Numerical simulation of membrane deformation under high-speed load

QIPA 2018

V. Aksenov, A. Vasyukov, K. Beklemysheva

December 4, 2018

Moscow Institute of Physics and Technology

Table of contents

Introduction

Problem overview

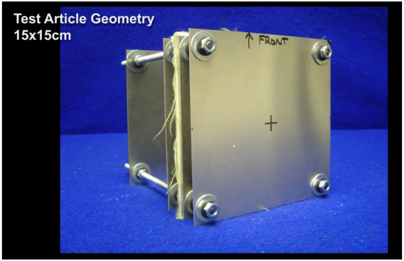


Figure 1: Protection cover example: two metal layers and textile layer

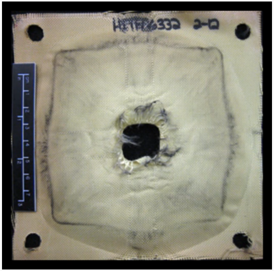


Figure 2: Textile layer after the impact

Subject area: meteorite protection for satellites

Scope of current project:

- Solving direct problem — membrane under high-speed load
- Modeling thin membrane with arbitrary rheology
- Data generation for inverse problem

Mathematical model

Requirements and assumptions

- Thin membrane — 2D object in 3D space
- Arbitrary load profile, both in time and space
- Large displacements; membrane moves as a whole

Numerical method

Finite element method (FEM)

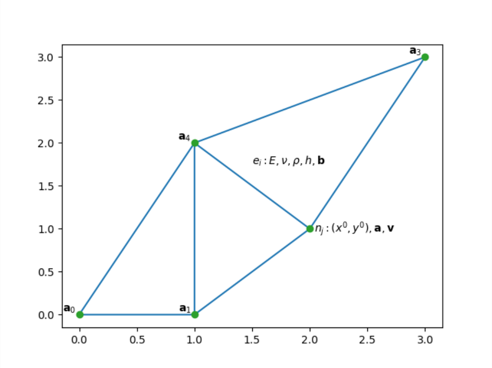


Figure 3: Domain division example; sample element e_i and node n_j with associated values

1. Domain division into elements (e.g. triangles)
2. Choice of shape functions $\mathbf{u} = \mathbf{N}\mathbf{a}$
3. Reduction to system of algebraic (static) or ordinary differential (dynamic) equations

Limitations

- Only distributed force over an element is considered
- Free boundary — no constraints or forces on the boundary
- Isotropic material
- Homogeneous material in element — any heterogeneity is accounted for at domain division

Shape functions N_i^e

Displacements in an element are approximated via *shape functions* N_i:

$$\mathbf{u}(x, y, z) = \sum_{i=1}^{3} \mathbf{N_i a_i} = \mathbf{Na} \tag{1.1}$$

where $\mathbf{N} = \begin{bmatrix} \mathbf{N}_1 & \mathbf{N}_2 & \mathbf{N}_3 \end{bmatrix}$, $\mathbf{a} = \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{bmatrix}$ — nodal displacements.

Assumption

Displacement doesn't depend on z:

$$\mathbf{N}_i = (\alpha_i + \beta_i x + \gamma_i y)\mathbf{E} \tag{1.2}$$

as in FEM for 2*D* problems[2]

Strains

Writing down strains:

$$\varepsilon = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial v}{\partial y} & \frac{\partial w}{\partial z} & \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} & \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} & \frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \end{bmatrix}^T \tag{1.3}$$

$$\mathsf{S} = \begin{bmatrix} \mathsf{S}_1 \\ \mathsf{S}_2 \end{bmatrix}$$

$$\mathsf{S}_1 = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \end{bmatrix}, \quad \mathsf{S}_2 = \begin{bmatrix} \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \end{bmatrix} \tag{1.4}$$

