Nonlocal Damage and Failure Options in LS-DYNA®

Gunther Blankenhorn, Jason Wang, Lee Bindeman (LSTC), Tobias Erhart (DynaMore)

Bamberg, Germany, March 17, 2018

LSTC
Livermore Software Technology Corp.

Outline

- Introduction
  - Motivation
- Nonlocal Approaches in LS-DYNA
  - Geometrical Entity
  - Physical Quantity
  - Smoothing Function
  - LS-DYNA Nonlocal Features
- *USER_NONLOCAL_SEARCH
  - Features and usage
  - Keyword
  - subroutine user_nunonl_smooth
  - Challenges (MPP)
  - Examples
  - keyword
- Summary

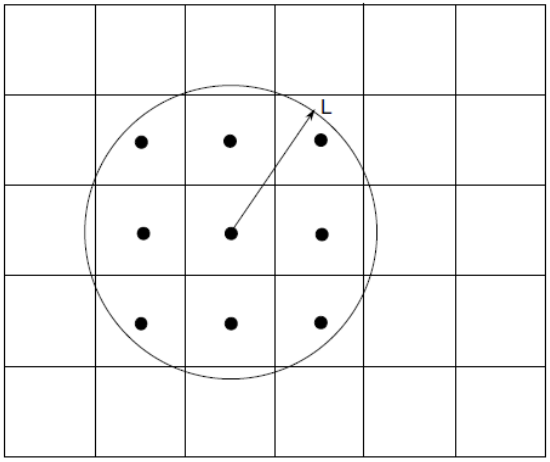
Motivation

Advantages of a nonlocal approach

- Reduced mesh size sensitivity on material strain softening so results converge to a unique solution as the mesh is refined.
- Without a nonlocal approach, strains can tend to localize so mesh refinement leading to results which can change significantly from mesh to mesh.
- The nonlocal approach can be a help in predicting the onset and the evolution of material softening and failure.

How does a nonlocal approach work?

- Use of an functional dependency based on data from neighboring elements, thus the term "nonlocal"
- This contrasts with a local approach, where damage and failure are based only on properties of the current element.
- Data is smoothed by a weighted average within geometric entities (circle, sphere, ellipsoids, etc.).
- Data is at integration point or node.



Example for a geometric entity: circle

LS-DYNA Nonlocal Features

- *MAT_NONLOCAL
  - Using a sphere (radius) as a geometric search entity to determine which elements are included in the averaging.
  - The physical quantities can be chosen from the integration point history.
  - Smoothing function is hard coded.
  - Implementation based on Worswick and Lalbin [1999] and Pijaudier-Cabot and Bazant [1987].

- *MAT_PLASTICITY_WITH_DAMAGE_ORTHO_RCDC1980
  - Using a sphere (radius) as a geometric search entity.
  - Physical quantities and smoothing function are hard coded.

- *MAT_CODAM2
  - Using a sphere (radius) as a geometric search entity.
  - Physical quantities and smoothing function are hard coded.
  - Implementation based on Forghani [2011], Williams et al. [2003].

LS-DYNA Nonlocal Features (cont.)

- *MAT_ADD_EROSION
  - Designed to simulate glass behavior.
  - Calculate the internal energy in a user defined radius
  - Erosion appears if the internal energy in this geometric entity is above a user defined threshold.

- *USER_NONLOCAL_SEARCH
  - Geometric entity is a sphere which also can be scaled in the three coordinate directions to generate an ellipsoid.
  - The local coordinate system of the ellipsoid can be arbitrarily oriented.
  - Physical quantities can be choose from either the node or the integration point.
  - The smoothing function can be defined via a user subroutine and is therefore also arbitrary.

