = 0.4 x volume/area
    » The maximum penetration is based on the contact thickness, e.g., *CONTACT_ASS: d = 0.5 x contact thickness
    » Thus, the penetrating nodes are released much earlier than in the case of contact problems with shell elements

- Using SLDTHK (optional card B), “virtual dummy shells” may be automatically generated in order to cover the surface of the solid part
- Optional solid element stiffness (only for contact treatment) can be modified using SLDSTF (optional card B)





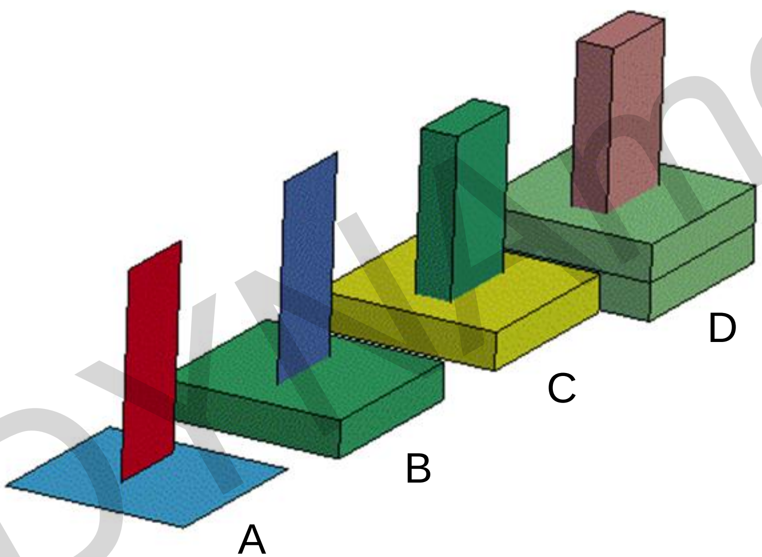
Experiment: top element is deleted after contact



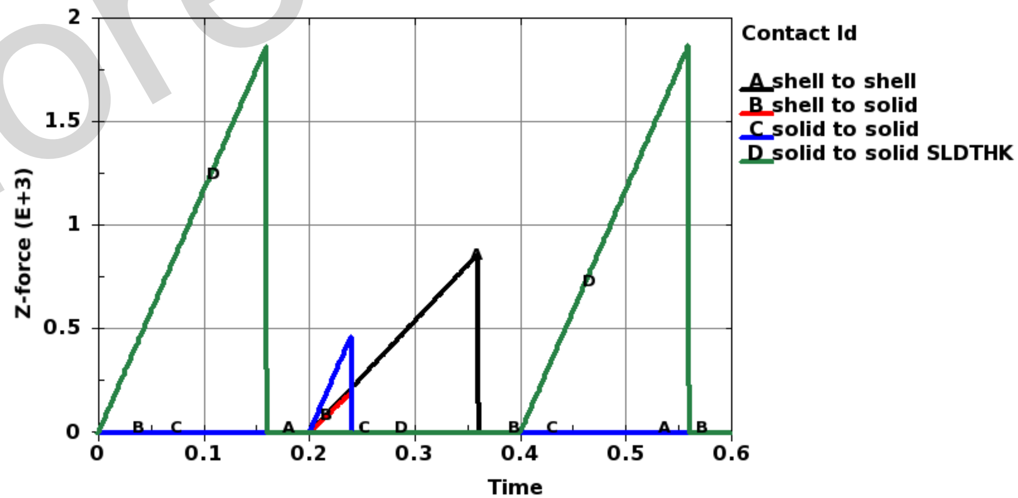
blue element can fall through inside of red element

– Numerical example – displacement driven, *CONTACT_AUTOMATIC_SINGLE_SURFACE
  - » Shell to shell is reference (curve A – black)
  - » Nodes are released much earlier in standard shell to solid and solid to solid contact treatment (curve B – red, curve C – blue)
  - » Contact thickness is equal to shell to shell contact using SLDTHK
  - » Two contact problems arise using SLDTHK (top and bottom of solid part)
  - » Global stiffness depends on the number of nodes in contact (curve C and D)

displacement-driven experiment



resulting penalty force of the contact



- **Contact Stiffness**
  - Penalty method – calculation of contact spring stiffness
    - SOFT=0

Segments on solids: $$k_{\{S/M\}} = \mathrm{SLSFAC}\,\{\mathrm{SFS}/\mathrm{SFM}\}\,K_{\{S/M\}}\frac{A^2_{\{S/M\}}}{V_{\{S/M\}}}$$

Shell elements: $$k_{\{S/M\}} = \mathrm{SLSFAC}\,\{\mathrm{SFS}/\mathrm{SFM}\}\,K_{\{S/M\}}\frac{A_{\{S/M\}}}{D^{\max}_{\{S/M\}}}$$

K:	Bulk modulus of slave/master
SLSFAC:	Penalty scale factor (*CONTROL_CONTACT) (DEFAULT : 0.1)
SFS/SFM:	Scale factor on slave/master (DEFAULT : 1.0)
A, V:	Area, volume

– By default, the stiffness of the contact springs is given via $k = \min\{k_S,\, k_M\}$
– Other possibilities can be defined using PENOPT (*CONTROL_CONTACT)
– If the stiffness of the materials is dissimilar, SOFT=0 is not recommended

- Penalty method – calculation of contact spring stiffness
  - SOFT=1 - To consider contact between parts with different material stiffness
    - Automatic optimization of the each single contact spring stiffness
    - Calculation of contact spring stiffness is based on the *Courant-Friedrichs-Lewy*-criterion of a discrete spring element:

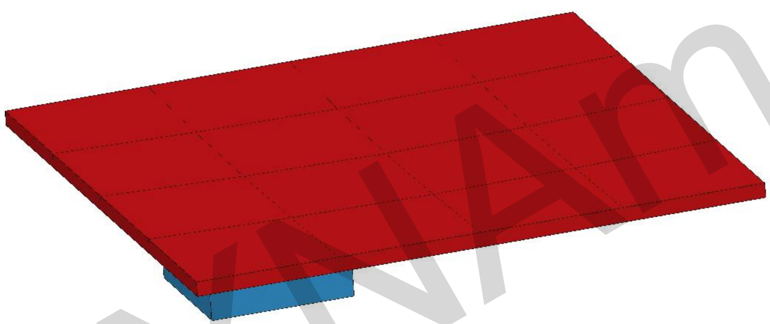
$$\Delta t_{\text{crit}} = 2\sqrt{\frac{m_1 m_2}{(m_1 + m_2)\,k}} = \sqrt{\frac{2M_1 M_2}{(M_1 + M_2)\,k}} \quad \rightarrow \quad k_{\{\text{SOFT}=1\}} = f\left(1/\Delta t^2,\, M_m,\, M_s\right)$$

$$\rightarrow \quad k = \max\{k_{\{\text{SOFT}=0\}},\, \text{SOFSCL} \cdot k_{\{\text{SOFT}=1\}}\}$$

    - Default: Scaling factor SOFSCL=0.1
    - SOFT=1 is recommended for impact analysis, where dissimilar materials come into contact
    - For the case of soft foam contacts metal, SOFT=1 gives interface stiffness that are one or two orders greater
    - Time step for calculation of *k* can be specified by the user: *CONTACT_, Optional C, DTSTIF
    - Occasionally, numerical instabilities in the contact behavior (d3hsp) occur → reduction of SOFSCL from the default value of 0.1 to 0.04-0.07 is recommended