Strains

$$\varepsilon = \mathsf{Su} = \mathsf{SNa} = \mathsf{Ba} \tag{1.5}$$

where B is the *gradient matrix*

Stresses

Writing down stresses:

$$\sigma = \begin{bmatrix} \sigma_{xx} & \sigma_{yy} & \sigma_{zz} & \tau_{xy} & \tau_{yz} & \tau_{xz} \end{bmatrix}^T \tag{1.6}$$

Assuming isotropic material, let

$$\mathsf{D}_1 = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu \\ \nu & 1-\nu & \nu \\ \nu & \nu & 1-\nu \end{bmatrix} \tag{1.7}$$

$$\mathsf{D}_2 = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} \frac{1-2\nu}{2} & 0 & 0 \\ 0 & \frac{1-2\nu}{2} & 0 \\ 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix}$$

Stresses

$$\sigma = \mathsf{DBa}, \quad \mathsf{D} = \begin{bmatrix} \mathsf{D}_1 & \mathbf{0} \\ \mathbf{0} & \mathsf{D}_2 \end{bmatrix} \tag{1.8}$$

Let's introduce fictive nodal forces $\mathbf{q}_e$ which are statically equivalent to actual forces. Considering virtual displacements $\delta\mathbf{u}, \delta\varepsilon$ virtual work per unit volume is

$$\delta W = \delta\varepsilon^{\mathsf{T}}\sigma - \delta\mathbf{u}^{\mathsf{t}}\overline{\mathbf{b}} \tag{1.9}$$

Substituting from (1.1), (1.8), (1.5) :

$$\delta\mathbf{u} = \mathbf{N}\delta\mathbf{a}, \quad \delta\varepsilon = \mathbf{B}\delta\mathbf{a} \tag{1.10}$$

we get

Principle of virtual work

$$\delta\mathbf{a}^T\left(\int_{V_e}\mathbf{B}^{\mathsf{T}}\mathbf{D}\mathbf{B}\mathbf{a}dV - \int_{V_e}\mathbf{N}^{\mathsf{T}}\overline{\mathbf{b}}dV\right) = \delta\mathbf{a}^T\mathbf{q}_e \tag{1.11}$$

Stiffness, mass and load matrices $\mathbf{K}_e, \mathbf{M}_e, \mathbf{f}_e$

Dynamic state can be considered as static with added inertial force $-\rho\ddot{\mathbf{u}}$ (d'Alembert's principle). Substituting

$$\bar{\mathbf{b}} = \mathbf{b} - \rho\ddot{\mathbf{u}}, \tag{1.12}$$

where $\mathbf{b}$ is actual distributed force, and assuming $\ddot{\mathbf{u}} = \mathbf{N}\ddot{\mathbf{a}}$ we get

Element matrices

$$\mathbf{M}_e\ddot{\mathbf{a}} + \mathbf{K}_e\mathbf{a} + \mathbf{f}_e = \mathbf{q}_e \tag{1.13}$$

$$\mathbf{K}_e = \int_{V_e} \mathbf{B}^\mathsf{T}\mathbf{D}\mathbf{B}dV - \text{stiffness matrix} \tag{1.14}$$

$$\mathbf{M}_e = \rho\int_{V_e} \mathbf{N}^\mathsf{T}\mathbf{N}dV - \text{mass matrix} \tag{1.15}$$

$$\mathbf{f}_e = -\int_{V_e} \mathbf{N}^\mathsf{T}\mathbf{b}dV_e - \text{load matrix} \tag{1.16}$$

Global matrices K, M, f

Now equations for each element should be assembled to acquire equations for the whole domain. Fictive nodal forces $\mathbf{q} = 0$, so:

Global matrices

$$\mathsf{M}\ddot{\mathsf{a}} + \mathsf{K}\mathsf{a} + \mathsf{f} = 0 \tag{1.17}$$

$$\mathsf{K}_{ij} = \sum_e \mathsf{K}^e_{ij} \tag{1.18}$$

$$\mathsf{M}_{ij} = \sum_e \mathsf{M}^e_{ij} \tag{1.19}$$

$$\mathsf{f}_i = \sum_e \mathsf{f}^e_i \tag{1.20}$$

Here summation is considered over all elements in which i and j are included together, K_{ij} meaning a 3×3 block in global $3N_{nodes} \times 3N_{nodes}$ matrix, corresponding to i-th and j-th nodes.