*USER_NONLOCAL_SEARCH

Features:

- Geometry entity is a sphere or ellipsoid (can be scaled in all three material directions).
- Use global or local (material) coordinate system to define orientation of geometric search entity.
- Full access to integrations point and nodal data in the search geometry
- A user subroutine is provided to give flexibility as how data is averaged

Usage:

- Two components needed to use *USER_NONLOCAL_SEARCH
  - The keyword itself *USER_NONLOCAL_SEARCH
  - `subroutine user_nunonl_smooth` (located in dyn21.F)

Keyword

- Can be defined several times in one input.
- Choose the parts included in the search and averaging.
- Define geometric search entity
  - Radius and three scale factors to define sphere or ellipsoid.
  - RADIUS: search distance from element center
  - SA, SB, SC: scale factor for a, b, c directions
  - VOLTYPE: Orientation (element or material axis)
- NFREQ: frequency to collect history data
- IFUNC: user defined function type (switch for multiple function)
- UCONST: up to 48 user parameters (P#) which are passed to the user subroutines
- Choose location of the history (H#)variables to be gathered.

SLAVE	MASTER	STYPE	MTYPE	RADIUS	SA	SB	SC
NFREQ	VOLTYPE	IFUNC	UCONST				
If uconst >0 and 48 max							
P1	P2	P3	…				
H1	H2	H3	…				

dyn21.f Interface: subroutine user_nunonl_smooth

```
      DO II=1,NELEMS
        ie = neid(ii)
        if(ixh(1,ie).eq.0) cycle
c vvvvvvvvvvv operations of neighboring data
        nstr = nrang(ii)
        nend = nrang(ii+1)-1
        do j=nstr,nend
          je=nlist(j)
          if(ifail(je).eq.0) cycle
          jje=nsrt(je)
          do ip=1,nip
            do k=1,nhisv
              htmp(k,ip) = histv(k,ip,je)
            enddo
          enddo
        enddo
c ^^^^^^^^^^^^^^^ of operation of the neighbor
c vvvvvvvvvvvvvv target element
        lav=lochvh(ie)-1
        do ip=1,nip
          do k=1,nhisv
            ipos = list(k)+7+(ip-1)*nmtcon
            auxvec(lav+ipos) =htmp(k,ip)
          enddo
        enddo
c ^^^^^^^^^^^^^^ target element
      ENDDO
```

Loop of master elements

Deleted?