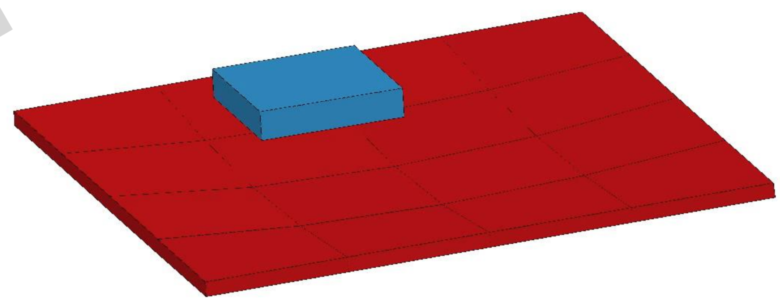
- Numerical example “Shell rebounds from plate”
  - Investigate influence of SOFT for parts with different stiffness
    - Red Material: $E_{\text{red}} = 210.0\,\text{GPa}$
    - Blue Material: $E_{\text{blue}} = 0.01 \cdot E_{\text{red}}$

```
*CONTACT_AUTOMATIC_SINGLE_SURFACE
SOFT = 0
```



contact fails (“does not hold”)

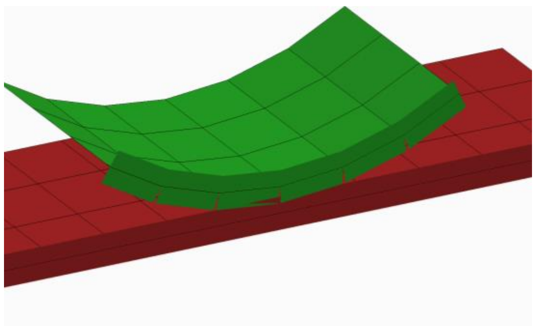
```
*CONTACT_AUTOMATIC_SINGLE_SURFACE
SOFT = 1
```



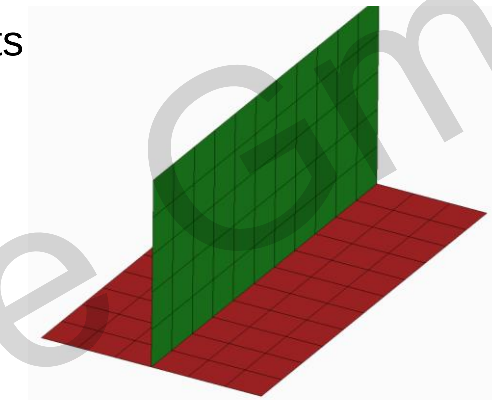
contact works fine

- **Initial Penetrations**
  - In general, initial penetration occurs if one or more nodes are within the contact range of their master segment during initialization of LS-DYNA

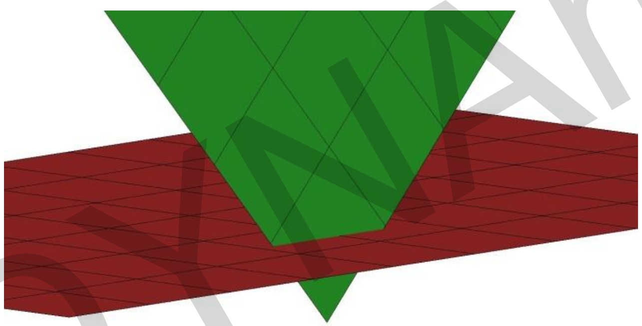
Elements too close to each other



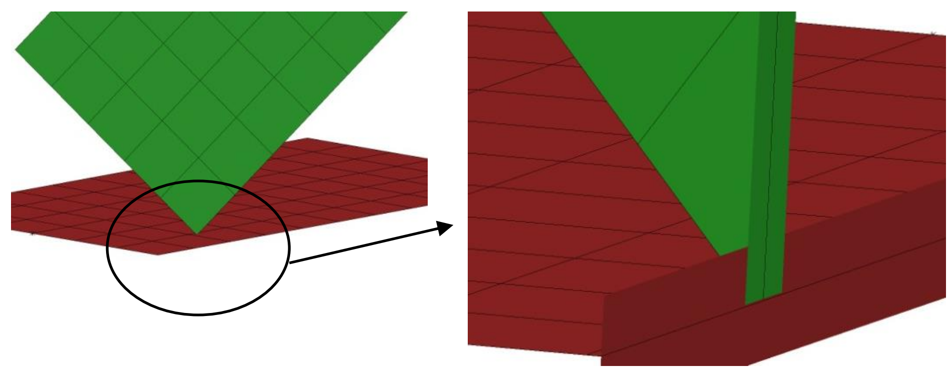
T-Joints



Stick-through problem



Mid-face penetrations



- finding initial penetrations with check-c (DYNA*more* tool → www.dynamore.de)

```
usage:  check-c [options] messag-file[s]

options:
-list        list all warned penetrating nodes
-top  ##     list and output top ## values  (default: ##=5)
-ptol ##     list and output only penetrations/separations > ##
-sid #,#,..  check only SIDs #,#,...
-bucket      show penetration buckets for each interface
-pid         print property for penetrating node
-psum        print property penetration summary
-rsort       sort for remaining distance
             (default: sort for penetrating distance)
-rtol ##     list and output only penetrations that have
             a remaining distance < ##
-title       print contact titles  if available  (read d3hsp-file)
-hsp <file>  d3hsp filename if other than "d3hsp" (needed for -title)
-typ         print contact type (eg: a3, 13, a13)
-typ -typ    print contact type in keyword (eg: SINGLE_SURFACE)
-timestep    print contact timestep if available
-all         list all contacts (even without any
             (usefull with -timestep and -typ)
-g <d3plot>  optional d3plot-file (needed for prop
             default: d3plot
```

Workshop on DYNA-Tools
Wednesday 8:20 AM

```
-ani         create Animator  session file for displaying nodes via "ide nod xxx"
-a4sel       create Animator4 session file for displaying nodes via "sel nod xxx"
             and create Animator4 groups named "id<sid>_<warn-type>"
-sel <typ>   selected warning type (default=all)
             values for <typ> in tied contacts:
             tied    - all warnings for tied contacts
             sep     - nodes with separation that are moved
             offset  - nodes with separation that are not moved for any reason
                       (OFFset remains but tied connection is applied - dangerous!)
             far     - nodes with separation that are too FAR away
                       (node is untied)
             short   - nodes with separation that would shorten connected beams
                       (node is untied)
             notfound- nodes that are not found to lie on any master segment
                       (node is untied)
             rigslav - nodes that are rigid slave nodes
                       (node is untied)
             rigmast - nodes that are found on master segment with any rigid node
                       (node is untied)
             conflict- nodes that conflict with other tied contact definition
                       (node is untied)
             untied  - all above warnings with untied nodes
                       (untied nodes of spotwelds should be corrected)

             values for <typ> in standard contacts (eg typ13):
             typ13   - all warnings for standard contacts that fo
             pen     - nodes penetrating segment
             mid     - nodes close to shell midplane
             del     - nodes deleted from contact aft
             below   - nodes too far below surface
             soft2   - element-element contact warning          contact
             inter   - element-element intersection war   ngs in soft2 contact (first state)
             last    - element-element intersection warnings in soft2 contact (last  state)
```