Time integtation

The problem has been reduced to a 2-nd order system of ODEs

$$\mathbf{M}\ddot{\mathbf{a}} + \mathbf{K}\mathbf{a} + \mathbf{f} = 0$$

We shall integrate it using Newmark β-method:

$$\begin{aligned}
\dot{\bar{\mathbf{a}}}_n &= \dot{\mathbf{a}}_n + \tau(1-\beta_1)\ddot{\mathbf{a}}_n \\
\bar{\mathbf{a}}_n &= \mathbf{a}_n + \tau\dot{\mathbf{a}}_n + \frac{1}{2}\tau^2(1-\beta_2)\ddot{\mathbf{a}}_n \\
\ddot{\mathbf{a}}_{n+1} &= -A^{-1}\left(\mathbf{f}_{n+1} + \mathbf{K}\bar{\mathbf{a}}_n\right), \quad A = \mathbf{M} + \frac{1}{2}\tau^2\beta_2\mathbf{K} \\
\dot{\mathbf{a}}_{n+1} &= \dot{\bar{\mathbf{a}}}_n + \beta_1\tau\ddot{\mathbf{a}}_{n+1} \\
\mathbf{a}_{n+1} &= \bar{\mathbf{a}} + \frac{1}{2}\tau^2\beta_2\ddot{\mathbf{a}}_{n+1}
\end{aligned} \tag{1.21}$$

Method properties

- If $\beta_2 \geq \beta_1 \geq \frac{1}{2}$ the method is stable regardless of timestep τ
- If $\beta_1 = \frac{1}{2}$, the method is second-order accurate [1]

Simulation results

Normal strike

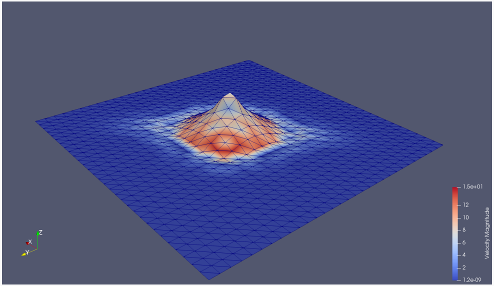
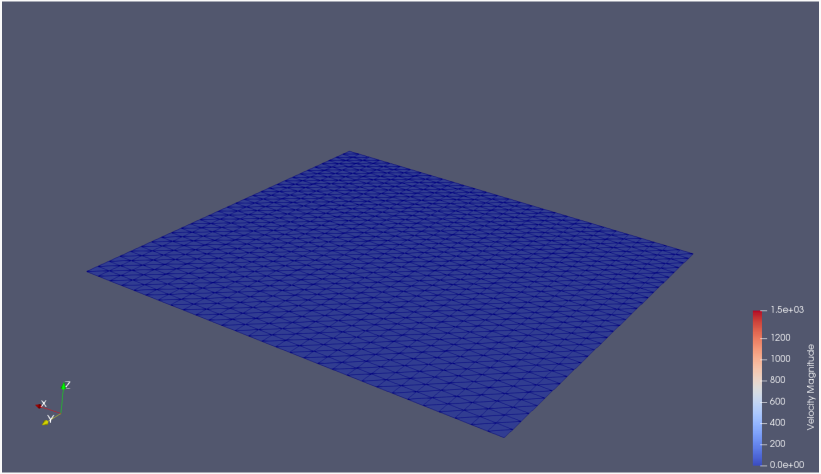


