Material Model Library Manual

Bořek Patzák

Czech Technical University
Faculty of Civil Engineering
Department of Structural Mechanics
Thákurova 7, 166 29 Prague, Czech Republic

November 30, 2010

Contents

List of Tables

1 Material Models for Structural Analysis

1.1 Elastic materials

1.1.1 Isotropic linear elastic material

Linear isotropic material model. The model parameters are summarized in table 1.

Description	Linear isotropic elastic material
Record Format	**IsoLE** $\text{num}_{(in)}$ # $\text{d}_{(rn)}$ # $\text{E}_{(rn)}$ # $\text{n}_{(rn)}$ # $\text{tAlpha}_{(rn)}$ #
Parameters	- *num* material model number - *d* material density - *E* Young modulus - *n* Poisson ratio - *tAlpha* thermal dilatation coefficient
Supported modes	3dMat, PlaneStress, PlaneStrain, 1dMat, 2dPlateLayer, 2dBeamLayer, 3dShellLayer, 2dPlate, 2dBeam, 3dShell, 3dBeam, PlaneStressRot
Features	Adaptivity support

Table 1: Linear Isotropic Material - summary.

1.1.2 Orthotropic linear elastic material

Linear orthotropic, linear elastic material model. The model parameters are summarized in table 2. Local coordinate system, which determines axes of material orthotrophy can by specified using *lcs* array. This array contains six numbers, where the first three numbers represent directional vector of a local x-axis, and next three numbers represent directional vector of a local y-axis. The local z-axis is determined using the vector product. The right-hand coordinate system is assumed.

Local coordinate system can also be specified using *scs* parameter. Then local coordinate system is specified in so called "shell" coordinate system, which is defined locally on each particular element and its definition is as follows: principal z-axis is perpendicular to shell mid-section, x-axis is perpendicular to z-axis and normal to user specified vector (so x-axis is parallel to plane, with being normal to this plane) and y-axis is perpendicular both to x and z axes. This definition of coordinate system can be used only with plates and shells elements. When vector is parallel to z-axis an error occurs. The *scs* array contain three numbers defining direction vector . If no local coordinate system is specified, by default a global coordinate system is used.

For 3D case the material compliance matrix has the following form

$$\boldsymbol{C} = \begin{bmatrix} 1/E_X & -\nu_{xy}/E_x & -\nu_{xz}/E_x & 0 & 0 & 0 \\ -\nu_{yx}/E_y & 1/E_y & -\nu_{yz}/E_y & 0 & 0 & 0 \\ -\nu_{zx}/E_z & -\nu_{zy}/E_z & 1/E_z & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/G_{yz} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/G_{xz} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/G_{xy} \end{bmatrix} \tag{1}$$

By inversion, the material stiffness matrix has the form

$$\boldsymbol{D} = \begin{bmatrix} d_{xx} & d_{xy} & d_{xz} & 0 & 0 & 0 \\ & d_{yy} & d_{yz} & 0 & 0 & 0 \\ \text{sym} & & d_{zz} & 0 & 0 & 0 \\ 0 & 0 & 0 & G_{yz} & 0 & 0 \\ 0 & 0 & 0 & 0 & G_{xz} & 0 \\ 0 & 0 & 0 & 0 & 0 & G_{xy} \end{bmatrix} \tag{2}$$

where $\xi = 1 - (\nu_{xy} * \nu_{yx} + \nu_{yz} * \nu_{zy} + \nu_{zx} * \nu_{xz}) - (\nu_{xy} * \nu_{yz} * \nu_{zx} + \nu_{yx} * \nu_{zy} * \nu_{xz})$ and

$$\begin{aligned} d_{xx} &= E_X(1 - \nu_{yz} * \nu_{zy})/\xi, && (3)\\ d_{xy} &= E_y * (\nu_{xy} + \nu_{xz} * \nu_{zy})/\xi, && (4)\\ d_{xz} &= E_z * (\nu_{xz} + \nu_{yz} * \nu_{xy})/\xi, && (5)\\ d_{yy} &= E_y * (1 - \nu_{xz} * \nu_{zx})/\xi, && (6)\\ d_{yz} &= E_z * (\nu_{yz} + \nu_{yx} * \nu_{xz})/\xi, && (7)\\ d_{zz} &= E_z * (1 - \nu_{yx} * \nu_{xy})/\xi. && (8) \end{aligned}$$

E_i is the Young's modulus in the i-th direction, G_{ij} is the shear modulus in ij plane, ν_{ij} major Poisson's ratio, and ν_{ji} minor Poisson's ratio. Assuming that $E_x > E_y > E_z$, $\nu_{xy} > \nu_{yx}$ etc., then the ν_{xy} is refered to as major Poisson's ratio, while ν_{yx} refered as minor Poisson's ratio. Note, that there is only nine independent material parameters, because of symmetry conditions. The symmetry condition yields

$$\nu_{xy}E_y = \nu_{yx}E_x, \quad \nu_{yz}E_z = \nu_{zy}E_y, \quad \nu_{zx}E_x = \nu_{xz}E_z$$

The model description and parameters are summarized in table 2.