Workshop on DYNA-Tools
Wednesday 8:20 AM

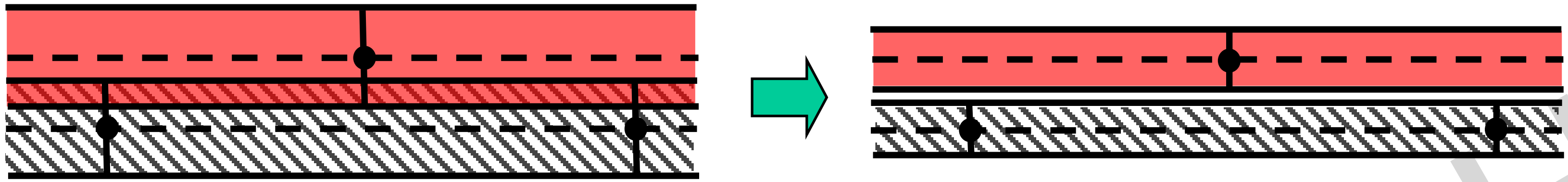
Initial Penetrations

```
-key        create LS-DYNA keyword file "check-c.key"
            with beams of length of the penetrations -
            can be loaded into LS-Post or Animator3/4
-medina     create Medina protocol files
-mnode      node id for new node creation in Medina protocol
            (default=9.999.999)
-version    print version
```

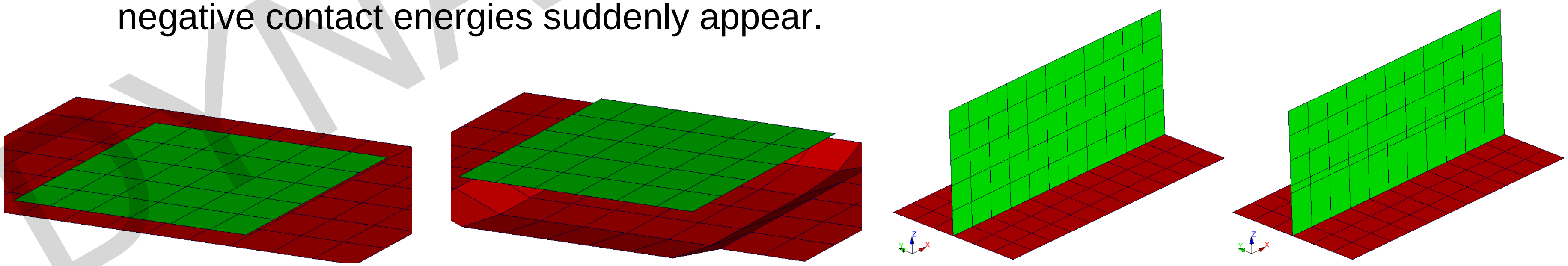
– example:

check-c -list -bucket -pid -title -typ -typ -timestep -key mes0000

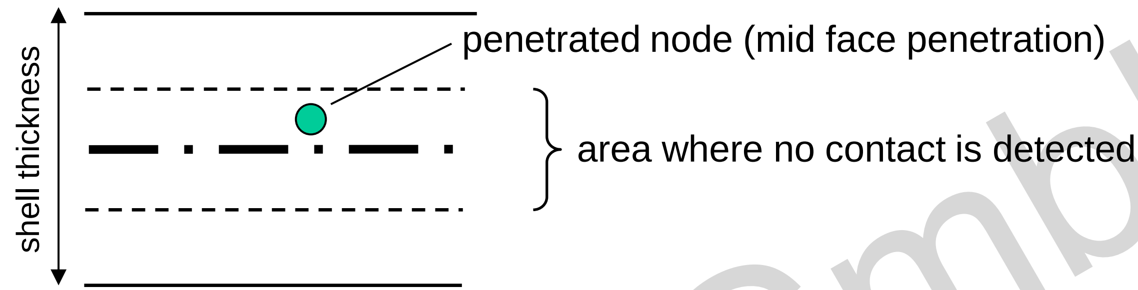
```
================================================================================
global contact
Interface ID:       1   AUTOMATIC_SURFACE_TO_SURFACE
    timestep: 1.589E-05
       range:    summ:    0.0-0.1: 0.1-0.2: 0.2-0.3: 0.3-0.4: 0.4-...:   max:
penetrations:       5         0        0        0        0        5     0.50000
   remaining:       5         0        0        0        0        5     2.00000
      node       penetration   remaining       pids
-------------------------------------------------------------
        14        0.50000000  2.00000000         1s
       101        0.50000000  2.00000000
       102        0.50000000  2.00000000
       103        0.50000000  2.00000000
       104        0.50000000  2.00000000
================================================================================
wrote check-c.key - LS-DYNA keyword (beams <=> penetrations)
```



- Contact surfaces are offset from shell mid-planes and from beam centerlines
  → extremely important to model appropriate gaps between shell and beam parts
- **Shooting node logic:** At any point during the simulation, if a node is suddenly found to be below the surface (node moves very fast and was not detected before penetration), LS-DYNA just moves the node to the master surface without applying any forces:
  - Advantages:
    - » eliminates manual removal process, thereby, saving user’s time and effort
  - Disadvantages:
    - » distorts original geometry at locations, where the penetrations are detected
    - » nodal coordinates after removal process could still penetrate other neighbouring segments and may lead to instability issues
- If shooting node logic is turned off (SNLOG=1), large forces and negative contact energies suddenly appear.

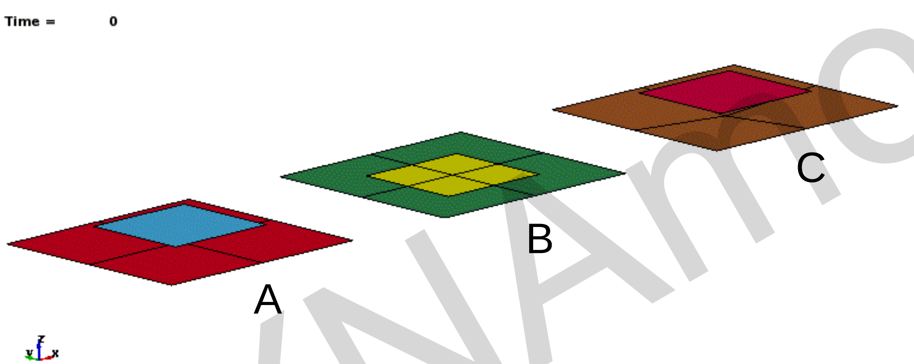


- Mid-face penetrations (very dangerous) – nodes are released in case of deep penetrations