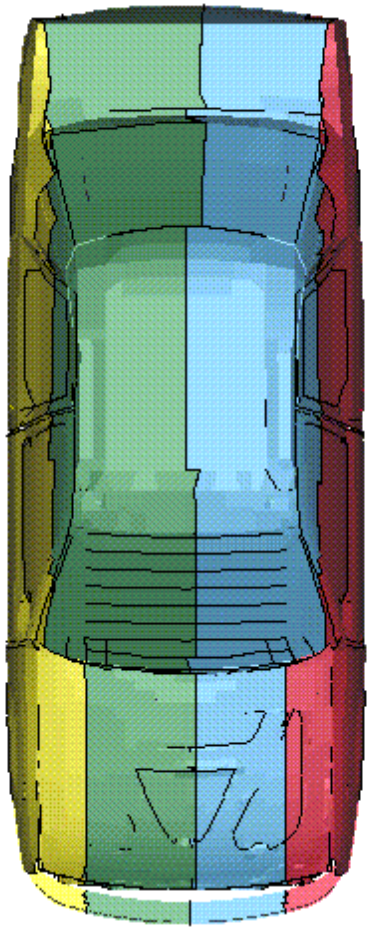
Loop of slave elements

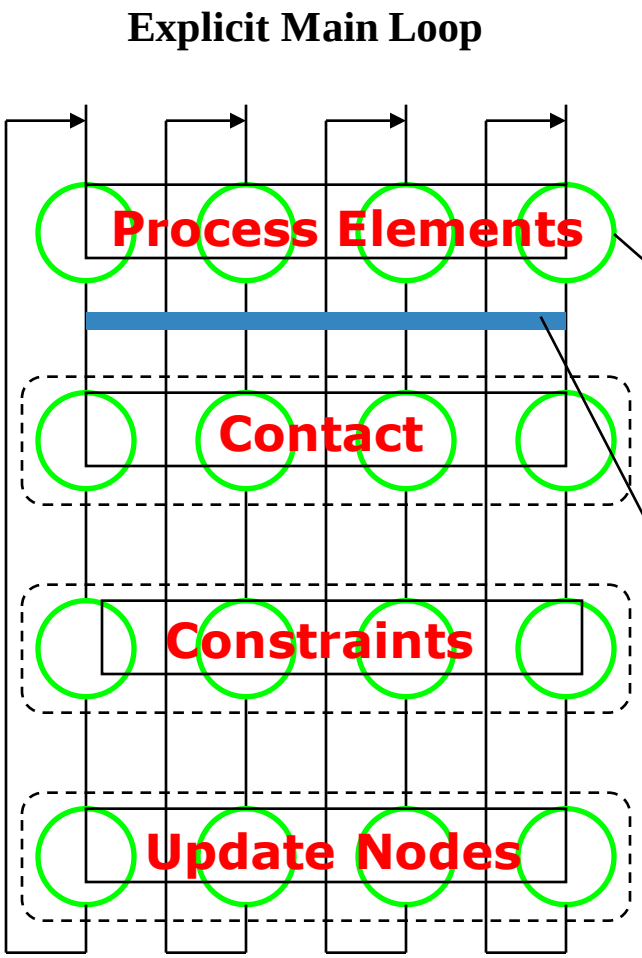
Deleted?

- History variables
- Element ID
- Element connectivity
- Nodal coordinates

Loop through all IP and history values

Challenges (MPP)





Regular usermat interface

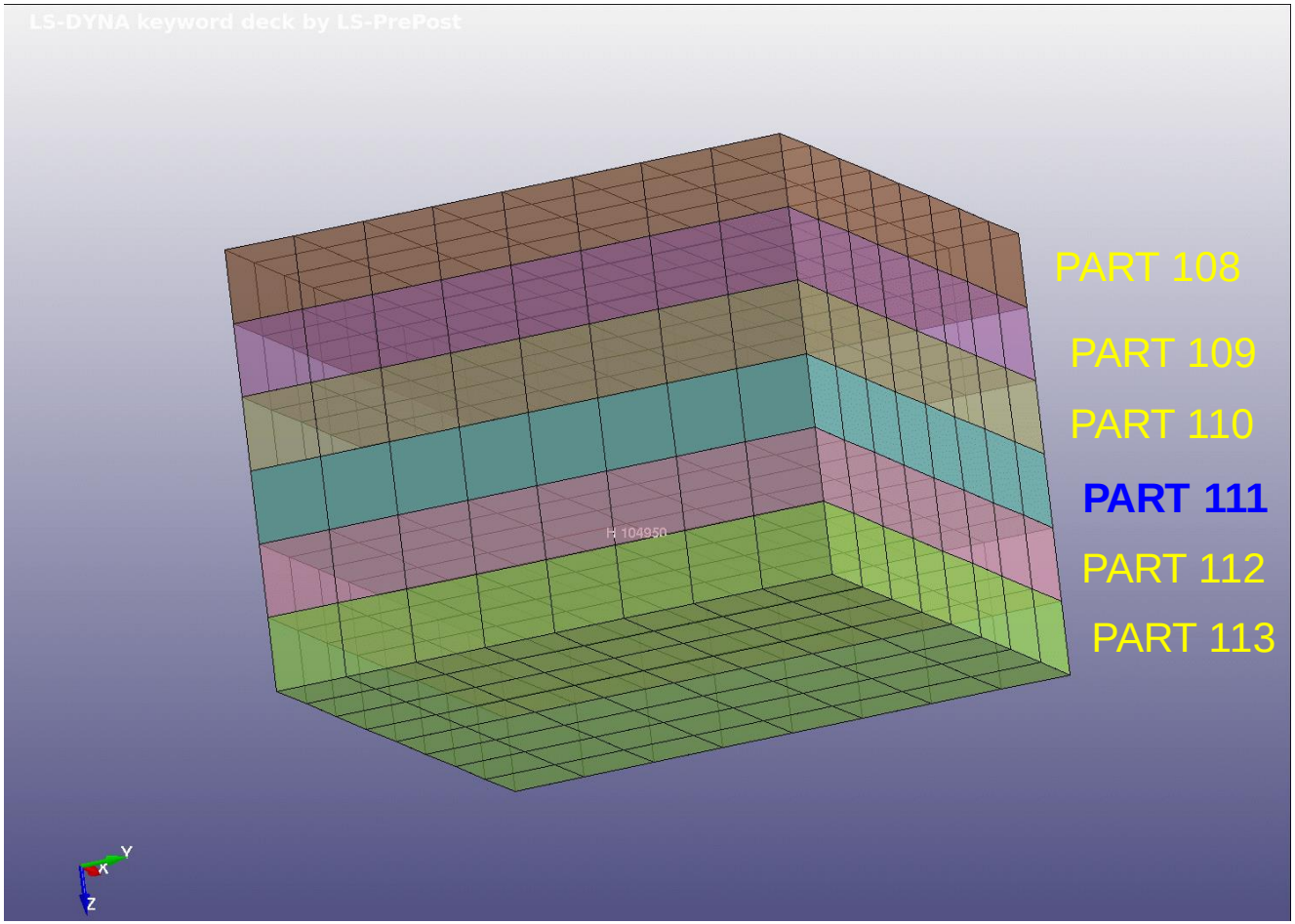
- Processor may work on different type of element/material model
- Elements in each processor are gathered in groups. Element calculation is performed group-by-group
- The history variables of neighbor elements cannot be accessed
- No communication allowed