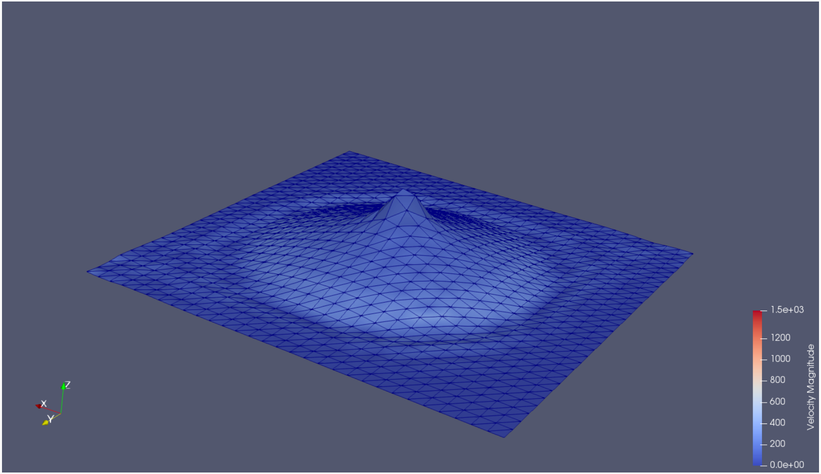
Figure 4: Deformed state at timestep 15τ

Material properties	
E, MPa	1.0
ν	0.45
ρ, $\frac{g}{cm^3}$	0.9
h, mm	1.0
Model parameters	
τ, s	$1.0 \cdot 10^{-4}$
Nodes	31×31
Size, m	1.0×1.0
Pressure, Pa	$2.0 \cdot 10^5$