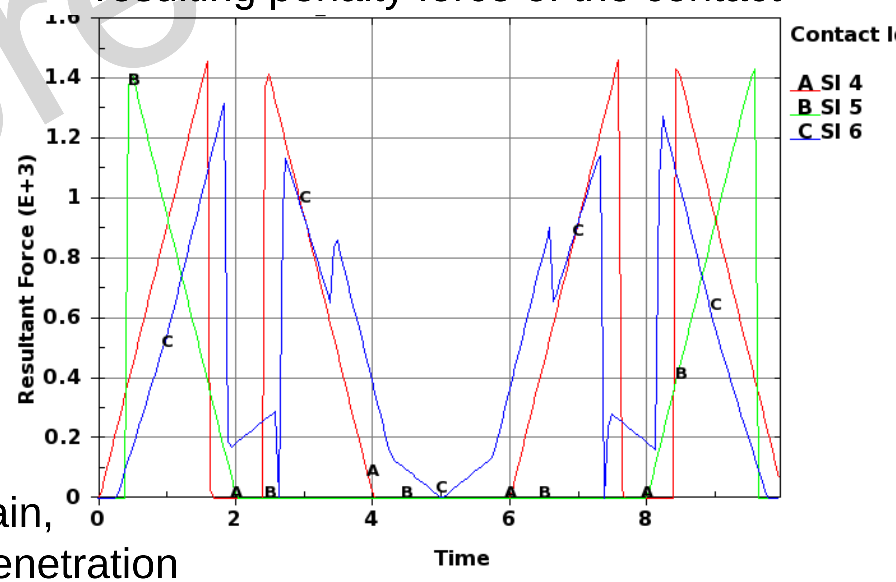


- Numerical example – *CONTACT_AUTOMATIC_SINGLE_SURFACE

displacement-driven experiment



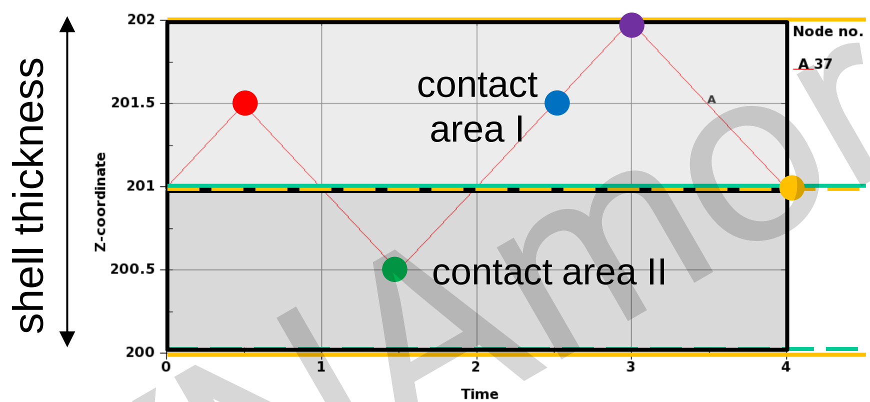
resulting penalty force of the contact



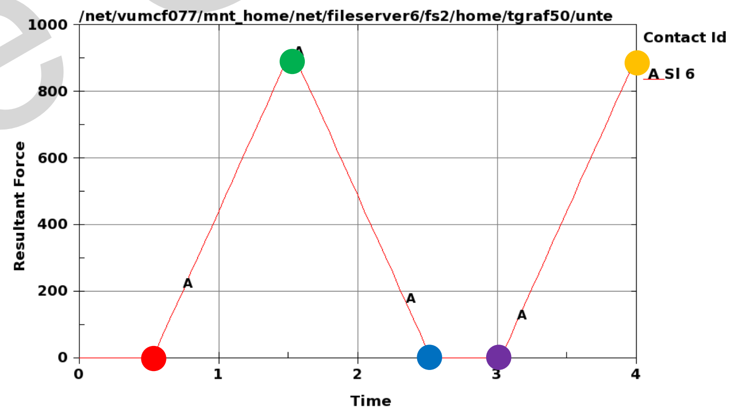
– Nodes are released for deep penetrations
– After the node is in the contact zone once again, a contact force is applied depending on the penetration

- IGNORE option
  - Ignores initial penetrations in the AUTOMATIC contact options.
    IGNORE=1 allow initial penetrations to exist by tracking the initial penetrations.
  - “Initial” in this context refers to the first time step at which a penetration is found
  - Shooting node logic has no effect
  - Contact forces will resist further penetration
  - Contact thickness is adjusted locally and is adjusted again,
    if penetration node *leaves* the contact region

z-coordinate of a node in the contact area



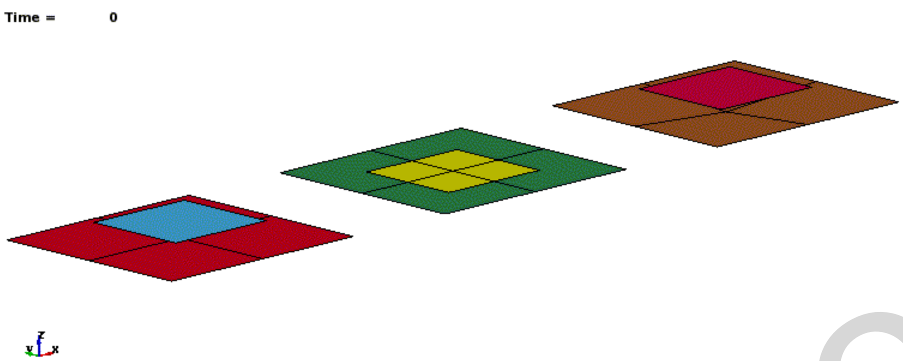
resulting contact force



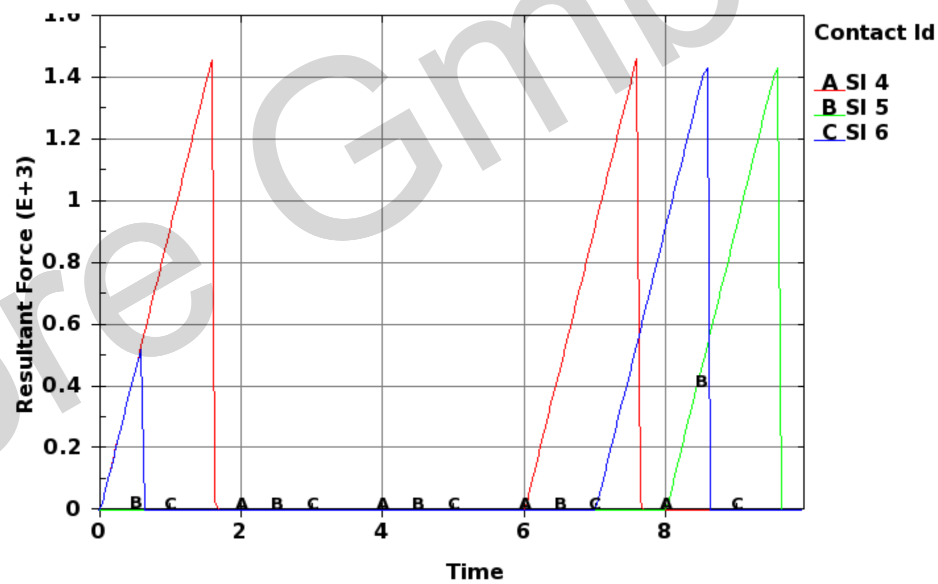
  - This option can be either specified globally in *CONTROL_CONTACT
    or for each interface in *CONTACT, Optional Card C
  - Using IGNORE=2, additional penetration warning messages are printed to
    the message-files with the original coordinates and the recommended coordinates
    of each slave node given.

