DYNAmore Express

Overview on Airbag-Modeling Possibilities in LS-DYNA

Steffen Mattern, DYNAmore GmbH

October 2020

Outline

- Introduction
- Uniform Pressure (UP)/ Control Volume (CV) approach
- Arbitrary Lagrangian Eulerian (ALE)
- Corpuscular Particle Method (CPM)

Introduction

- the representation of airbags in modern CAE models has become a standard application in crash simulation
- first, the focus was on capturing the influence and improve the results of dummy impact on fully inflated airbags
- as quality and level of detail of CAE models increased over the years, more sophisticated airbag models were necessary to discuss
  - the influence of different folding schemes
  - size, geometry and position of vent holes
  - interaction of the airbag with its surrounding, especially in the deployment phase
  - out-of-position load cases
- over the years, three different methods for airbag modeling have been implemented in LS-DYNA
  - the Uniform Pressure (or Control Volume) approach (UP/ CV)
  - the Arbitrary Lagrangian Eulerian approach (ALE)
  - the Corpuscular Particle Method (CPM)
- each method is still available in current versions of LS-DYNA (but probably not developed further)
- further relevant points to set-up a CAE model of an airbag
  - material modeling
  - contact modeling

Introduction

- over the years, three different methods for airbag modeling have been implemented in LS-DYNA
  - the Uniform Pressure (or Control Volume) approach (UP/ CV)
  - the Arbitrary Lagrangian Eulerian approach (ALE)
  - the Corpuscular Particle Method (CPM)
- each method is still available in current versions of LS-DYNA (but probably not developed further)
- further relevant points to set-up a CAE model of an airbag
  - material modeling
  - contact modeling
  - …

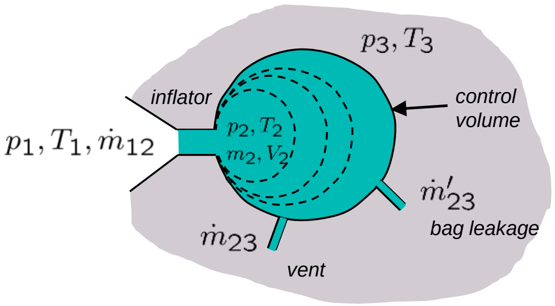
- briefly discussed within the following 45 minutes

- please visit the advanced seminars for details:
  - CPM Airbag Modeling
  - LS-DYNA Compact: CPM Airbag Modeling

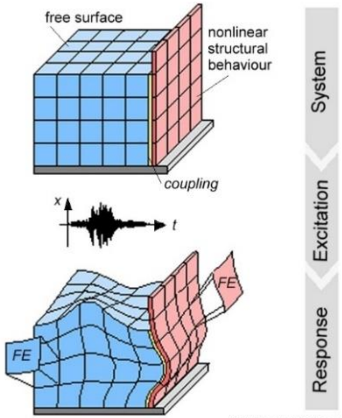
https://www.dynamore.de/en/training/seminars/

Different approaches for modeling of airbags in LS-DYNA:

Uniform pressure method (UP/CV)

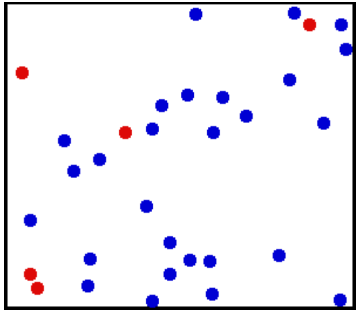


Arbitrary Lagrangian-Eulerian method (ALE)



(c) University of Stuttgart

Corpuscular Particle method (CPM)



https://en.wikipedia.org/

time of implementation in LS-DYNA

Outline

- Introduction
- Uniform Pressure Models (UP)
- Arbitrary Lagrangian Eulerian (ALE)
- Corpuscular Particle Method (CPM)

Uniform Pressure (UP/CV) approach

- Uniform pressure distribution is assumed
- Application of forces perpendicular to the defined airbag surfaces
- There is no discretization of the fluid flow
- Concept is based upon scalar thermodynamic equations
- Pressure is applied normal to the airbag fabric
- Widely used for side and front-crash simulations

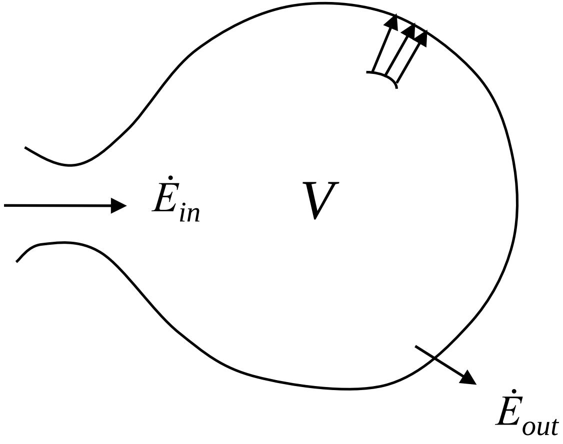
$$\dot{E}_{in} = \dot{m}_{in}\, c_p\, T_{in}$$

$\dot{E}_{in}$ → Energy into airbag by mass flow (inflator)

$\dot{E}_{out}$ → Energy out of airbag by mass flow (vents, leakage)

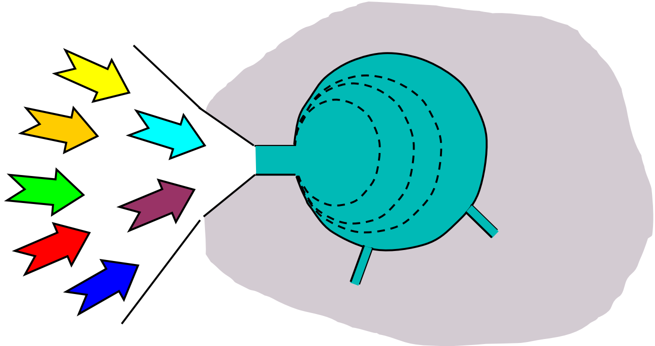
$$\boxed{\dot{E} = \dot{E}_{in} - \dot{E}_{out} - p\,\dot{V}}$$

$$p = (\gamma - 1)\frac{E}{V}$$



Uniform Pressure (UP/CV) approach

- Different Uniform Pressure models available via `*AIRBAG_...` keyword
- rather simple models currently used e.g. for modeling tires
- Most commonly used for airbags: **hybrid model** (`*AIRBAG_HYBRID`)
  - Gas parameters of up to 17 gas fractions including initial air
  - Temperature vs. time data for the mixture is needed as input (usually measured through tank test)
  - Detailed input parameters for gas outflow (vents, fabric porosity) are taken into account
  - Jetting parameters may be specified



Individual gas fractions

Theory of Wang's hybrid inflation model

- Energy balance equation for the airbag control volume

$$\boxed{\frac{d}{dt}(mu)_{cv} = \sum \dot{m}_i h_i - \sum \dot{m}_o h_o - \dot{W}_{cv} - \dot{Q}_{cv}}$$

$\frac{d}{dt}(mu)_{cv}$ … Rate of change of energy

$\sum \dot{m}_i h_i$ … Energy into airbag by mass flow (inflator)

$\sum \dot{m}_o h_o$ … Energy out of airbag by mass flow (vents leakage)

$\dot{W}_{cv} = \int P \, \mathrm{d}\dot{V}$ … Work done by airbag expansion

$\dot{Q}_{cv}$ … Energy out by heat transfer through airbag surface

- Temperature dependent specific heat capacities are used

HYBRID definition in LS-DYNA allows quadratic definition of $c_p = f(A, B, C, T)$

$$\boxed{\bar{c}_v = \bar{a} + \bar{b}T - \bar{r}} \qquad \boxed{\bar{c}_p = \bar{a} + \bar{b}T}$$

$\bar{a}, \bar{b}$ … Material parameters $\left(a : \left[\frac{\mathrm{J}}{\mathrm{kg\,K}}\right], b : \left[\frac{\mathrm{J}}{\mathrm{kg\,K^2}}\right]\right)$

T … Temperature [K]

$\bar{r}$ … Gas constant $\left(r = 8.314 \frac{\mathrm{J}}{\mathrm{kg\,K}}\right)$