***USER_NONLOCAL_SEARCH**

- The history variables of neighbor elements are collected for user
- All data has been collected for user and MPI communication is transparent to users

Example

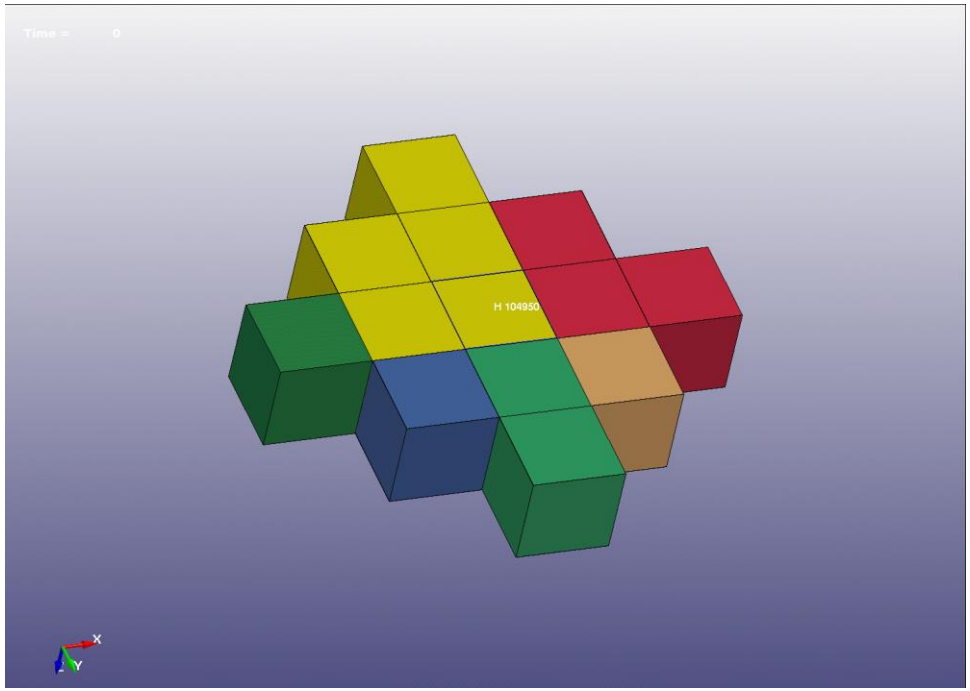
- 6 parts, 64 elements/part
- SOLID formulation = 1
- *MAT_41
- Target part 111