□ Numerical example – *CONTACT_AUTOMATIC_SINGLE_SURFACE

displacement-driven experiment



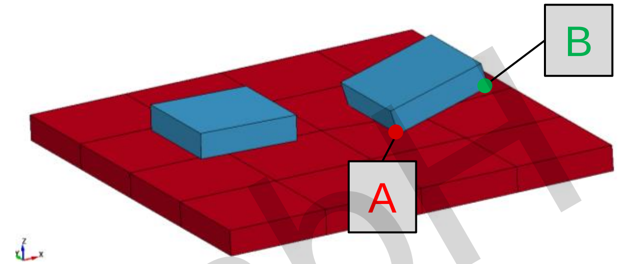
resulting penalty force of the contact



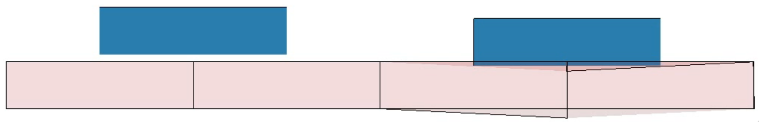
– Slave nodes are released for deep penetrations

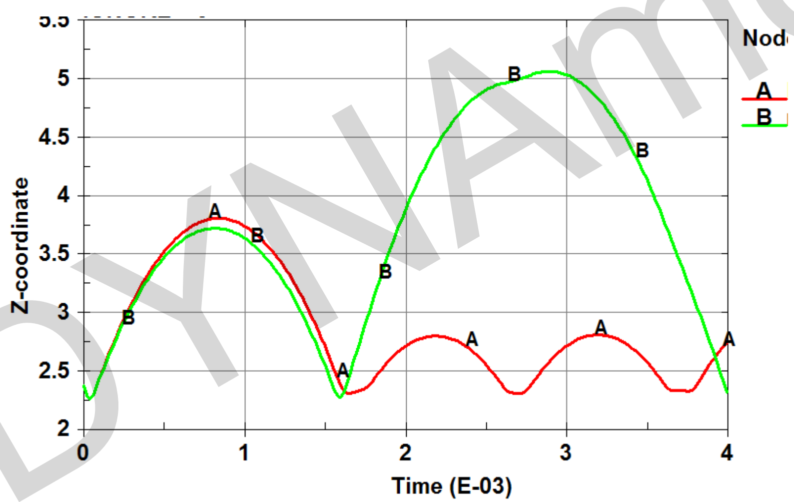
– A penalty force is not applied until they totally leave once the whole contact area and not only the "contact free zone" around the midplane

- Numerical example “Shell rebounds from plate”
  - Investigate influence of IGNORE on initial penetrations
    - – Blue plate has downward initial velocity
    - – Gravity pulls blue plate back down

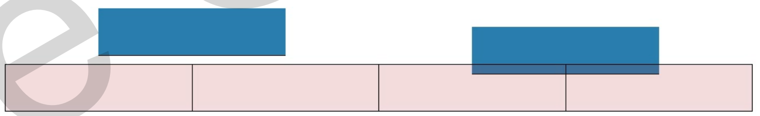


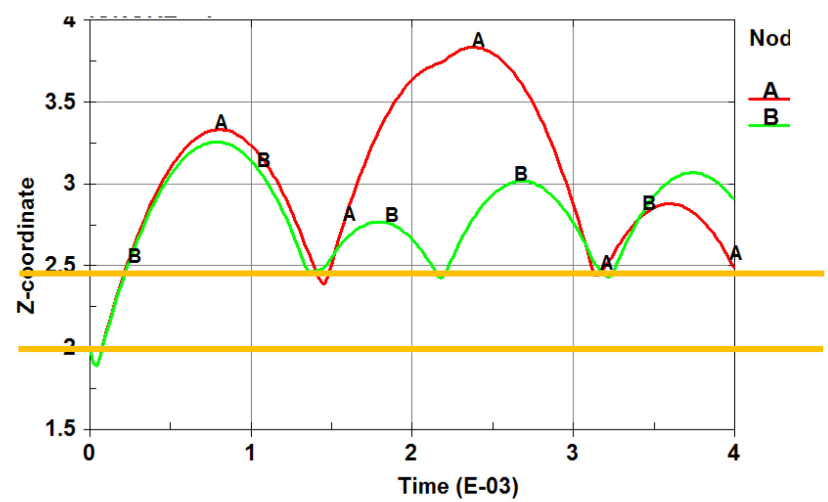
***CONTACT_AUTOMATIC_SINGLE_SURFACE**
IGNORE = 0



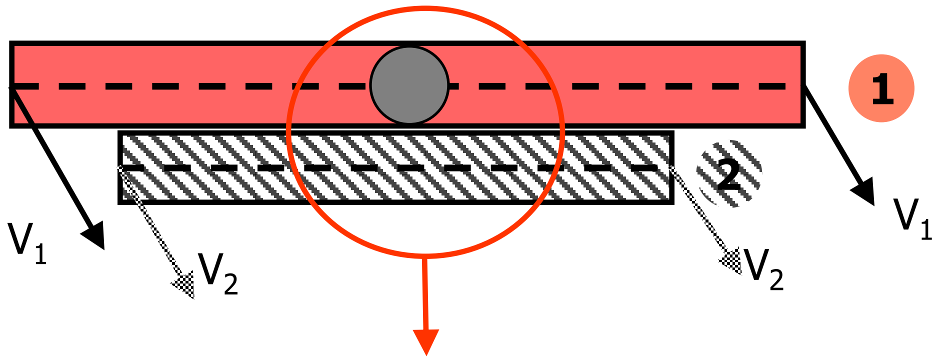


***CONTACT_AUTOMATIC_SINGLE_SURFACE**
IGNORE = 1





- **Tied contacts**
  - Mode of operation



node of part 1 (slave) is tied to segment of part 2 (master)

- □ "Normal" tied contact types
  - – Slave node is moved to the master segment
  - – The isoparametric position of the slave node with respect to its master segment is held fixed using **kinematic constraint equations**
  - – "Constrained_offset" tied contact types cannot be used with rigid bodies

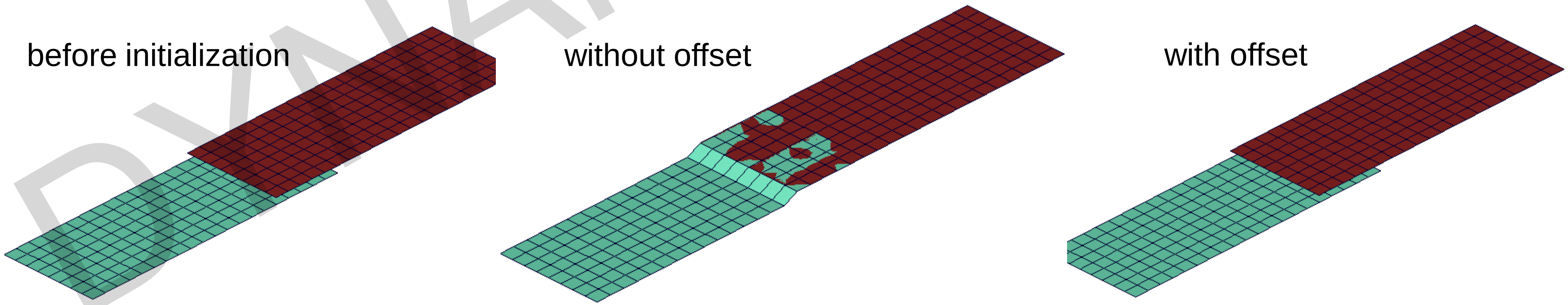
- □ "Offset" tied contact types
  - – Offset between master surface and slave node is permitted
  - – Offset tied contacts use a **penalty-based formulation** and can be used to tie rigid bodies
  - – Stiffness of tied contact springs is calculated similar to the penalty-based sliding interface contact types