1.1.3 Hyperelastic material

This material model can describe elastic behavior at large strains. A hyperelastic model postulates the existence of free energy potential. Existence of the potential implies reversibility of deformations and no energy dissipation during loading process. Here we use the free energy function introduced in [9]

$$\rho_0\psi = \frac{1}{4}\left(K - \frac{2}{3}G\right)\left(J^2 - 2\text{lnJ} - 1\right) + G\left(\boldsymbol{E} : \boldsymbol{I} - \text{lnJ}\right) \tag{9}$$

where K is the bulk modulus, G is the shear modulus, J is the Jacobian (determinant of the deformation gradient, corresponding to the ratio of the current and initial volume) and $\boldsymbol{E}$ is the Green-Lagrange strain. Then stress-strain law can be derived from (9) as

$$\boldsymbol{S} = \rho_0\frac{\partial\psi}{\partial\boldsymbol{E}} = \frac{1}{2}\left(K - \frac{2}{3}G\right)\left(J^2 - 1\right)\boldsymbol{C}^{-1} + G\left(\boldsymbol{I} - \boldsymbol{C}^{-1}\right) \tag{10}$$

where $\boldsymbol{S}$ is the second Piola-Kirchhoff stress, $\boldsymbol{E}$ is the Green-Lagrange strain and $\boldsymbol{C}$ is the right Cauchy-Green tensor. The model description and parameters are summarized in table 3.

Description	Orthotropic, linear elastic material
Record Format	**OrthoLE** num$_{(in)}$ # d$_{(rn)}$ # Ex$_{(rn)}$ # Ey$_{(rn)}$ # Ez$_{(rn)}$ # NYyz$_{(rn)}$ # NYxz$_{(rn)}$ # NYxy$_{(rn)}$ # Gyz$_{(rn)}$ # Gxz$_{(rn)}$ # Gxy$_{(rn)}$ # tAlphax$_{(rn)}$ # tAlphay$_{(rn)}$ # tAlphaz$_{(rn)}$ # [lcs$_{(ra)}$ #] [scs$_{(ra)}$ #]
Parameters	- *num* material model number - *d* material density - *Ex*, *Ey*, *Ez* Young moduluses for x,y, and z directions - *NYyz*, *NYxz*, *NYxy* major Poisson's ratio coefficients - *Gyz*, *Gxz*, *Gxy* Shear moduluses - *tAlphax*, *tAlphay*, *tAlphaz* thermal dilatation coefficients in x,y,z directions - *lcs* Array defining local material x and y axes of orthotrophy - *scs* Array defining a normal vector n. The local x axis is parallel to plane with n being plane normal. The material local z-axis is perpendicular to shell mid-section.
Supported modes	3dMat, PlaneStress, PlaneStrain, 1dMat, 2dPlateLayer, 2dBeamLayer, 3dShellLayer, 2dPlate, 2dBeam, 3dShell, 3dBeam, PlaneStressRot

Table 2: Orthotropic, linear elastic material – summary.

Description	Hyperelastic material
Record Format	**HyperMat** $_{(in)}$ # d$_{(rn)}$ # K$_{(rn)}$ # G$_{(rn)}$ #
Parameters	- material number - *d* material density - *K* bulk modulus - *G* shear modulus
Supported modes	3dMat

Table 3: Hyperelastic material - summary.