Example

Specification

- Axis related to the global coordinate system VOLTYPE = 1
- Radius of geometric entity is R= 2.0
- Scale factors are SA = SB = SC = 1.0
- One layer SLAVE = MASTER



Elements of same color reside on same rank

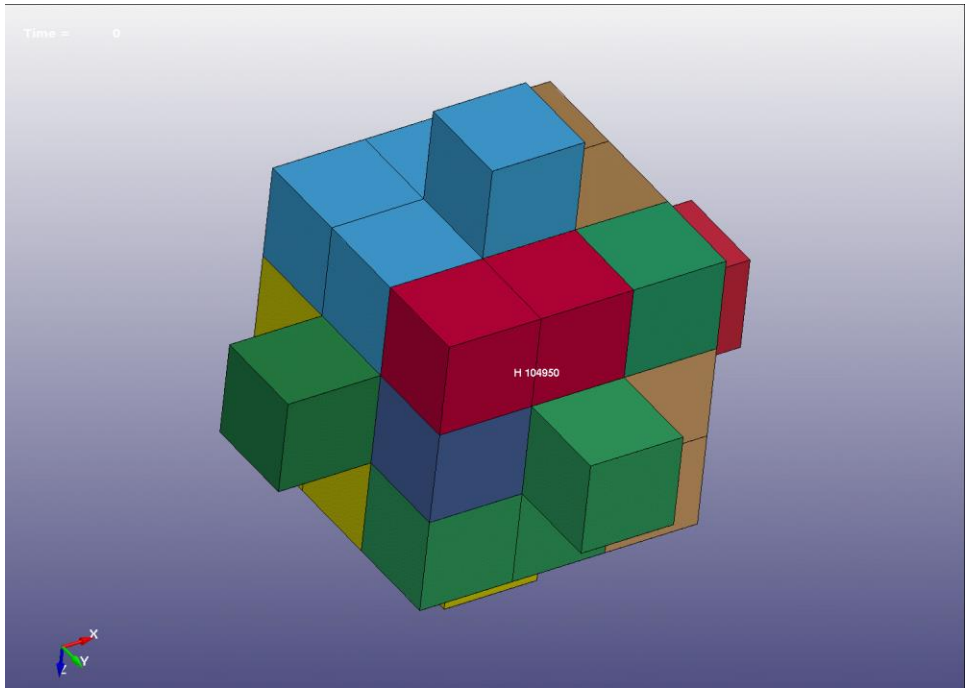
Slave element ID's reported in user subroutine for
Master element 104950
Slave > 104849
Slave > 104949
Slave > 104850
Slave > 104750
Slave > -105049
Slave > -105050
Slave > -105150
Slave > -104851
Slave > -104951
Slave > -104952
Slave > -105051
Slave > -104948

From remote Processors

Example

Specification

- Axis related to the global coordinate system VOLTYPE = 1
- Radius of geometric entity is R = 2.0
- Scale factors are SA = SB = SC = 1.0
- All elements SLAVE = all parts (sphere)