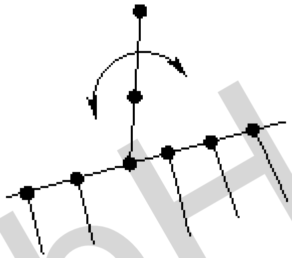
- Preliminary remarks
  - If a slave node is found close to the master segment (special criteria) the slave node is moved to the master segment
  - Initial geometry is slightly altered without invoking any stresses.
  - How close does it have to be:

$$\left.\begin{aligned} d_1 &= 0.6\,(t^s + t^m) \\ d_2 &= 0.05\,\min\,(\mathrm{diag}^m) \end{aligned}\right\} \rightarrow \left\{\begin{aligned} &\text{shell}: \; d = \max(d_1,\, d_2) \\ &\text{solid}: \; d = d_2 \end{aligned}\right.$$

  - If there is a large difference in element areas between the master and the slave side, the distance d_2 may be too large and may cause the unexpected projection of nodes that should not be tied.
  - To avoid this problem the slave and master thickness can be specified as negative values (SST, MST) in which case $d = \mathrm{abs}(d_1)$

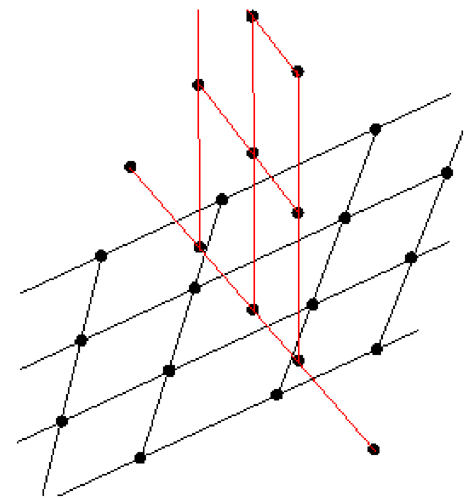
    Recommendation: Use only SST to define an additionally distance, otherwise oscillations can occur (b7 in comb. with volume elements).



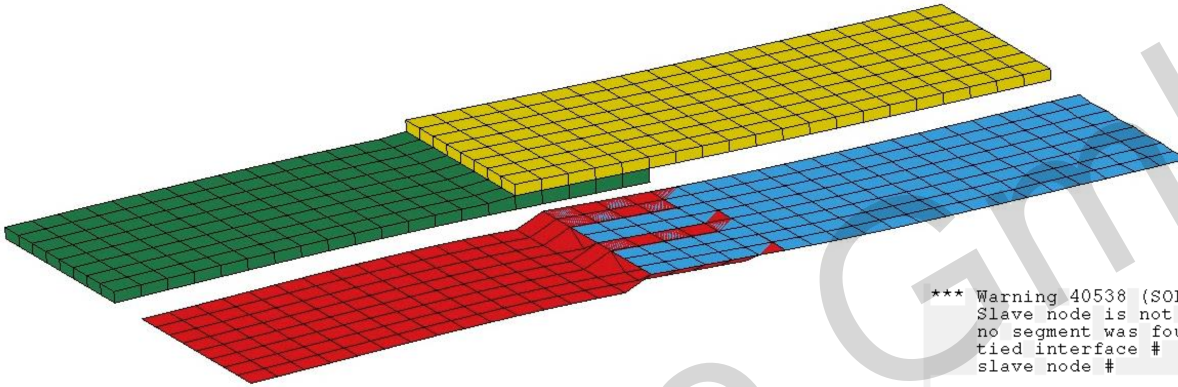


- Tying translational DOF
  - General remarks
    - No rotational DOF are affected, i.e., unrealistic soft behavior for shell elements → should be only used with solid elements
    - Tying similar materials, the master surface should be more coarsely meshed, because the constraints are not applied symmetrically
  - Kinematic constraint method:
    - Cannot be used with rigid bodies (use *CONSTRAINED_EXTRA_NODES, OFFSET)
    - Examples: *CONTACT_TIED_NODES_TO_SURFACE (type 6)
      *CONTACT_TIED_SURFACE_TO_SURFACE (type 2)
  - Penalty-based method (_OFFSET):
    - Can be used with rigid bodies
    - Examples: *CONTACT_TIED_SURFACE_TO_SURFACE_OFFSET (type o2)
      → works best if surfaces are very close (no moments are considered)
      *CONTACT_TIED_SURFACE_TO_SURFACE_BEAM_OFFSET (type b2)
  - Failure:
    - Extremely important to have the contact segment orientation aligned properly as it determines the tensile and compression direction
    - Example: *CONTACT_TIEBREAK_SURFACE_TO_SURFACE (type 9)

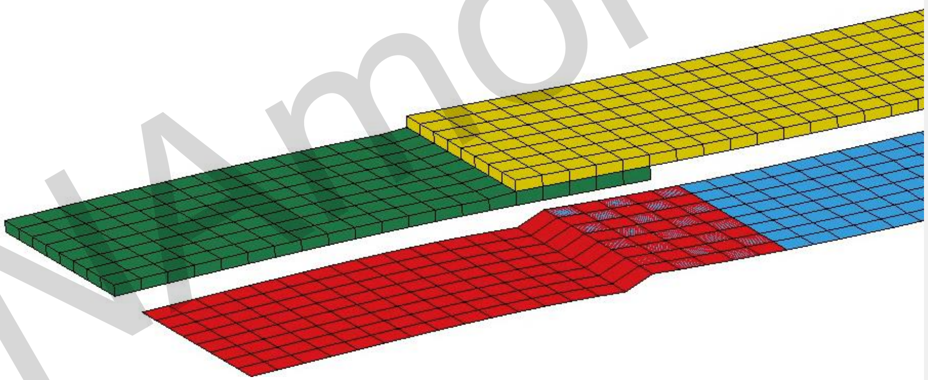
- Tying translational and rotational DOF
  - Translational as well as rotational DOF are affected
  - Kinematic constraint method:
    - Cannot be used with rigid bodies (use *CONSTRAINED_EXTRA_NODES, OFFSET)
    - Examples: *CONTACT_TIED_SHELL_EDGE_TO_SURFACE (type 7)
      *CONTACT_SPOTWELD (type 7)
  - Penalty-based method (_OFFSET):
    - Can be used with rigid bodies
    - Examples:
      *CONTACT_TIED_SHELL_EDGE_TO_SURFACE_OFFSET (type o7)
      *CONTACT_TIED_SHELL_EDGE_TO_SURFACE_BEAM_OFFSET (type b7)
  - Using the {BEAM} and {CONSTRAINED} option,
    moments resulting from the offset are considered



- Numerical example “Tied contacts”
  - Why are some slave nodes not tied?