- For ideal gas mixtures the molecular weight is given as

$$\boxed{M = \frac{1}{\sum f_i / M_i}}$$

M_i … molecular weight of fraction i

f_i … mass fraction of gas i

Theory of Wang’s hybrid inflation model

- The constant volume and pressure specific heats are obtained from

$$\boxed{c_v = \sum f_i \, c_{v,i}} \quad \boxed{c_p = \sum f_i \, c_{p,i}}$$

$c_{v,i}$ … constant volume specific heat of fraction i
$c_{p,i}$ … constant pressure specific heat of fraction i

- Energy insertion through mass flow

$$\boxed{\sum \dot{m}_i \, h_i = \sum \dot{m}_i \left(a_i \, T_i + \frac{b_i \, T_i^2}{2} \right)}$$

$\dot{m}_i$ … mass inflow vs. time ($\equiv \dot{m}_{12}$)
T_i … temperature vs. time
a_i ,b_i … constants for gas i
h_i … specific enthalpy

- Energy discharge through mass flow

$$\boxed{\sum \dot{m}_o \, h_o = \sum \dot{m}_o \left[\sum_{\text{gases}} f_i \left(a_i \, T_{cv} + \frac{b_i \, T_{cv}^2}{2} \right) \right]}$$

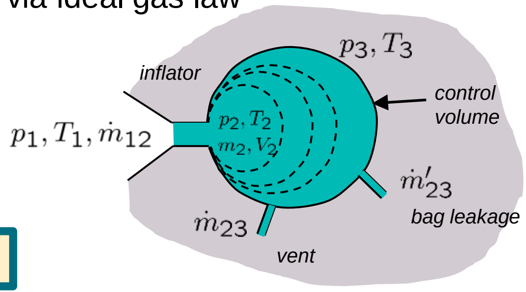
m_0 … mass outflow by vents ($\dot{m}_{23}$) and fabric leakage ($\dot{m}'_{23}$)
T_{cv} … temperature of control volume
a_i, b_i … constants for gas I

- Conservation of mass

$$\dot{m}_{cv} = \dot{m}_i - \dot{m}_o \qquad m_{cv} = \int \dot{m}_{cv} \, \mathrm{dt}$$

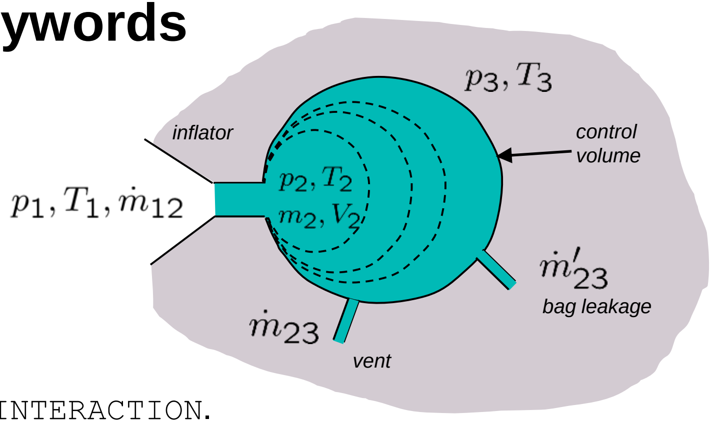
- Pressure is obtained via ideal gas law

$$p_{cv} = \frac{m_{cv} \, r \, T_{cv}}{V_{cv}}$$



for more details please refer to LS-DYNA Theory Manual, section 33

Uniform Pressure models in LS-DYNA – Keywords



- In UP or CV-models the spatial domain, which shall represent a closed volume, has to be defined by shells or segments (`SID` and `SIDTYP`).
- Any holes or unconnected segments will be automatically closed with planes by the algorithm.
- By default, the normals of the shells or segments shall point outwards of the airbag.
- Interaction between different airbags can be defined by using `*AIRBAG_INTERACTION`. In this case an individual ID has to be given to each `*AIRBAG`-card.

CARDID	**ABID**	**HEADING**						
CARD1a	**SID**	**SIDTYP**	**RBID**	**VSCA**	**PSCA**	**VINI**	**MWD**	**SPSF**

`ABID` … Airbag ID

`HEADING` … Airbag Title

`SID` … Set ID

`SIDTYP` … Set type: EQ.0: segment
NE.0: part set ID

Uniform Pressure models in LS-DYNA – Keywords

CARDID	**ABID**	**HEADING**						
CARD1a	**SID**	**SIDTYP**	**RBID**	**VSCA**	**PSCA**	**VINI**	**MWD**	**SPSF**

`RBID`	… Rigid body part ID that defines user defined sensor subroutine (`RBID>0`) or internal sensor subroutine (`RBID<0`). Optional card(s) A and B have to be defined!
`VSCA/PSCA`	… Volume and pressure scale factors (needed if inflator has different units)
`VINI`	… Initial filled volume (needed if inflator has different units)
`MWD`	… Mass weighted damping factor, D

$$F_i^d = m_i\, D(v_i - v_{cg})$$

m_i … nodal mass

v_i … nodal velocity

v_{cg} … mass weighted avg. velocity of CV-structure

`SPSF` … Stagnation pressure (maximum pressure action on a flat plate orientated perpendicular to a steady state flow field) scale factor (0-1), alternative to `MWD`

$$p = \gamma\, \rho\, v^2$$

γ … SPSF

ρ … ambient air density

v … normal velocity of the CV relative to the ambient velocity (stagnation velocity)

→ Further optional cards available (1b – 1f, 2a – 2b) – see manual

Uniform Pressure models in LS-DYNA – Keywords

- `*AIRBAG_SIMPLE_PRESSURE_VOLUME`

CARD3	CN	BETA	LCID	LCIDR				

- Mainly used for modeling air in tires
- No leakage and no temperature and input mass flow is assumed
- Scale factor BETA, CN coefficient (unit of pressure) and optionally a load curve for CN as a function of time can be defined.
- In addition a load curve pressure vs. relative volume may be specified directly

$$\text{pressure} = \beta \frac{CN}{\text{relative volume}}$$

$$\text{relative volume} = \frac{\text{current volume}}{\text{initial volume}}$$

Uniform Pressure models in LS-DYNA – *AIRBAG_HYBRID

CARD1	**ATMOST**	**ATMOSP**	**ATMOSD**	**GC**	**CC**	**HCONV**		
CARD2	**C23**	**LCC23**	**A23**	**LCA23**	**CP23**	**LCCP23**	**AP23**	**LCAP23**
CARD3	**OPT**	**PVENT**	**NGAS**	**LCEFR**	**LCIDM0**	**VNTOPT**		
CARD4	**LCIDM**	**LCIDT**		**MW**	**INITM**	**A**	**B**	**C**
CARD5	**FMASS**							

`ATMOST`	…	Atmospheric temperature
`ATMOSP`	…	Atmospheric pressure
`ATMOSD`	…	Atmospheric density
`GC`	…	Universal gas constant
`CC`	…	Conversion constant, set to 1.0
`HCONV`	…	Effective heat transfer coefficient
`C23`	…	Vent orifice coefficient
`LCC23`	…	C23 as a function of time
`A23`	…	Vent orifice area
`LCA23`	…	A23 as a function

`CP23`	…	Orifice coefficient for leakage (fabric porosity)
`LCCP23`	…	`CP23` as a function of time
`AP23`	…	Area for leakage (fabric porosity)
`LCAP23`	…	`A23` as a function of absolute pressure
`OPT`	…	Fabric venting option, if used `CP23`, `LCCP23`, `AP23`, `LCAP23` are set to zero. For the options 1..8 different Wang Nefske formulas for venting and leakage are used, see also LS-DYNA 971 Users Manual
`PVENT`	…	Pressure when venting begins

Uniform Pressure models in LS-DYNA – `*AIRBAG_HYBRID`

CARD1	**ATMOST**	**ATMOSP**	**ATMOSD**	**GC**	**CC**	**HCONV**		
CARD2	**C23**	**LCC23**	**A23**	**LCA23**	**CP23**	**LCCP23**	**AP23**	**LCAP23**
CARD3	**OPT**	**PVENT**	**NGAS**	**LCEFR**	**LCIDM0**	**VNTOPT**		
CARD4	**LCIDM**	**LCIDT**		**MW**	**INITM**	**A**	**B**	**C**
CARD5	**FMASS**							

`NGAS`	…	Number of gas fractions defined below, including initial air
`LCIDM`	…	Load curve id for inflator mass flow rate vs. time
`LCIDT`	…	Load curve id for inflator gas temperature vs. time
`MW`	…	Molecular weight
`INITM`	…	Initial mass fraction of gas component
`A, B, C`	…	Parameter to obtain the temperature dependence of c_v, c_p [J/(mol K)], [J/(mol K^2)], [J/(mol K^3)]