Elements of same color reside on same rank

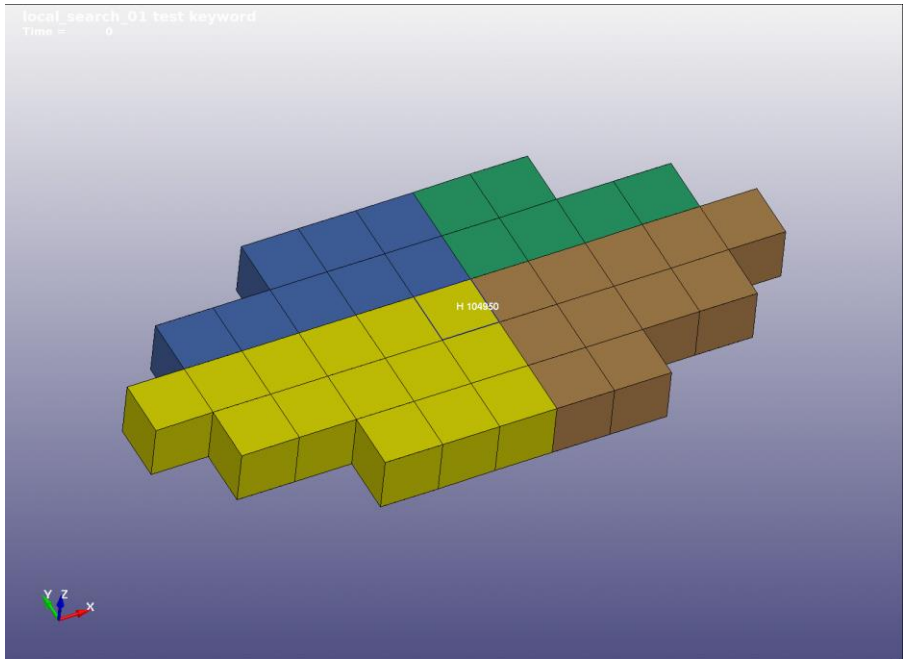
For all slave elements the following information is available

- History variables of integration point(s)
- Element ID
- Node ID
- Flag if element is erroded
- Element connectivity
- Nodal coordinates

Example

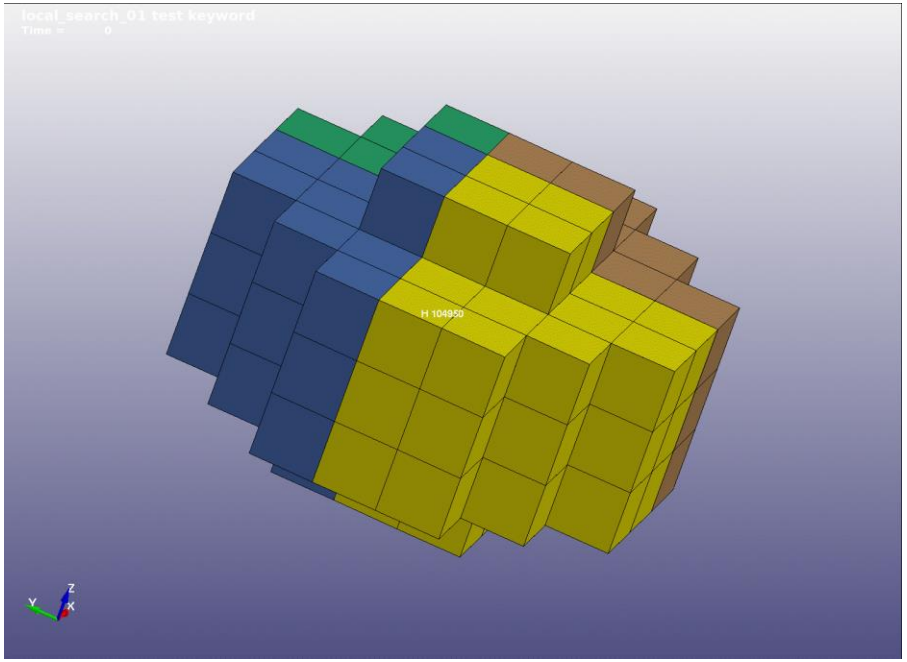
Specification

- Axis related to the global coordinate system VOLTYPE = 1
- Radius of geometric entity is R = 2.0
- Scale factors are SA = 2.0, SB = SC = 1.0
- One layer SLAVE = MASTER