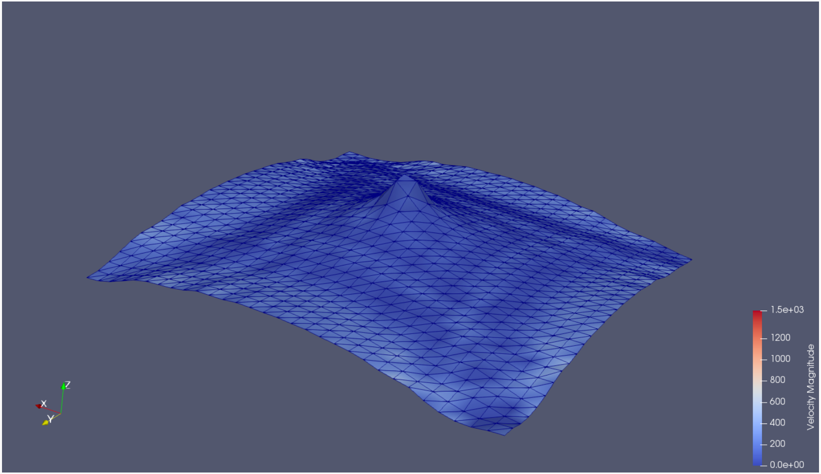
Normal strike



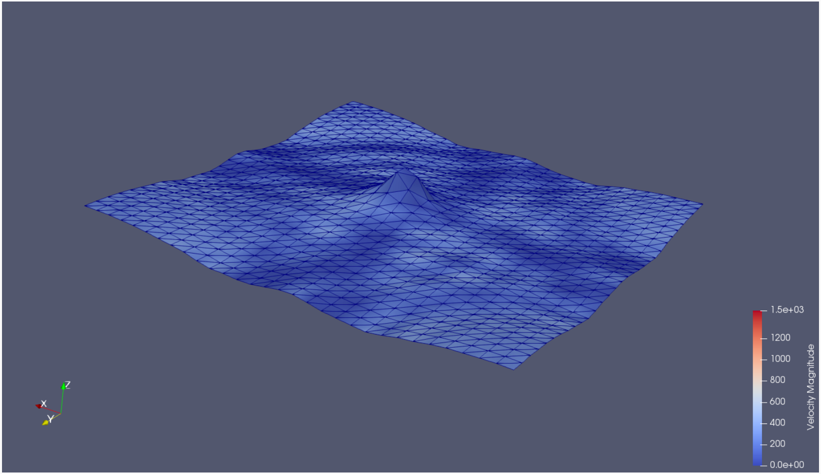
Normal strike



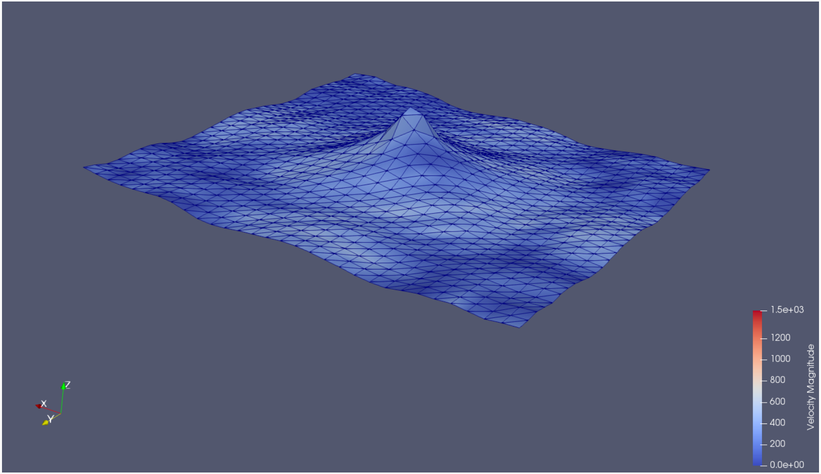
Normal strike



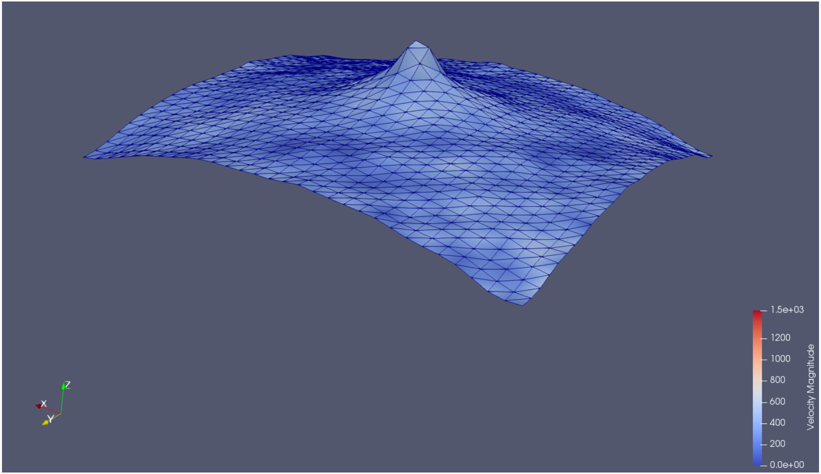
Normal strike



Normal strike



Normal strike



Skew strike

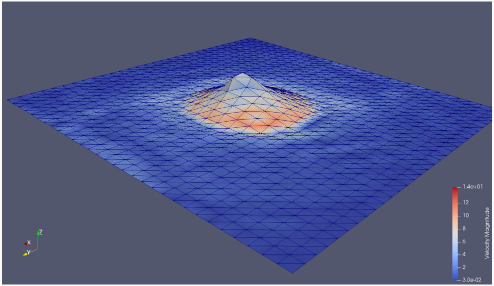
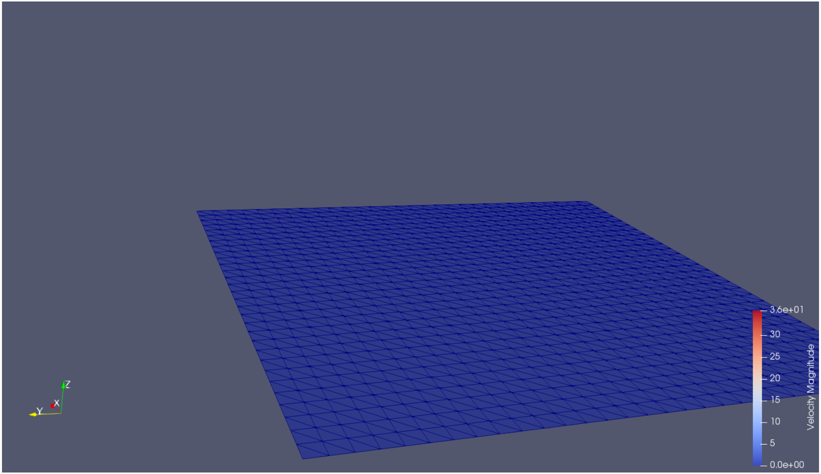