`FMASS`	…	Fraction of additionally aspirated mass
`LCEFR`	…	Load curve id for exit flow rate vs. pressure
`LCIDM0`	…	Load curve id for inflator’s total mass inflow rate
`VNTOPT`	…	additional options for venting area definition

Remark:

Card 4 and Card 5 have to be repeated for the number of gas fraction defined with `NGAS`

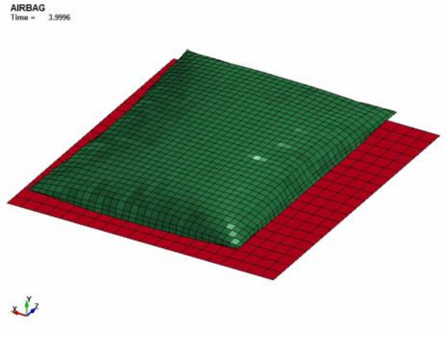
Uniform Pressure models in LS-DYNA – further keywords

- `*AIRBAG_ADIABATIC_GAS_MODEL`
  - Expansion of a pre-loaded volume via gamma law
  - No outflow out of the airbag (vents, leakage) can be defined
- `*AIRBAG_SIMPLE_AIRBAG_MODEL`
  - rudimentary UP model
  - inflow and outflow can be defined
- `*AIRBAG_WANG_NEFSKE`
  - predecessor of hybrid airbag model
  - parameters of gas mixture have to be defined
- `*AIRBAG_HYBRID_JETTING`
  - jetting option provides a simple model to take into account local gas flow effects during unfolding
  - forces in the line of sight of a virtual origin are locally scaled
- `*AIRBAG_INTERACTION`
  - define interaction of two connected airbags which vent into each other
  - allows modeling of multi-chamber airbags

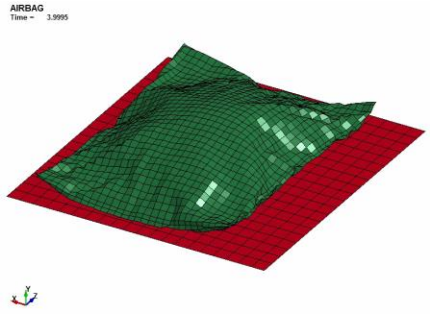
Uniform Pressure approach – pros and cons

- pros
  - Numerically cheap and robust method
  - Airbag definition is quite simple
- cons
  - Fluid is represented by pressure boundary condition → local effects are missing
  - Deployment phase is quite inaccurate
  - Validation of the complete airbag model (Bag + Inflator) necessary

- example – inflation test w/o jetting



- example – inflation test with jetting

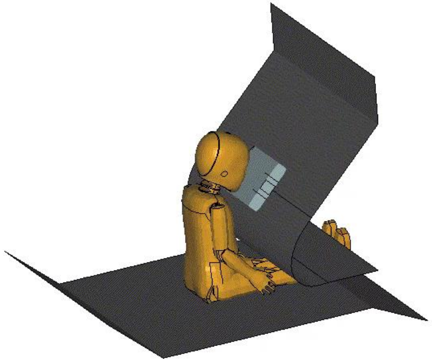


Outline

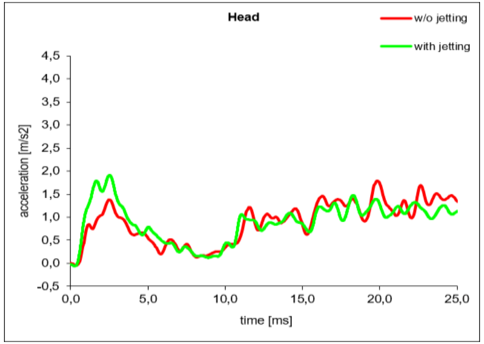
- Introduction
- Uniform Pressure Models (UP)
- Arbitrary Lagrangian Eulerian (ALE)
- Corpuscular Particle Method (CPM)

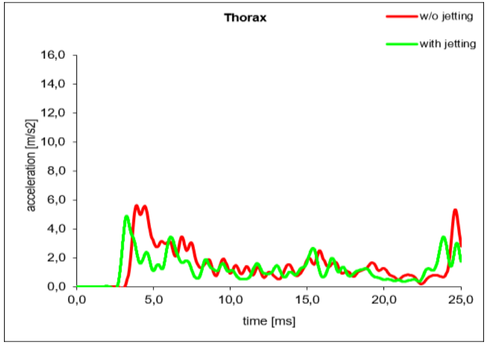
Arbitrary Lagrangian Eulerian (ALE) – Motivation

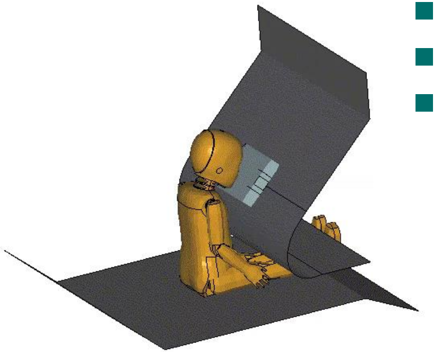
- Accurate representation of deployment phase difficult with UP
  - opening of airbag cover
  - interaction of airbag with surroundings
  - out-of-position
- discretization of gas necessary in order to capture local gas flow effects



- out-of-position simulation
- 3 year old child
- UP w/o jetting







- out-of-position simulation
- 3 year old child
- UP with jetting

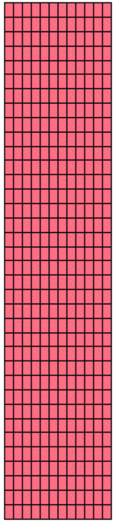
higher accelerations expected in the first few milliseconds

Arbitrary Lagrangian Eulerian (ALE) – Motivation

- Idea: discretization of structure **and** fluid domain with finite elements
- Problem: large mesh distortion problematic with standard (Lagrangian) finite elements
- Example: Taylor-Bar (Courtesy of Lars Olovsson)

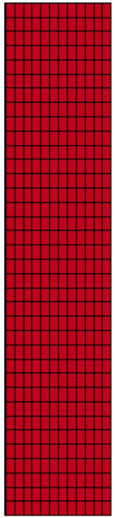
Problem:

- Large deformations/distortions
- Element performance degrades



Solution:

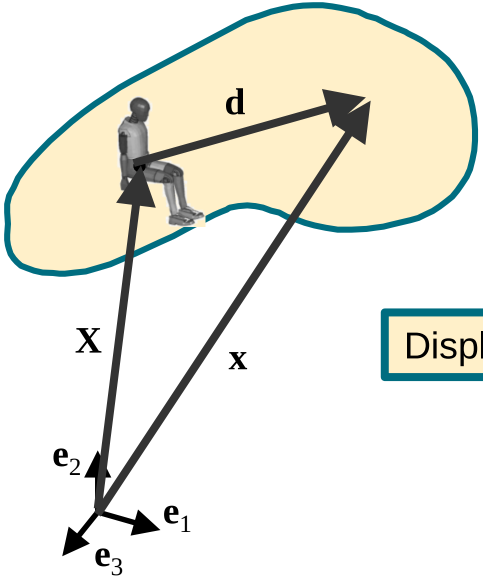
- Mesh-adaptivity (re-meshing)
- ALE approach



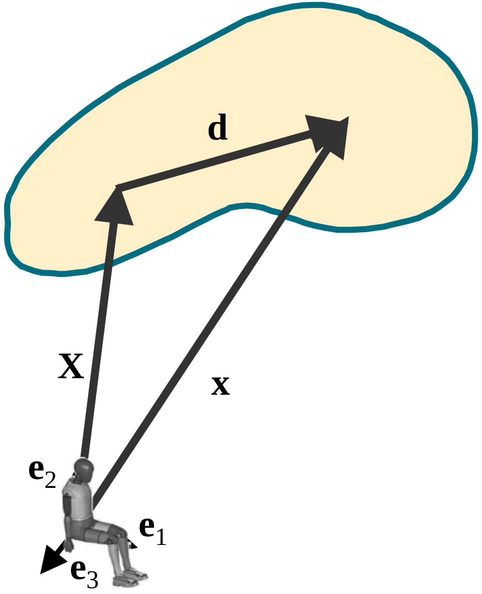
Arbitrary Lagrangian Eulerian (ALE) – continuum mechanics