  - After modification: MST = 4.5; SST = 0.001

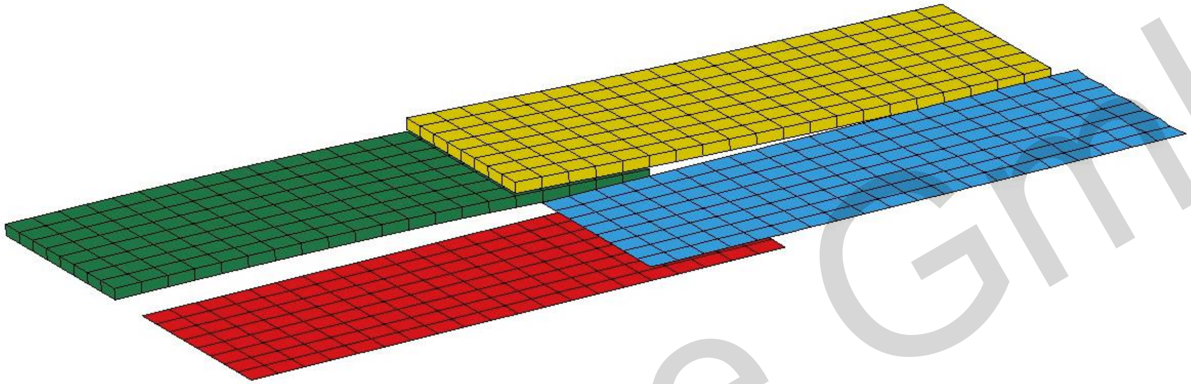


```
*** Warning 40538 (SOL+538)
    Slave node is not constrained
    no segment was found in bucket sort.
    tied interface #              =          1
    slave node #                  =         13

*** Warning 40538 (SOL+538)
    Slave node is not constrained
    no segment was found in bucket sort.
    tied interface #              =          1
    slave node #                  =         14

*** Warning 40538 (SOL+538)
    Slave node is not constrained
    no segment was found in bucket sort.
    tied interface #              =          1
    slave node #                  =         15

*** Warning 40538 (SOL+538)
    Slave node is not constrained
    no segment was found in bucket sort.
    tied interface #              =          1
    slave node #                  =         16

*** Warning 40538 (SOL+538)
    Slave node is not constrained
    no segment was found in bucket sort.
    tied interface #              =          1
    slave node #                  =         17

*** Warning 40538 (SOL+538)
    Slave node is not constrained
    no segment was found in bucket sort.
    tied interface #              =          1
    slave node #                  =         18
```

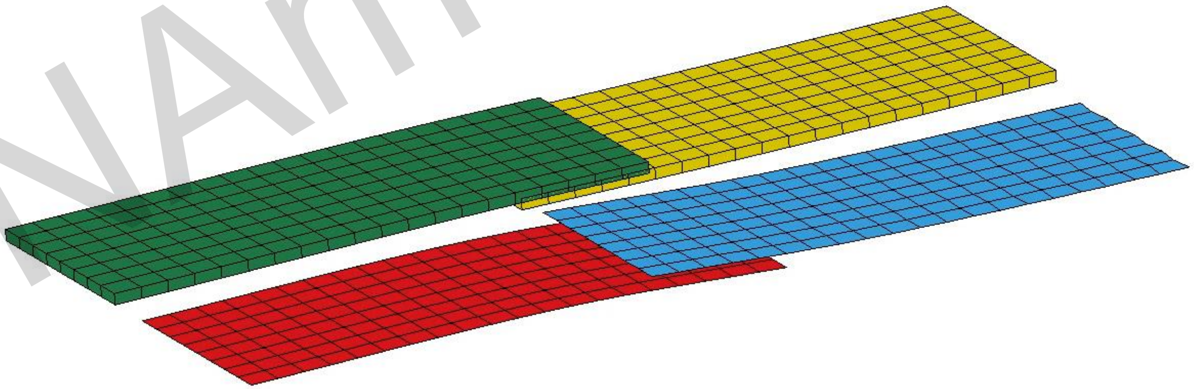
  - How to avoid the warning message “Slave node is not constrained”?
    → use predefined node sets

Numerical example “Tied contacts”

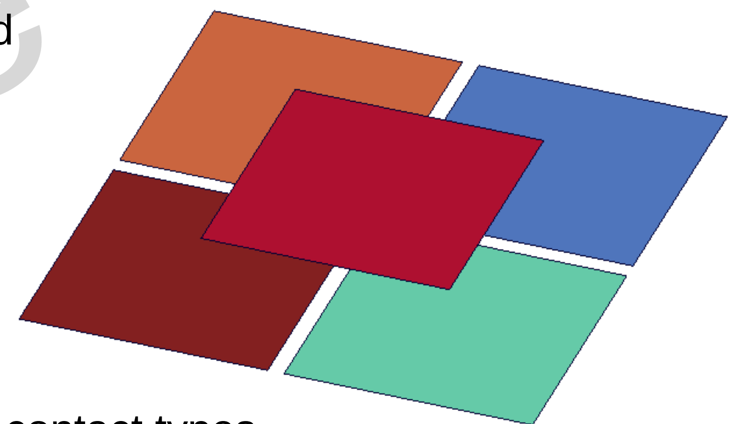
- Numerical example “Tied contacts”
  - *CONTACT_TIED_SHELL_EDGE_TO_SURFACE_OFFSET



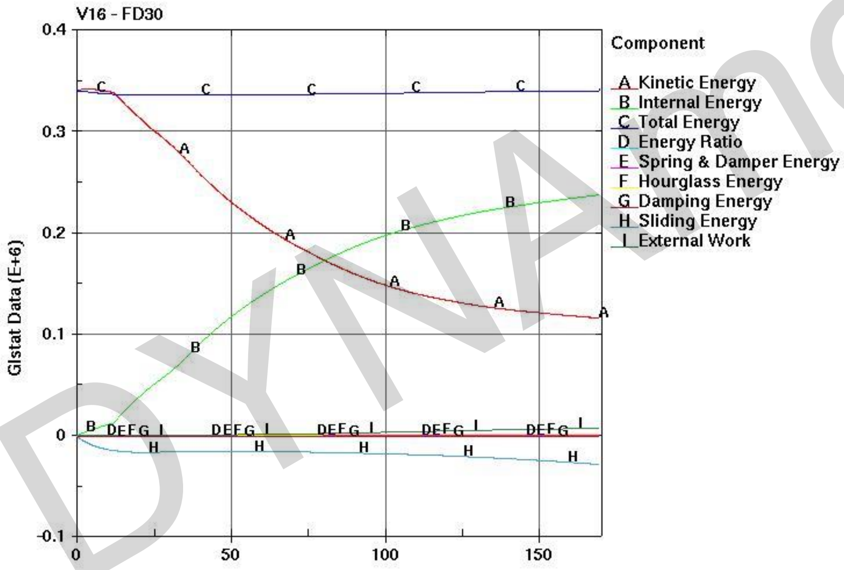
  - *CONTACT_TIED_SHELL_EDGE_TO_SURFACE_BEAM_OFFSET