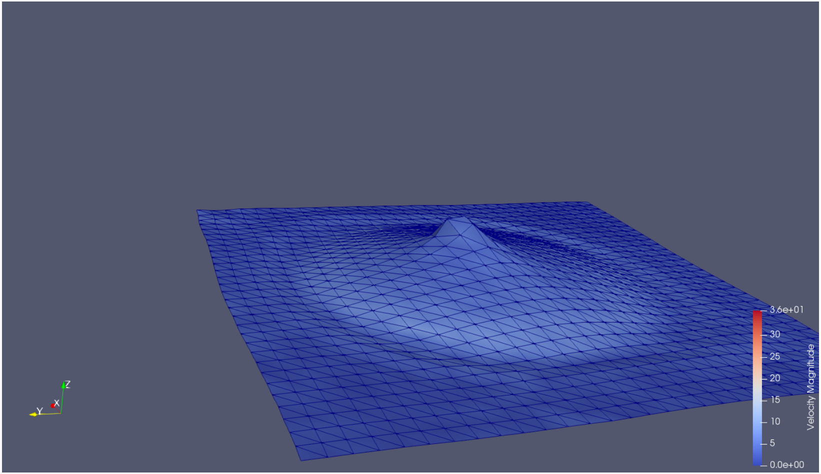
Figure 5: Deformed state at timestep 15τ, strike is 45° to normal

Material properties	
$E,\ MPa$	1.0
ν	0.45
$\rho,\ \frac{g}{cm^3}$	0.9
$h,\ mm$	1.0
Model parameters	
τ, s	$1.0 \cdot 10^{-4}$
Nodes	31×31
Size, m	1.0×1.0
Pressure, Pa	$2.0 \cdot 10^5$