- Lagrangian description (material description)
  - Observer follows motion of a material point
  - observes the changes of the variables attached to this point

- Eulerian description (spatial description)
  - Observer is fixed in space
  - observes the variables attached to the material points as they pass by



Displacement field: $\mathbf{d} = \mathbf{X} - \mathbf{x}$



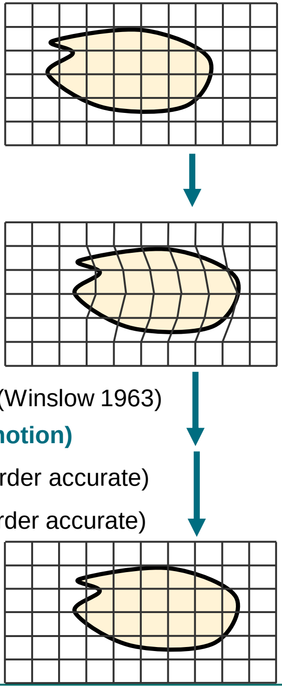
Arbitrary Lagrangian Eulerian (ALE) – advection

- ALE approach uses both Larangian and Eulerian description to complete an time increment
- Illustration of an increment

Initial geometry

Material

void



Lagrangian step

Deformed geometry

Smoothing

- Simple average method
- Equipotential smoothing (Winslow 1963)

Advection step (Eulerian motion)

- Donor cell scheme (1st order accurate)
- Van Leer scheme (2nd order accurate)

Deformed geometry with smoothed mesh

- the advection step
  - Relative motion between material and mesh
  - Leads to more complex evolution of variables

Local time derivative of history variable w.r.t. the Lagrangian reference system in which the mesh follows the material

material velocity

$$\frac{d\phi}{dt} = \frac{\partial \phi}{\partial t} + \nabla \phi \cdot (\mathbf{v} - \dot{\mathbf{x}})$$

mesh velocity

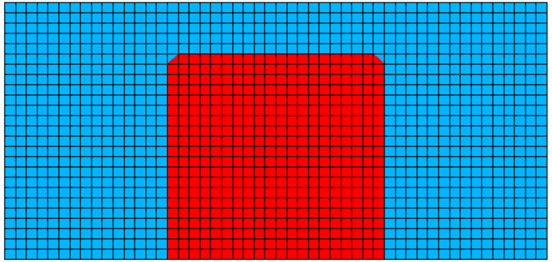
Total material time derivative of a history variable in the ALE reference system

Arbitrary Lagrangian Eulerian (ALE) – multi material ALE

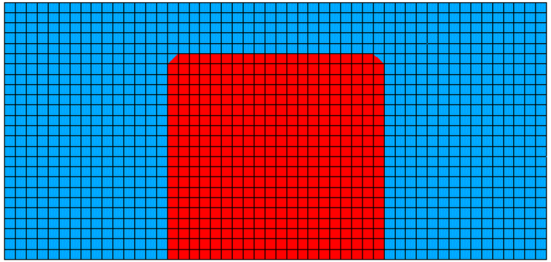
- Any material can flow through a defined domain
- The domain may be fixed in space (Eulerian) or may move arbitrarily (ALE)
- Interfaces between different materials will be traced and reconstructed
- Stresses can be iterated on element level (if bulk moduli are different)
- Multi-material Euler and Multi-Material ALE
  - Material can flow through
    - Fixed mesh (Eulerian)
    - Moveable/deformable mesh (ALE)
  - Flow is subjected stability constraint, i. e.

$$\Delta t_{cr} \approx \min_{nel}\left[\frac{\Delta x^e}{c}, \frac{2\Delta x^e}{v^e}\right]$$

  - Material-interface re-construction based on the computed volume fraction needed



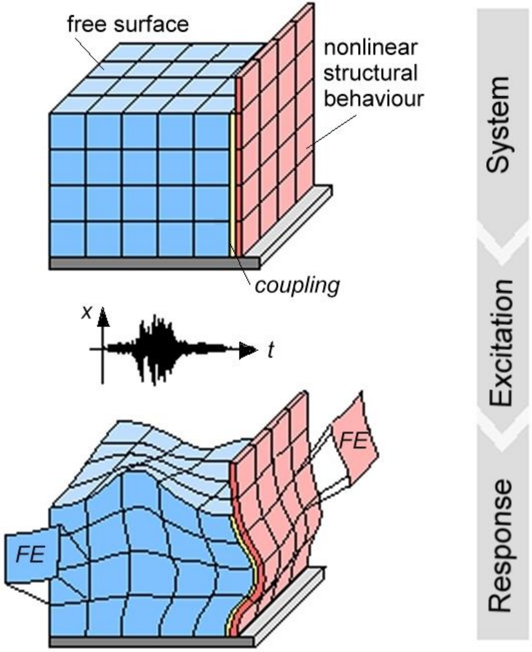
Multi-material Euler



Multi-material ALE

Arbitrary Lagrangian Eulerian (ALE) – What can be done with this?

- It is advantageous to use ALE for modeling
  - gases
  - fluids
  - massive/bulky solid materials (with large deformations)
- Often these parts are contained in or are constrained by other parts. In many cases it might also be advantageous to model these structures Lagrangian.
- Interaction between Eulerian/ALE- and Lagrangian parts (FSI)
- Applied to airbags
  - fabric and structure → Lagrangian finite elements
  - inflator gas and surrounding air → ALE



(c) University of Stuttgart

Arbitrary Lagrangian Eulerian (ALE) – `*AIRBAG_ALE`

CARD1	**SID**	**SIDTYP**					**MWD**	**SPSF**
CARD2	**ATMOST**	**ATMOSP**		**GC**	**CC**	**TNKVOL**	**TNKFINP**	
CARD3	**NQUAD**	**CTYPE**	**PFAC**	**FRIC**	**FRCMIN**	**NORMTYP**	**ILEAK**	**PLEAK**
CARD4	**IVSETID**	**IVTYPE**	**IBLOCK**	**VNTCOF**				

- **core cards**

`SID`	… Set ID of airbag definition (Lagrangian elements)
`SIDTYP`	… Type of Set (segment or part set)
`MWD`	… mass weighted damping factor → used after switch to UP
`SPSF`	… stagnation pressure scale factor → used after switch to UP
`ATMOST`	… atmospheric ambient temperature
`ATMOSP`	… atmospheric ambient pressure
`GC`	… universal molar gas constant
`CC`	… conversion constant
`TNKVOL`	… tank volume from inflator tank test
`TNKFINP`	… tank final pressure

Arbitrary Lagrangian Eulerian (ALE) – *AIRBAG_ALE

CARD1	**SID**	**SIDTYP**					**MWD**	**SPSF**
CARD2	**ATMOST**	**ATMOSP**		**GC**	**CC**	**TNKVOL**	**TNKFINP**	
CARD3	**NQUAD**	**CTYPE**	**PFAC**	**FRIC**	**FRCMIN**	**NORMTYP**	**ILEAK**	**PLEAK**
CARD4	**IVSETID**	**IVTYPE**	**IBLOCK**	**VNTCOF**				

- **coupling card**

`NQUAD`	… Number of (quadrature) coupling points for coupling Lagrangian slave parts to ALE master solid parts
`CTYPE`	… coupling type (EQ.4 or EQ.6) → see `*CONSTRAINT_LAGRANGE_IN_SOLID`
`PFAC`	… penalty scale factor for scaling the estimated stiffness of the interacting (coupling) system
`FRIC`	… coupling coefficient of friction
`FRCMIN`	… Minimum fluid volume fraction in an ALE element to activate coupling.
`NORMTYP`	… Penalty coupling spring direction. Normal vectors are EQ.0: interpolated from nodal normal (default) EQ.1: interpolated from segment normals.
`ILEAK`	… Leakage control flag.
`PLEAK`	… Leakage control penalty factor (default = 0.1)

Arbitrary Lagrangian Eulerian (ALE) – *AIRBAG_ALE

CARD1	**SID**	**SIDTYP**					**MWD**	**SPSF**
CARD2	**ATMOST**	**ATMOSP**		**GC**	**CC**	**TNKVOL**	**TNKFINP**	
CARD3	**NQUAD**	**CTYPE**	**PFAC**	**FRIC**	**FRCMIN**	**NORMTYP**	**ILEAK**	**PLEAK**
CARD4	**IVSETID**	**IVTYPE**	**IBLOCK**	**VNTCOF**				

- **venting hole card**

`IVSETID` … Set ID defining the venting hole surface(s)

`IVTYPE` … Set type of IVSETID:

  - EQ.0: Part Set (default).
  - EQ.1: Part ID.
  - EQ.2: Segment Set.