Specification

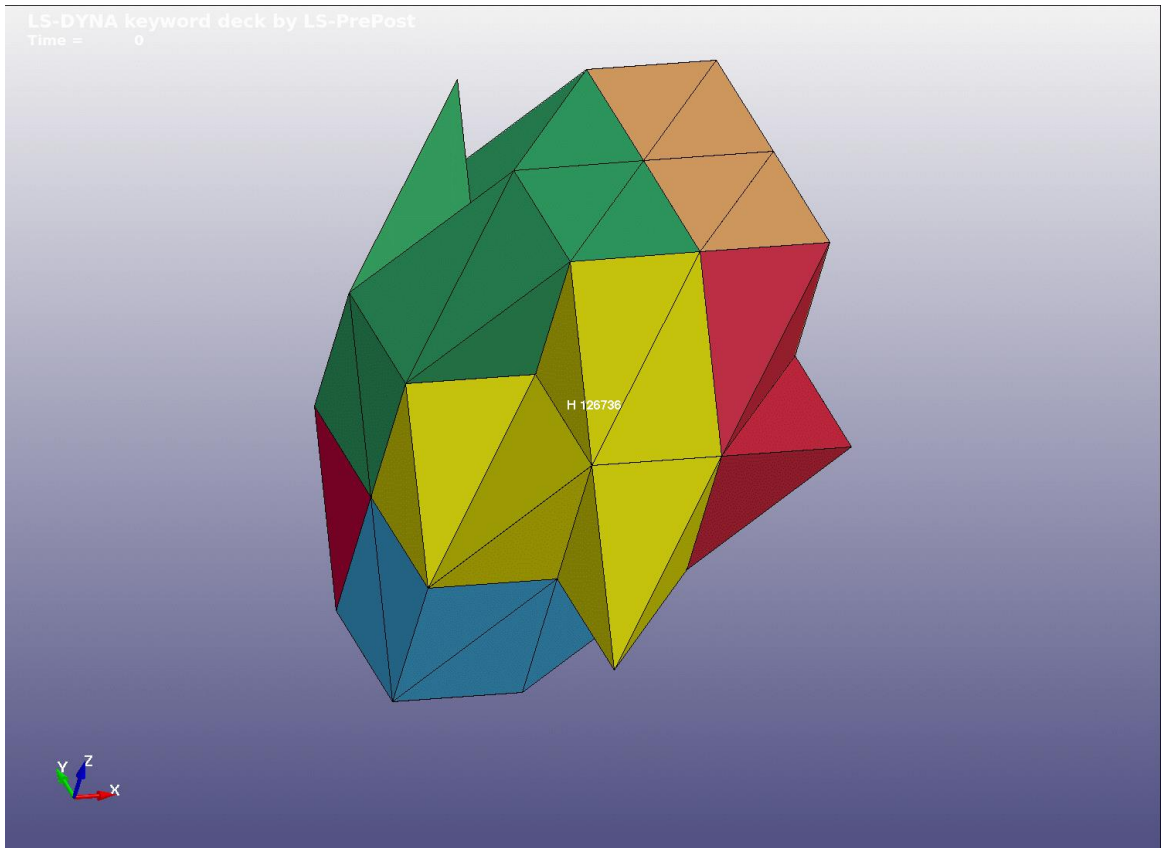
- Axis related to the global coordinate system VOLTYPE = 1
- Radius of geometric entity is R = 2.0
- Scale factors are SA = 3.0, SB = 3.0, SC = 1.5
- SLAVE = all parts



Example

Specification

- Axis related to the local (material) coordinate system VOLTYPE = 2 ELEMENT_ORTHO using vectors (1.0 ,1.0, 1.0) and (1.0, 1.0, 0.0)
- Radius of geometric entity is R = 1.0
- Scale factors are SA = 2.0, SB = 1.5, SC = 1.0
- SLAVE = all parts



Summary

- Allow multiple keyword and each uses its own code path
- Master and slave data are all at current time
- All slave data are collected in different space and won't be overwritten by updating history variables
- MPP enabled
- Available after svn 124005 R10 (will be in R10.2), R11 beta and Dev beta

Current limitation

- *ELEMENT_SOLID_(ORTHO) (can be extended by request)
- Solid formulation 1,16,19,21 (can be extended by request)
- ESORT=0
- Parts have to use same material model within each card