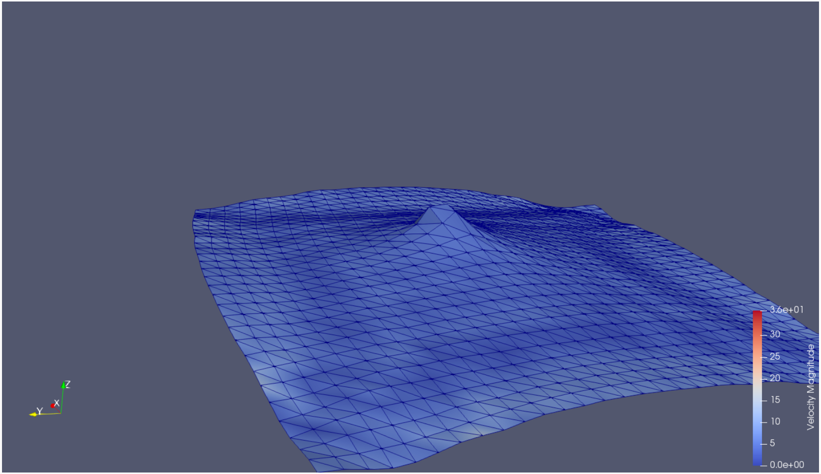
Skew strike



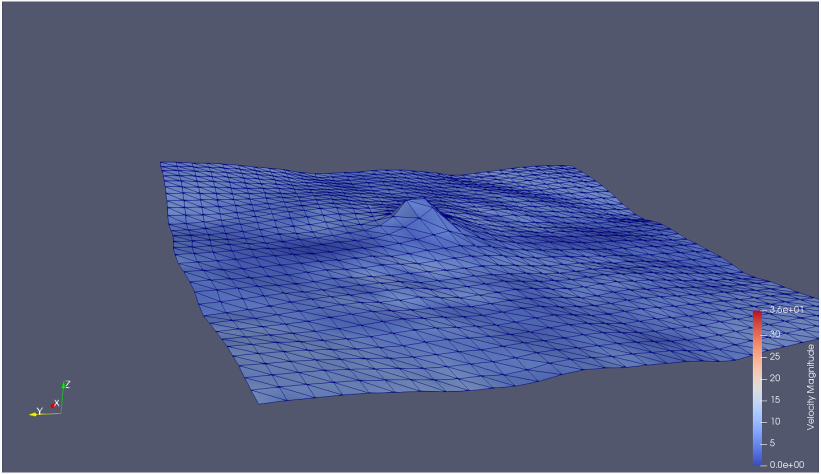
Skew strike



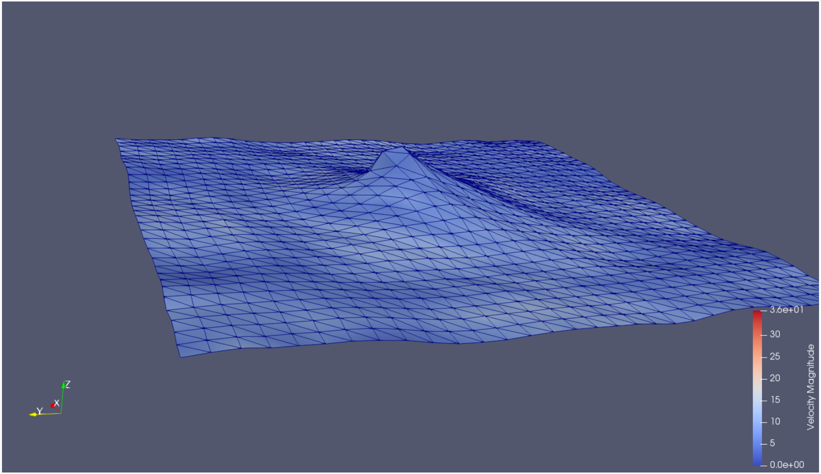
Skew strike



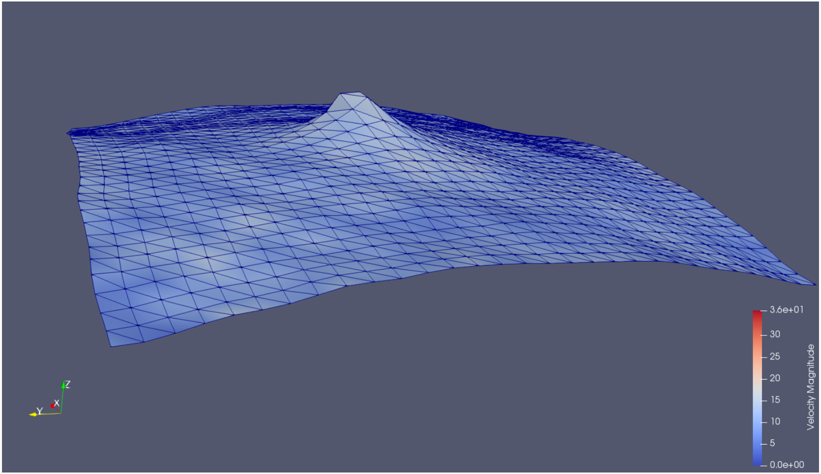
Skew strike



Skew strike



Skew strike



Results overview

- Deformation cone for isotropic material model (instead of deformation cross obtained in related works based on pure textile membrane model)
- Model can be generalized to handle anisotropic materials with arbitrary anisotropy type
- Model handles large displacements and membrane movement as a whole

Future plans

Model:

- Shape functions and material properties depending on z to simulate multi-layered composite fabrics
- Destruction modelling

Application:

- RAM usage optimization — sparse matrices libraries
- Advanced grid generation and dynamic refining
- GPU acceleration

Questions?