`IBLOCK` … Flag for considering blockage effects for porosity and vents

  - EQ.0: no (blockage is NOT considered, default).
  - EQ.1: yes (blockage is considered).

`VNTCOF` … Vent Coefficient for scaling the flow

Arbitrary Lagrangian Eulerian (ALE) – `*AIRBAG_ALE`

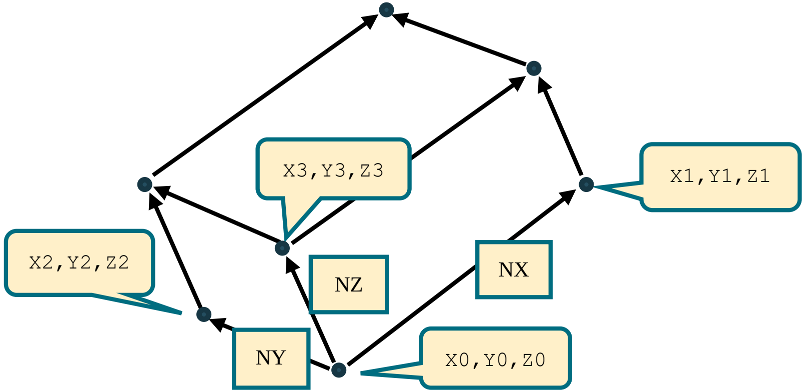
CARD5	NX/IDA	NY/IDG	NZ	MOVERN	ZOOM			
CARD5.1	X0	Y0	Z0	X1	Y1	Z1		
CARD5.2	X2	Y2	Z2	X3	Y3	Z3		

only if `NZ!=0`

- **geometry cards**

`NX/IDA`	… number of elements in x-direction or part ID of air
`NY/IDG`	… number of elements in x-direction or part ID of gas
`NZ`	… number of elements in x-direction EQ.0: `IDA` and `IDG` are used
`MOVERN`	… ALE mesh automatic motion option GT.0: Node group ID for `*ALE_REFERENCE_SYSTEM_NODE`
`ZOOM`	… ALE mesh automatic expansion option EQ.1: Expand/contract ALE mesh by keeping all airbag parts contained within the ALE mesh

`XYZ,1-3` … Define Euler-domain that will be meshed by `NX*NY*NZ` hexahedron elements



Arbitrary Lagrangian Eulerian (ALE) – `*AIRBAG_ALE`

CARD6	**SWTIME**		**HG**	**NAIR**	**NGAS**	**NORIF**	**LCVEL**	**LCT**
CARD7				**MWAIR**	**INITM**	**AIRA**	**AIRB**	**AIRC**
CARD8	**LCMF**			**MWGAS**		**GASA**	**GASB**	**GASC**
CARD9	**NODEID**	**VECID**	**ORIFARE**					

repeat `NGAS` times

repeat `NORIF` times

- **gas and air cards**

`SWTIME` … time to switch to UP

`HG` … Hourglass control for ALE fluid mesh(es)

`NAIR` … number of air components

`NGAS` … number of inflator gas components

`NORIF` … Number of point sources or orifices

`LCVEL` … Load curve ID for inlet velocity

`LCT` … Load curve ID for inlet gas temperature

`MWAIR` … Molecular weight of air component

`INITM` … Initial Mass Fraction of air component(s)

`AIRA, AIRB, AIRC` … coefficient for temperature dependent heat capacities (c_p, c_v) of air

`LCMF` … Load curve ID for mass flow rate

`MWGAS` … Molecular weight of gas component

`GASA, GASB, GASC` … coefficient for temperature dependent heat capacities (c_p, c_v) of gas component

Arbitrary Lagrangian Eulerian (ALE) – *AIRBAG_ALE

CARD6	**SWTIME**		**HG**	**NAIR**	**NGAS**	**NORIF**	**LCVEL**	**LCT**	
CARD7				**MWAIR**	**INITM**	**AIRA**	**AIRB**	**AIRC**	
CARD8	**LCMF**			**MWGAS**		**GASA**	**GASB**	**GASC**	repeat NGAS times
CARD9	**NODEID**	**VECID**	**ORIFARE**						repeat NORIF times

- **orifice/ point source definition cards**

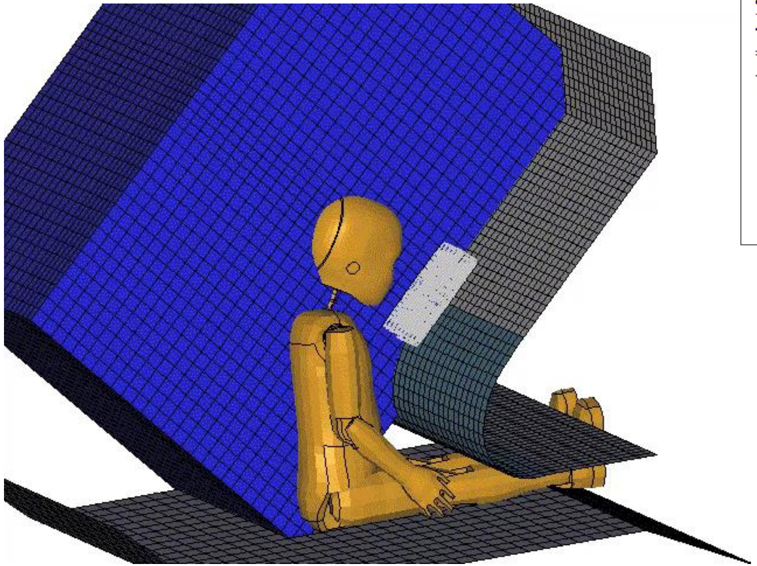
`NODEID` … Node ID of point source

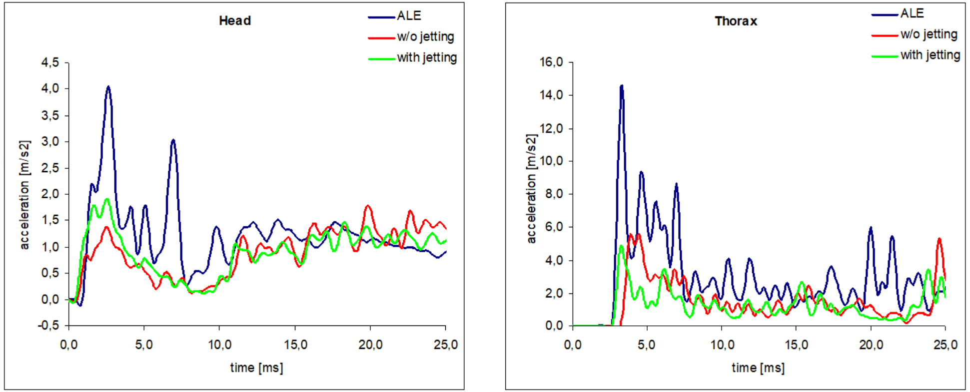
`VECID` … Vector ID defining the direction of flow at the point source

`ORIFARE` … The orifice area at the point source

Arbitrary Lagrangian Eulerian (ALE) – Application

- out-of-position simulation
- 3 year old child

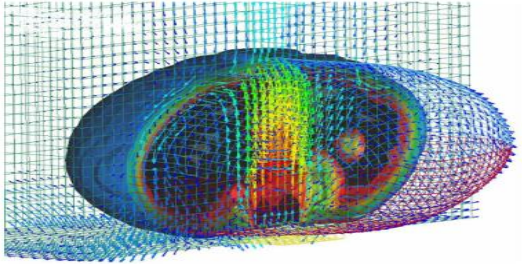




- local gas flow effects are simulated
- more realistic behavior while unfolding
- higher accelerations predicted during deployment phase

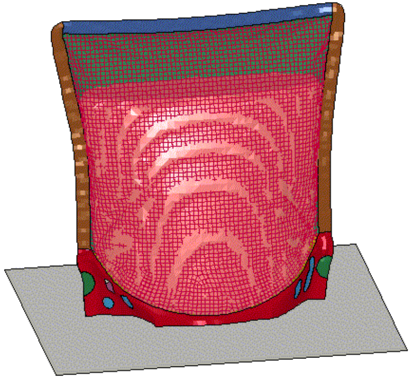
Arbitrary Lagrangian Eulerian (ALE) – pros and cons