- **Contact Output**
  - Resultant forces – *DATABASE_RCFORC
    - ASCII file containing resultant contact forces for each master and slave side of each contact interface
    - Forces are written in the global coordinate system
    - *CONTACT_FORCE_TRANSDUCER_{PENALTY/CONSTRAINT}
      - Allows the total contact forces applied by all contacts to be picked up
      - Does not produce any contact force
      - Measures contact forces produced by other contact interfaces
      - Generally, only a slave interface is defined
      - Interactions between two surfaces:
        - » Define furthermore a master surface
        - » Only contact forces between slave and master surfaces are kept
        - » Master surface option is only implemented for the PENALTY option and works only with the AUTOMATIC contact types
        - » **If contact interface is included in more than one two-surface-force-transducer, define FTALL=1 (*CONTROL_CONTACT, CARD 6).**

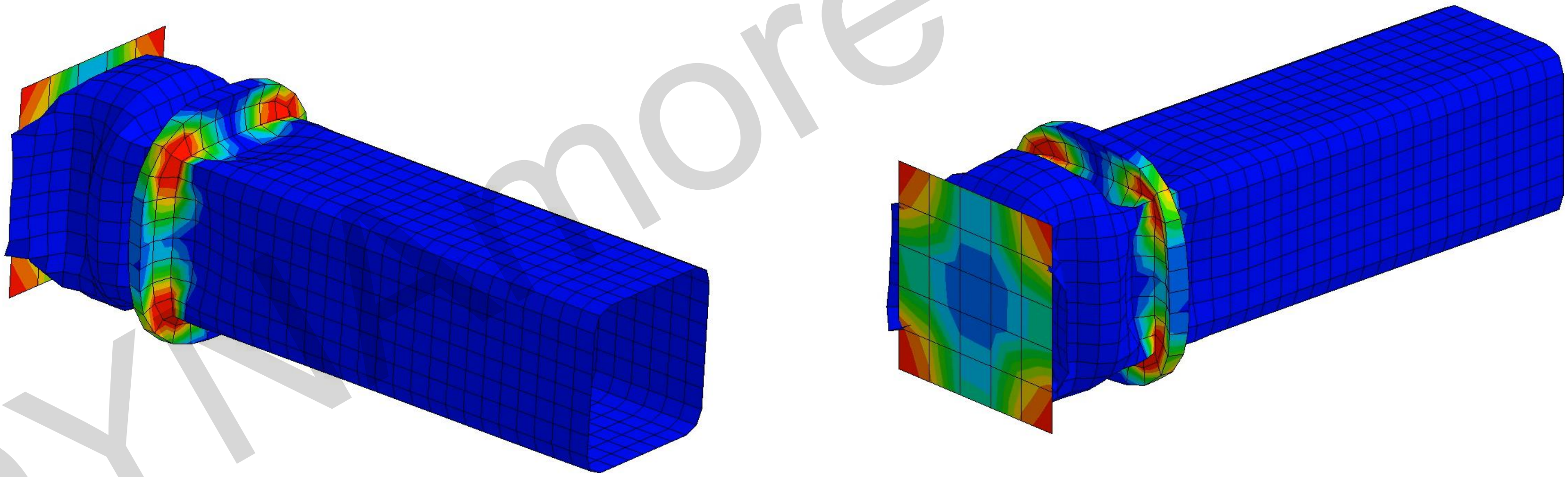


- Nodal forces – *DATABASE_NCFORC
  - Reports contact forces of each node in the NCFORC file (ASCII)
  - SPR and MPR = 1 on Card 1 in *CONTACT_... required
- Global energies – *DATABASE_GLSTAT
  - Contact interface energies for all contacts are written to the ASCII file GLSTAT
- Individual energies – *DATABASE_SLEOUT
  - Contact interface energies for each contact are written to the ASCII output file SLEOUT: Slave-, master-, frictional energy, sum of slave and master
  - In cases where there are more than one contact and the global statistics file (GLSTAT) indicates a problem with contact energy, the SLEOUT is useful for isolating which contact interfaces are responsible



  - Frictional energy is included in slave and master energy as well
  - Sliding energy is the only reasonable quantity to be checked
  - Sliding energy (without frictional energy) should be small compared to internal energy
  - Negative sliding energy as an indicator for a bad model quality → energy is generated

- *DATABASE_BINARY_INTFOR
  - Visualize contact interfaces and produce fringe plots of contact stress
    1. Including a *DATABASE_BINARY_INTFOR command in the input deck
    2. Setting the contact print flags SPR and MPR
    3. Including the option „s=filename“ on the LS-DYNA execution line
  - FRCENG=1 in *CONTROL_CONTACT calculate frictional energy stored as “Surface Energy Density“ in the binary INTFOR file



contact pressure

- **Summary and Recommendations**
  - Overview of contact types
    - □ *CONTACT_{}_NODES_TO_SURFACE_{}
      - – one way contact
      - – slave nodes are checked for penetration with master segments
    - □ *CONTACT_{}_SURFACE_TO_SURFACE_{}
      - – most often a two way contact
      - – slave is checked for penetration with master
      - – master is checked for penetration with slave
    - □ *CONTACT_{}_SINGLE_SURFACE_{}, *CONTACT_{}_GENERAL
      - – two way contact including self contact
    - □ *CONTACT_AUTOMATIC_{}
      - – most automatic contacts work in a similar manner
      - – segment based bucket sort
      - – thickness considerations
      - – release condition similar
    - □ *CONTACT_{}_MORTAR
      - – new contact formulation
      - – primarily intended for implicit time integration

Start you analysis using one global contact:
*CONTACT_AUTOMATIC_SINGLE_SURFACE

- **Modeling guidelines for full vehicle contact**
  - Default values are good reference values
  - Global or local contact
    - One global single-surface contact
      → simplicity in preprocessing
      → numerical robustness
      → computational efficiency
    - Definition of local contact interfaces with non-default parameters, for certain areas of the vehicle that require special considerations
  - Standard penalty-based or soft constraint stiffness method
    - Soft constraint stiffness method depends on global time step
      →SOFSCL can be reduced to 0.04-0.07
    - If standard penalty-based method is used in a global contact definition, the soft constraint approach can be used locally
  - Contact thickness
    - The user is cautioned against setting the contact thickness to an extremely small value as this practice will often cause contact failure
    - For treating contact of very thin shells (<0.5 mm), it may be necessary to increase the contact thickness to prevent contact failure

Use as little contact definitions as possible!
One global contact, very few special contacts!

- Definition of slave set
  - Several ways to define the slave set for the global contact definition. e.g.:
    - – All parts (default)
    - – Included parts by *SET_PART
    - – Excluded parts by *SET_PART
- The option to ignore penetrations on the *CONTROL_CONTACT keyword (IGNORE=1, IGNORE=2) is recommended
  (check: Penetration free for SST~0.5 mm, OPPT=0, IGNORE=0)
- Define FTALL=1 (*CONTACT_FORCE_TRANSDUCER, *CONTROL_CONTACT)
- When friction is expected to play a significant role, the use of *DEFINE_FRICTON to specify friction coefficients on a part-by-part basis is recommended
- Uniform meshes improve result
- Avoid sharp corners
- Make master side with coarser mesh for one way treatment
- Automatic contact input simplifies problem translation
- Contact stiffness affects time step

Thank you for your attention!

Your LS-DYNA distributor and more