1.2 Plasticity-based material models

1.2.1 Drucker-Prager model

The Drucker-Prager plasticity model[1] is an isotropic elasto-plastic model based on a yield function

$$f(\boldsymbol{\sigma}, \tau_{\mathrm{Y}}) = F(\boldsymbol{\sigma}) - \tau_{\mathrm{Y}} \tag{11}$$

with the pressure-dependent equivalent stress

$$F(\boldsymbol{\sigma}) = \alpha I_1 + \sqrt{J_2} \tag{12}$$

As usual, $\boldsymbol{\sigma}$ is the stress tensor, τ_{Y} is the yield stress under pure shear, and I_1 and J_2 are the first invariant and second deviatoric invariant of the stress tensor. The friction coefficient α is a positive parameter that controls the influence of the pressure on the yield limit, important for cohesive-frictional materials such

[1] Contributed by Simon Rolshoven, LSC, FENAC, EPFL.

as concrete, soils or other geomaterials. The flow rule is derived from the plastic potential

$$g\left(\boldsymbol{\sigma}\right) = \alpha_\psi I_1 + \sqrt{J_2} \tag{13}$$

where α_ψ is the dilatancy coefficient. An associated model with $\alpha = \alpha_\psi$ would overestimate the dilatancy of concrete, so the dilatancy coefficient is usually chosen smaller than the friction coefficient. The model is described by the equations

$$\begin{aligned}
\boldsymbol{\sigma} &= \boldsymbol{D} : (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_\mathrm{p}) && (14)\\
\tau_\mathrm{Y} &= h(\kappa) && (15)\\
\dot{\boldsymbol{\varepsilon}}_\mathrm{p} = \dot{\lambda}\frac{\partial g}{\partial \boldsymbol{\sigma}} &= \dot{\lambda}\left(\alpha_\psi \boldsymbol{\delta} + \frac{\boldsymbol{s}}{2\sqrt{J_2}}\right) && (16)\\
\dot{\kappa} &= \sqrt{\frac{2}{3}}\,\|\dot{\boldsymbol{\varepsilon}}_\mathrm{p}\| && (17)
\end{aligned}$$

$$\dot{\lambda} \geq 0, \quad f\left(\boldsymbol{\sigma}, \tau_\mathrm{Y}\right) \leq 0, \quad \dot{\lambda}\, f\left(\boldsymbol{\sigma}, \tau_\mathrm{Y}\right) = 0 \tag{18}$$

which represent the linear elastic law, hardening law, evolution laws for plastic strain and hardening variable, and the loading-unloading conditions. In the above, $\boldsymbol{D}$ is the elastic stiffness tensor, $\boldsymbol{\varepsilon}$ is the strain tensor, $\boldsymbol{\varepsilon}_\mathrm{p}$ is the plastic strain tensor, λ is the plastic multiplier, $\boldsymbol{\delta}$ is the unit second-order tensor, $\boldsymbol{s}$ is the deviatoric stress tensor, κ is the hardening variable, and a superior dot marks the derivative with respect to time. The flow rule has the form given in Eq. (16) at all points of the conical yield surface with the exception of its vertex, located on the hydrostatic axis.

For the present model, the evolution of the hardening variable can be explicitly linked to the plastic multiplier. Substituting the flow rule (16) into Eq. (17) and computing the norm leads to

$$\dot{\kappa} = k\dot{\lambda} \tag{19}$$

with a constant parameter $k = \sqrt{1/3 + 2\alpha_\psi^2}$, so the hardening variable is proportional to the plastic multiplier. For $\alpha = \alpha_\psi = 0$, the associated J_2-plasticity model is recovered as a special case.

In the simplest case of linear hardening, the hardening function is a linear function of κ, given by

$$h(\kappa) = \tau_0 + HE\kappa \tag{20}$$

where τ_0 is the initial yield stress, and H is the hardening modulus normalized with the elastic modulus. Alternatively, an exponential hardening function

$$h(\kappa) = \tau_\mathrm{limit} + (\tau_0 - \tau_\mathrm{limit})\,\mathrm{e}^{-\kappa/\kappa_\mathrm{c}} \tag{21}$$

can be used for a more realistic description of hardening.

The stress-return algorithm is based on the Newton-iteration. In plasticity, this is commonly called Closest-Point-Projection (CPP), and it generally leads to quadratic convergence. The implemented algorithm is convergent in any stress case, but in the vicinity of the vertex region, quadratic convergence might be lost because of insufficient regularity of the yield function.

The algorithmic tangent stiffness matrix is implemented for both the regular case and the vertex region. Generally, the error decreases quadratically

(of course only asymptotically). Again, in the vicinity of the vertex region, quadratic convergence might be lost due to insufficient regularity. Furthermore, the tangent stiffness matrix does not always exist for the vertex case. In these cases, the elastic stiffness is used instead. It is generally safer (but slower) to use the elastic stiffness if you encounter any convergence problems, especially if your problem is tension-dominated.