- pros
  - actual fluid-structure interaction including simulation of gas flow effects
  - enhanced post-processing capabilities

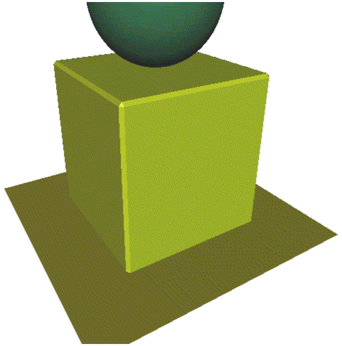


- cons
  - rather complicated model set up
  - avoiding unwanted leakage through Lagrangian boundaries requires high number of coupling points
  - computationally quite expensive

- applications beyond airbag modeling
  - bag partially filled with fluid
  - bag in Lagrangian shell elements
  - fluid in ALE



  - ball impacting foam block
  - ball in Lagrangian solids
  - block in ALE

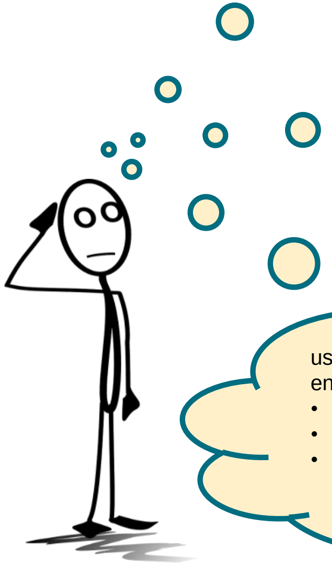


Outline

- Introduction
- Uniform Pressure Models (UP)
- Arbitrary Lagrangian Eulerian (ALE)
- Corpuscular Particle Method (CPM)

Corpuscular Particle Method (CPM) – Motivation

- “users wish”:
  less computational effort but similar accuracy compared to ALE
- “developers idea”:
  particle approach based on the kinetic molecular theory, where the gas is represented by molecules in constant, rapid, random motion
  - particle approach allows good scalability with MPP
  - no discretization of the surrounding gas necessary
  - the number of molecules inside an airbag volume is typically $10^{23} – 10^{24}$
    → some kind of simplification is mandatory



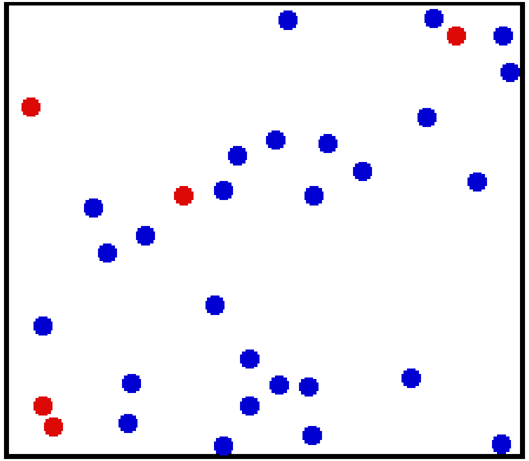
consider the effect of transient gas dynamics and thermodynamics by using a particle to represent a set of air or gas molecules

consider the effect of transient gas dynamics and thermodynamics by using a particle to represent a set of air or gas molecules

use a set of individual particles to represent the entire air or gas molecule in the space of interest
- particles interact with each other via contact
- particles interact with environment via contact
- contact events to environment result into pressure

Kinetic Molecular Theory

- Based on the following assumptions:
  - The average distance between molecules is large compared to their size
  - Molecule-molecule and molecule-structure collisions are perfectly elastic
  - Molecules obey Newton‘s laws of motion
  - Molecules are in random motion



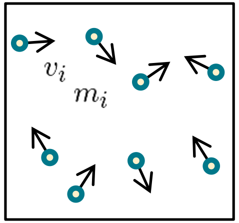
Kinetic Molecular Theory

- The specific internal energy of a gas can be divided into translational kinetic energy, vibration and spin. The translational kinetic energy is the component that produces pressure.
- The ideal gas law and the kinetic molecular theory predict the same pressure at thermal equilibrium.
- The kinetic molecular theory matches the ideal gas law for the change in internal energy due to adiabatic expansion

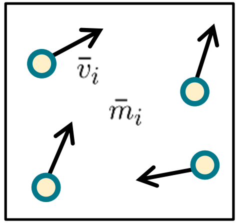
- Since the pressure is a function of the specific translational energy only, a few large molecules with total mass m_{tot} will produce the same pressure as many small molecules with the same total mass, as long as the following conditions hold:
    - Root mean square velocities v_{rms} are the same
    - Ratio of translational kinetic energy and total internal energy (ξ) are same
    - The Maxwell-Boltzmann velocity distribution is maintained
- This is of fundamental importance for the corpuscular method in LS-DYNA!

Corpuscular Particle Method

- It is not possible to model every single molecule inside the airbag
  → Continuum treatment of the gas
- System reduced from many molecules to a “few” particles in such a way that the translational kinetic energy remains constant

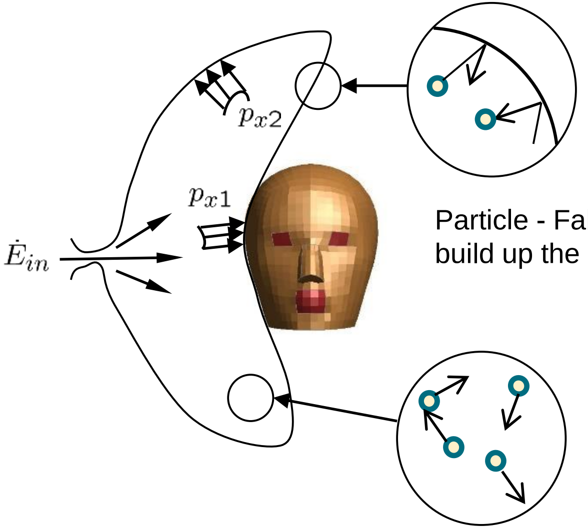


$$E_k = \frac{1}{2} \sum_{i=1}^{N_m} m_i \, v_i^2$$



$$\bar{E}_k = \frac{1}{2} \sum_{i=1}^{N_p} \bar{m}_i \, \bar{v}_i^2$$

$$\boxed{\bar{E}_k = E_k \quad \Rightarrow \quad \bar{p} = p}$$



Particle - Fabric contacts build up the pressure

Particle - Particle contacts are necessary for a realistic dynamical behavior of the gas

Corpuscular Particle Method – theory summary

- The corpuscular method in LS-DYNA is based on the kinetic molecular theory.
- The system is reduced from many molecules to a “few” particles with each particle representing many molecules.
- The particles are spherical in shape for efficient treatment of contact.
- For each particle, a balance exists between the translational kinetic energy and the vibration/ spin energy. This balance can be determined from the heat capacities (or from ξ).
- Since many molecules are represented by a single particle, it leads to dispersion and the generation of noise in the pressure signal. The noise is reduced by smoothing out the pressure applied internally.

Corpuscular Particle Method – `*AIRBAG_PARTICLE`

CARD1	SID1	STYPE1	SID2	STYPE2	BLOCK	NPDATA	FRIC	IRPD
CARD2	NP	UNIT	VISFLG	TATM	PATM	NVENT	TEND	TSW
CARD4	IAIR	NGAS	NORIF	NID1	NID2	NID3	CHM	CD_EXT

`SID1`	… Set defining the complete airbag
`STYPE1`	… Set type
`SID2`	… Set defining the internal parts of the airbag
`STYPE2`	… Set type
`BLOCK`	… Blocking (reduced leakage due to contact)

- EQ.00
- EQ.01
- EQ.10
- EQ.11

The 1‘s digit controls the treatment of **leakage**:
0: Always consider porosity leakage without considering blockage due to contact.
1: Check if airbag node is in contact or not. If yes, 1/4 (quad) or 1/3 (tria) of the segment surface will not have porosity leakage due to contact.
The 10‘s digit controls the treatment of particles that **escape due to deleted elements**:
0: Active particle. Particles will be put back into the bag
1: Leaked through vent

`NPDATA`	… Number of Parts or Part set Data
`FRIC`	… Friction factor for particles
`IRPD`	… Dynamic scaling of particle radius