Formulas i

$$\alpha_i = \frac{1}{S_e}\begin{vmatrix} x_j & y_j \\ x_k & y_k \end{vmatrix} \quad \beta_i = -\frac{1}{S_e}(y_k - y_j) \quad \gamma_i = \frac{1}{S_e}(x_k - x_j)$$

$$S_e = \begin{vmatrix} 1 & x_i & y_i \\ 1 & x_j & y_j \\ 1 & x_k & y_k \end{vmatrix} \text{ — doubled triangle area}$$

$$\mathsf{B} = \begin{bmatrix} \mathsf{B}_1^1 & \mathsf{B}_2^1 & \mathsf{B}_3^1 \\ \mathsf{B}_1^2 & \mathsf{B}_2^2 & \mathsf{B}_3^2 \end{bmatrix} \tag{0.1}$$

$$\mathsf{B}_i^1 = \begin{bmatrix} \beta_i & 0 & 0 \\ 0 & \gamma_i & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathsf{B}_i^2 = \begin{bmatrix} \gamma_i & \beta_i & 0 \\ 0 & 0 & \beta_i \\ 0 & 0 & \gamma_i \end{bmatrix} \tag{0.2}$$

Formulas ii

$$\mathbf{f}_e = -\int_{V_e} \mathbf{N}^T \mathbf{b} dV_e = -\int_{V_e} \begin{bmatrix} \mathbf{N}_1^T \mathbf{b} \\ \mathbf{N}_2^T \mathbf{b} \\ \mathbf{N}_3^T \mathbf{b} \end{bmatrix} dV_e = \frac{\frac{1}{2} S_e h_e}{3} \begin{bmatrix} \mathbf{b} \\ \mathbf{b} \\ \mathbf{b} \end{bmatrix} \tag{0.3}$$

$$\mathbf{K}_e = \left[\mathbf{K}_{ij} \right]_{i,j \in \overline{1,3}} - \text{matrix of } 3 \times 3 \text{ blocks } \mathbf{K}_{ij} \text{ of } 3 \times 3 \tag{0.4}$$

$$\mathbf{K}_{ij} = \mathbf{B}_i^1 \mathbf{D}_1 \mathbf{B}_j^1 + \mathbf{B}_i^2 \mathbf{D}_2 \mathbf{B}_j^2$$

Formulas iii

$$\mathsf{M}_e = \left[\mathsf{M}_{ij}\right]_{i,j\in\overline{1,3}}$$

$$\mathsf{M}_{ij} = \frac{\rho_e h_e S_e}{24}(12\alpha_i\alpha_j + \beta_i\beta_j(x_i^2 + x_j^2 + x_k^2)+ \\ + \gamma_i\gamma_j(y_i^2 + y_j^2 + y_k^2) + (\beta_i\gamma_j + \beta_j\gamma_i)(x_i y_i + x_j y_j + x_k y_k))\mathsf{E} \quad (0.5)$$

Here node coordinates (x, y) are taken in triangle barycenter system, to get them one needs to perform a shift:

$$x_i = x_i^0 - \frac{1}{3}(x_1^0 + x_2^0 + x_3^0), \quad y_i = y_i^0 - \frac{1}{3}(y_1^0 + y_2^0 + y_3^0)$$

N. Newmark.
A Method of Computation for Structural Dynamics.
Number №№ 179-181 in A Method of Computation for Structural Dynamics. American Society of Civil Engineers, 1959.

O. Zienkiewicz and R. Taylor.
Finite Element Method: Volume 1 - The Basis.
Butterworth-Heinemann, Oxford, 5th edition, 2000.