Description	DP material
Record Format	**DruckerPrager** num$_{(in)}$ # d$_{(rn)}$ # tAlpha$_{(rn)}$ # E$_{(rn)}$ # n$_{(rn)}$ # alpha$_{(rn)}$ # alphaPsi$_{(rn)}$ # ht$_{(in)}$ # iys$_{(rn)}$ # lys$_{(rn)}$ # hm$_{(rn)}$ # kc$_{(rn)}$ # [yieldtol$_{(rn)}$ #]
Parameters	- *num* material model number - *d* material density - *tAlpha* thermal dilatation coefficient - *E* Young modulus - *n* Poisson ratio - *alpha* friction coefficient - *alphaPsi* dilatancy coefficient - *ht* hardening type, 1: linear hardening, 2: exponential hardening - *iys* initial yield stress in shear, τ_0 - *lys* limit yield stress for exponential hardening, τ_{limit} - *hm* hardening modulus normalized with E-modulus (!) - *kc* κ_c for the exponential softening law - *yieldtol* tolerance of the error in the yield criterion, default value 1.e-14

Table 4: DP material - summary.

1.2.2 Hardening plasticity model with Mises yield condition

This model is appropriate for plastic yielding in material such as metals. It is based on the Mises yield condition (in terms of the second deviatoric invariant, J_2), associated flow rule and linear isotropic hardening driven by the cumulative plastic strain. The model can be used in the small-strain context, with additive split of the strain tensor into the elastic and plastic parts, or in the large-strain context, with multiplicative split of the deformation gradient and with yield condition formulated in terms of Kirchhoff stress (which is the true Cauchy stress multiplied by the Jacobian).

Small strain formulation: The basic equations include the additive decomposition of strain into elastic and plastic parts,

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_{\mathrm{e}} + \boldsymbol{\varepsilon}_{\mathrm{p}}, \tag{22}$$

the stress strain law

$$\boldsymbol{\sigma} = \boldsymbol{D} : (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_{\mathrm{p}}), \tag{23}$$

the yield condition

$$f(\boldsymbol{s}, \kappa) = \sqrt{\frac{3}{2}\boldsymbol{s} : \boldsymbol{s}} - \sigma_Y(\kappa), \tag{24}$$

the incremental definition of cumulative plastic strain

$$\dot{\kappa} = \|\dot{\boldsymbol{\varepsilon}}_{\mathrm{p}}\|, \tag{25}$$

the linear hardening law

$$\sigma_Y(\kappa) = H\kappa, \tag{26}$$

the evolution law for the plastic strain

$$\dot{\boldsymbol{\varepsilon}}_{\mathrm{p}} = \dot{\lambda}\frac{\partial f}{\partial \boldsymbol{s}}, \tag{27}$$

and the loading-unloading conditions

$$\dot{\lambda} > 0 \qquad f(\boldsymbol{s}, \kappa) \leq 0 \qquad \dot{\lambda} f(\boldsymbol{s}, \kappa) = 0. \tag{28}$$

In the equations above, $\boldsymbol{\varepsilon}$ is the strain tensor, $\boldsymbol{\varepsilon}_{\mathrm{e}}$ is the elastic strain tensor, $\boldsymbol{\varepsilon}_p$ is the plastic strain tensor, $\boldsymbol{D}$ is the elastic stiffness tensor, $\boldsymbol{s}$ is the deviatoric stress tensor, σ_Y is the magnitude of stress at yielding under uniaxial tension (or compression), κ is the cumulated plastic strain, H is the hardening modulus and λ is the plastic multiplier.

Large strain formulation is based on the introduction of an intermediate local configuration, relative to which the elastic response is characterized. This notion leads to a multiplicative decomposition of deformation gradient into elastic and plastic parts:

$$\boldsymbol{F} = \boldsymbol{F}^e \boldsymbol{F}^p. \tag{29}$$

The stress-evaluation algorithm can be based on the classical radial return mapping; see [9] for more details. The model description and parameters are summarized in table 5.

Description	Von Mises Plasticity with isotropic hardening
Record Format	**MisesMat** (in) # d(rn) # E(rn) # n(rn) # sig0(rn) # H(rn) #
Parameters	- material number - *d* material density - *E* Young modulus - *n* Poisson ratio - *sig0* initial yield stress in uniaxial tension(compression) - *H* hardening modulus
Supported modes	3dMat,3dMatF

Table 5: Mises plasticity – summary.

1.2.3 Perfectly plastic material with Mises yield condition

This is an older model, kept here for compatibility with previous versions. It uses Mises plasticity condition with no hardening and under small strain only. The model description and parameters are summarized in table 6.