- Eq. 0: Off
- Eq. 1: On

Corpuscular Particle Method – `*AIRBAG_PARTICLE`

CARD1	**SID1**	**STYPE1**	**SID2**	**STYPE2**	**BLOCK**	**NPDATA**	**FRIC**	**IRPD**
CARD2	**NP**	**UNIT**	**VISFLG**	**TATM**	**PATM**	**NVENT**	**TEND**	**TSW**
CARD4	**IAIR**	**NGAS**	**NORIF**	**NID1**	**NID2**	**NID3**	**CHM**	**CD_EXT**

`NP` … Number of particles (default 200,000)

`UNIT` … Unit system

- Eq. 0: kg-mm-ms-K
- Eq. 1: kg-m-s-K (SI-units)
- Eq. 2: ton-mm-s-K
- Eq. 3: User defined units → leads to additional CARD3

`VISFLG` … Visibility of particles

- Eq. 0: No
- Eq. 1: Yes
- Eq. 2: Yes (reduced particle database)
- Eq. 3: Yes (summary only)

`TATM` … Atmospheric temperature (default 293 K)

`PATM` … Atmospheric pressure (default 101.3 kPa)

`NVENT` … Number of vent hole definitions

`TEND` … Time when all the particles have entered the bag (default 1.0E10)

`TSW` … Time for switch to control volume formulation (default 1.0E10)

Corpuscular Particle Method – `*AIRBAG_PARTICLE`

CARD1	**SID1**	**STYPE1**	**SID2**	**STYPE2**	**BLOCK**	**NPDATA**	**FRIC**	**IRPD**
CARD2	**NP**	**UNIT**	**VISFLG**	**TATM**	**PATM**	**NVENT**	**TEND**	**TSW**
CARD4	**IAIR**	**NGAS**	**NORIF**	**NID1**	**NID2**	**NID3**	**CHM**	**CD_EXT**

`IAIR` … Initial air inside the bag considered

- Eq.-1: Yes (using UP) It intakes ambient air into bag when PATM is greater than bag pressure
- Eq. 0: No
- Eq. 1: Yes (using UP)
- Eq. 2: Yes (using particles)
- Eq. 4: Yes (using particles – gas front tracking algorithm!)

`NGAS` … Number of gas components

`NORIF` … Number of orifices

`NID1-NID3` … Nodes defining a moving coordinate system for the direction of flow through the gas inlet nozzles.

`CHM` … Chamber ID used in `*DEFINE_CPM_CHAMBER`

`CD_EXT` … Drag coefficient for external air

Corpuscular Particle Method – *AIRBAG_PARTICLE

- Optional Card3 if UNIT = 3

CARD1	**SID1**	**STYPE1**	**SID2**	**STYPE2**	**BLOCK**	**NPDATA**	**FRIC**	**IRPD**
CARD2	**NP**	**UNIT**	**VISFLG**	**TATM**	**PATM**	**NVENT**	**TEND**	**TSW**
CARD3		**Mass**		**Time**		**Length**		

`Mass, Time, Length` … Conversion factor from current unit to MKS unit.
For example, if the current unit is using kg-mm-ms, the input should be 1.0, 0.001, 0.001

- Optional IAIR card (If `IAIR > 0`)

CARD8	**PAIR**	**TAIR**	**XMAIR**	**AAIR**	**BAIR**	**CAIR**	**NPAIR**	**NP_RELAX**

`Mass, PAIR` … Initial pressure inside the bag (default `PAIR=PATM`)

`TAIR` … Initial temperature inside the bag (default `TAIR=TATM`)

`XMAIR` … Molar mass of air initially inside the bag

`AAIR – CAIR` … Constant, linear and quadratic heat capacities at constant pressure $([J/\mathrm{K\,mol}],\ [J/\mathrm{K}^2\,\mathrm{mol}],\ [J/\mathrm{K}^3\,\mathrm{mol}])$

`NPAIR` … Number of initial air particles (Total no. of particles = `NP + NPAIR`)

`NP_RELAX` … Number of cycles for thermal equilibrium

Corpuscular Particle Method – `*AIRBAG_PARTICLE`

- initial air represented by particles (`IAIR=2/4`)
  - `NP` … Number of gas particles
  - `NP_AIR` … Number of initial air particles

$$\left(\frac{(n_{mole})_{gas}}{\mathtt{NP}}\right) = \left(\frac{(n_{mole})_{air}}{\mathtt{NP\_AIR}}\right)$$

  - LS-DYNA checks for the above condition in recent versions and generates a warning if the condition is not satisfied!
  - Warning in mesXXXX-Files:

```
[…]
*** Warning 41232 (SOL+1232)
     CPM bag #: 1
      mole per particle for inflator gas...........6.359810E-06
      mole per particle for initial air............ 1.9057E-05
      number of initial air particle............... 1000
      number of initial air particle suggested..... 2996
[…]
```

Corpuscular Particle Method – `*AIRBAG_PARTICLE`

- `NPDATA` cards (If `NPDATA>0`)

CARD6	**SIDH**	**STYPEH**	**H**	**PFRIC**	**SDFBLK**	**KP**	**INIP**	

repeat `NPDATA` times

`SIDH` … Set defining heat convection

`STYPEH` … Set type

- Eq. 0: Part
- Eq. 1: Part Set

`H` … Heat convection coefficient (default 0.0) [W/(K m2)]

`PFRIC` … Friction factor (Default is `FRIC` from 1st card, 7th field)

`SDFBLK` … Scale down factor for blockage factor (Default=1, no scale down).

`KP` … Effective Convection Heat Transfer Coefficient. If the thermal conductivity, `KP`, is given, then the effective convection heat transfer coefficient is given by:

$$H_{\text{eff}} = \left(\frac{1.0}{\texttt{HCONV}} + \frac{\text{shell thickness}}{\texttt{KP}} \right)^{-1}$$

if `KP` is not given, H_{eff} defaults to `HCONV`.

`INIP` … Place initial air particles on surface.

- Eq. 0 yes (default)
- Eq. 1 no

This feature excludes surfaces from initial particle placement. It is useful for preventing particles from being trapped between adjacent fabric layers.

Corpuscular Particle Method – `*AIRBAG_PARTICLE`

- `NVENT` cards (If `NVENT>0`)

CARD7	**SID3**	**STYPE3**	**C23**	**LCTC23**	**LCPC23**	**ENH_V**	**PPOP**	

repeat `NVENT` times

`SID3` … Set defining the vent holes

`STYPE3` … Set type

- Eq. 0: Part
- Eq. 1: Part Set
- Eq. 2: Part set (treat all parts as one vent)
  → Important in combination with `ENH_V`!

`C23` … Vent hole coefficient (parameter for Wang-Nefske leakage) (default 1.0)

`LCTC23` … Load curve defining vent hole coefficient as a function of time

`LCPC23` … Load curve defining vent hole coefficient as a function of pressure

`ENH_V` … Enhanced venting option

- Eq. 0: Off (default)
- Eq. 1: On
- Eq. 2: Two way flow from internal vent; treated as hole for external vent.

When enhanced venting is on, the vent hole's equivalent radius R_{eq} will calculated. Particles within R_{eq} on the high pressure side from the vent hole geometry center will be moved toward the hole. This will increase the collision frequency near the vent to detect small structural features and produce better flow through the vent hole.

`PPOP` … Pressure difference between interior and ambient pressure to open the vent hole

Corpuscular Particle Method – `*AIRBAG_PARTICLE`

- `NGAS` cards (i=1,2,…,n)

CARD10	**LCM_i**	**LCT_i**	**XM_i**	**A_i**	**B_i**	**C_i**	**$INFG_i$**	

repeat `NGAS` times

`LCMi`	…	Mass flow rate curve for component i
`LCTi`	…	Temperature load curve for component i
`XMi`	…	Molar mass of component i
`Ai – Ci`	…	Constant, linear and quadratic heat capacities at constant pressure ($J/mol\ K$)
`INFGi`	…	Inflator ID that this gas component belongs to

Corpuscular Particle Method – `*AIRBAG_PARTICLE`

- `NORIF` cards (i=1,2,…,n)

CARD11	**NID$_i$**	**AN$_i$**	**VD$_i$**	**CA$_i$**	**INFO$_i$**	**IMOM**	**IANG**	**CHM_ID**

repeat `NORIF` times

`NIDi` … Node ID/ Shell ID defining location of nozzle i
Node and shell based nozzle should not be used in the same airbag definition. For shell based definition, nozzle direction can be defined by shell normal/reversed normal.

`ANi` … Area of nozzle i

`VDi` … ID of vector defining initial direction of gas inflow at nozzle i

- `VDi` > 0: Vector ID
- `VDi` < 0:
  - Eq. -1: direction of gas inflow is using shell normal
  - Eq. -2: direction of gas inflow is reversed shell normal

`CAi` … Cone angle in radians (jet angle, default 30°) (only used if IANG=1)

`INFOi` … Inflator ID that orifice i belongs to

`IMOM` … Inflator reaction forces

- Eq. 0: Off
- Eq. 1: On

`IANG` … Activation of cone angle to be used for friction calibration (not normally used; eliminates thermal energy of particles from inflator)

- Eq. 0: Off
- Eq. 1: On

`CHM_ID` … Chamber ID where the inflator node resides

Corpuscular Particle Method – *AIRBAG_PARTICLE

- Typical *AIRBAG_PARTICLE keyword

```
*AIRBAG_PARTICLE_ID
$#        id                                                               title
         1 CPMBag
$#    sid1    stype1      sid2    stype2     block    npdata      fric      irdp
       601         1         1         1         1
$#        np      unit    visflg      tatm      patm     nvent      tend       tsw
&np                           1     296.0       0.0         4       0.0       0.0
$#      iair      ngas     norif      nid1      nid2      nid3       chm    cd_ext
         1         1         2
$#      sid3    stype3       c23    lctc23    lcpc23     enh_v      ppop
       606                 1.0                             1
$#      sid3    stype3       c23    lctc23    lcpc23     enh_v      ppop
       624                 1.0                             1
$#      sid3    stype3       c23    lctc23    lcpc23     enh_v      ppop
       626                 1.0       650                   1
$#      sid3    stype3       c23    lctc23    lcpc23     enh_v      ppop
       627                 1.0       650                   1
$#      pair      tair     xmair      aair      bair      cair    np_air  np_relax
1.01325E-4     296.0     0.028 26.299999    0.0077-1.4000E-6
$#      lcmi      lcti       xmi        ai        bi        ci     infgi
       607       608     0.004 20.790001                             0
$#      nidi       ani       vdi       cai     infoi      imom      iang    chm_id
    615001                 601       0.0         1                           996
$#      nidi       ani       vdi       cai     infoi      imom      iang    chm_id
    615002                 601       0.0         1                           996
```

- several additional options are available via further keywords
  - `*DEFINE_CPM_GAS_PROPERTIES`
  - `*DEFINE_CPM_VENT`
  - `*DEFINE_CPM_CHAMBERS`
- to be discussed in the advanced seminar!

Corpuscular Particle Method – summary

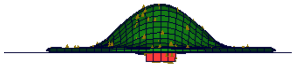
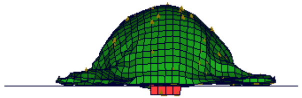
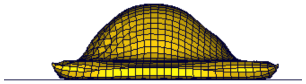
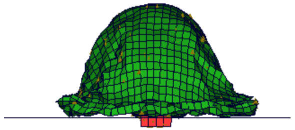
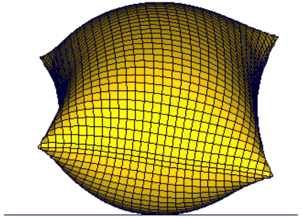
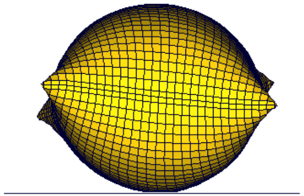
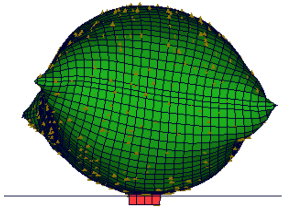
- Advantages
  - Simple and numerically robust
  - Relatively easy to convert from `*AIRBAG_HYBRID` cards
  - Good accuracy also in Out-of-position (Oop) simulations
  - Widely used and preferred over other airbag formulations in crash simulations

- Drawbacks
  - A certain level of noise exists in the pressure signal
  - The method cannot describe the actual flow field accurately

- Efforts to incorporate more options in progress ……

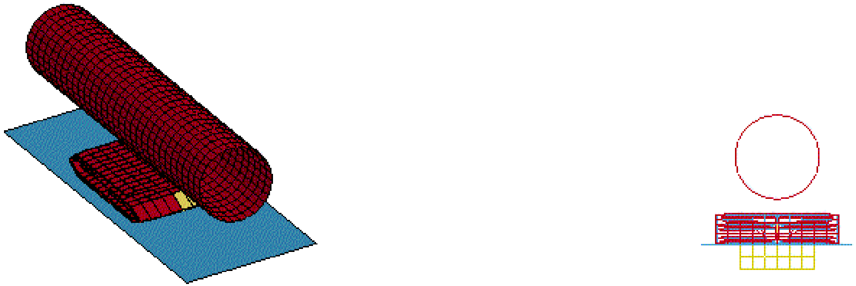
Corpuscular Particle Method – examples

- simple unfolded airbag

	*AIRBAG_HYBRID	*AIRBAG_HYBRID_JETTING	*AIRBAG_PARTICLE (NP=10.000)
t=0ms			
t=1ms			
t=2ms			
t=3ms			
t=25ms			

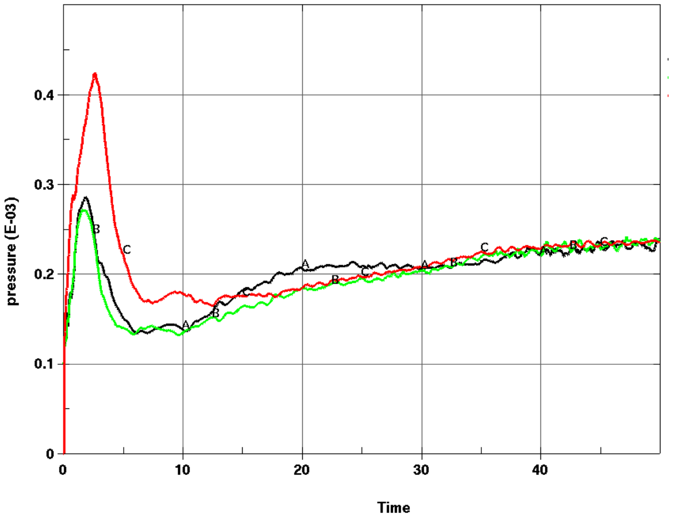
Corpuscular Particle Method – examples

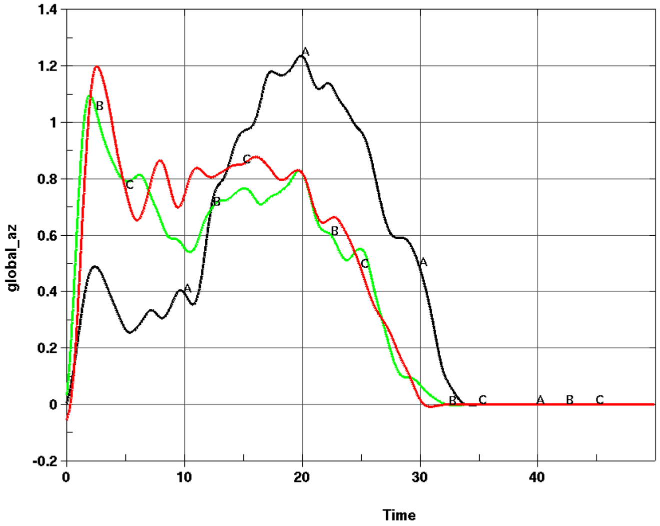
- impactor test with folded airbag



Corpuscular Particle Method – examples

- impactor test with folded airbag
  - `*AIRBAG_HYBRID`
  - `*AIRBAG_HYBRID_JETTING`
  - `*AIRBAG_PARTICLE`


Airbag pressure


Impactor acceleration

Conclusions

UPM

+ Numerically cheap and robust
+ Airbag definition quite simple
– Fluid represented by pressure boundary condition
– local effects missing
– non-physical parameters
– inaccuracy in deployment phase

ALE

+ actual fluid-structure-interaction
+ better results during airbag deployment
– rather difficult model set-up
– numerically expensive
– problems with unwanted leakage

CPM

+ model set-up similar to UPM
+ good accuracy also during deployment
+ good scalability for MPP
– pressure noise
– numerical cost increase with number of particles

Thank you for your attention!