Description	Perfectly plastic material with Mises condition
Record Format	**Steel1** num$_{(in)}$ # d$_{(rn)}$ # E$_{(rn)}$ # n$_{(rn)}$ # tAlpha$_{(rn)}$ # Ry$_{(rn)}$ #
Parameters	- *num* material model number - *d* material density - *E* Young modulus - *n* Poisson ratio - *tAlpha* thermal dilatation coefficient - *Ry* uniaxial yield stress
Supported modes	3dMat, PlaneStress, PlaneStrain, 1dMat, 2dPlateLayer, 2dBeamLayer, 3dShellLayer 3dBeam, PlaneStressRot

Table 6: Perfectly plastic material with Mises condition – summary.

1.2.4 Composite plasticity model for masonry

Masonry is a composite material made of bricks and mortar. Nonlinear behavior of both components should be considered to obtain a realistic model able to describe cracking, slip, and crushing of the material. The model is based on paper by Lourenco and Rots [6]. It is formulated on the basis of softening plasticity for tension, shear, and compression (see fig.(1)). Numerical implementation is based on modern algorithmic concepts such as implicit integration of the rate equations and consistent tangent stiffness matrices.

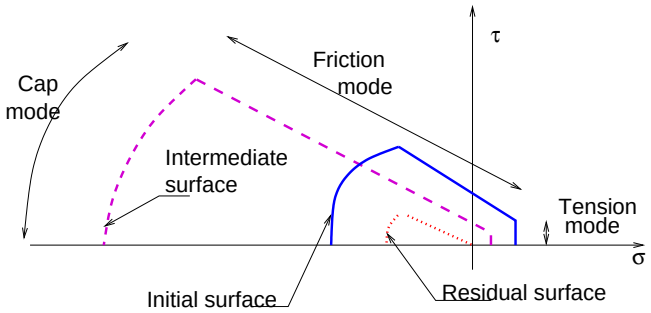

Figure 1: Composite yield surface model for masonry

The approach used in this work is based on idea of concentrating all the damage in the relatively weak joints and, if necessary, in potential tension cracks in the bricks. The joint interface constitutive model should include all important damage mechanisms. Here, the concept of interface elements is used. An interface element allows to incorporate discontinuities in the displacement field and its behavior is described in terms of a relation between the tractions and relative displacement across the interface. In the present work, these quantities will be denoted as $\boldsymbol{\sigma}$, generalized stress, and $\boldsymbol{\varepsilon}$, generalized strain. For 2D configuration, $\boldsymbol{\sigma} = \{\sigma, \tau\}^T$ and $\boldsymbol{\varepsilon} = \{u_n, u_s\}^T$, where σ and τ are the normal and shear components of the traction interface vector and n and s subscripts distinguish between normal and shear components of displacement vector. The elastic response is characterized in terms of elastic constitutive matrix $\boldsymbol{D}$ as

$$\boldsymbol{\sigma} = \boldsymbol{D}\boldsymbol{\varepsilon} \tag{30}$$

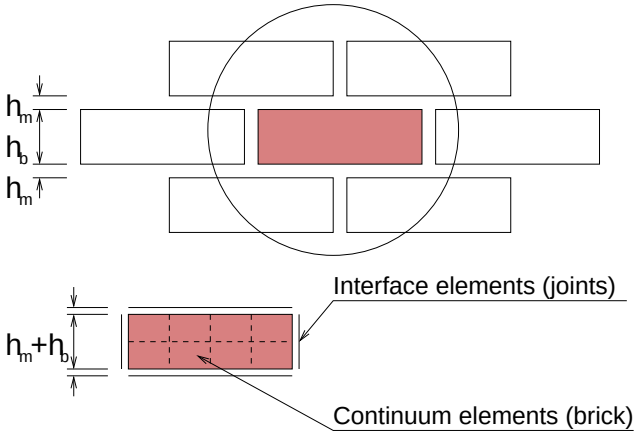

Figure 2: Modeling strategy for masonry

For a 2D configuration $\boldsymbol{D} = diag\{k_n, k_s\}$. The terms of the elastic stiffness matrix can be obtained from the properties of both masonry and joints as

$$k_n = \frac{E_b E_m}{t_m(E_b - E_m)};\ k_s = \frac{G_b G_m}{t_m(G_b - G_m)} \tag{31}$$

where E_b and E_m are Young's moduli, G_b and G_m shear moduli for brick and mortar, and t_m is the thickness of joint. One should note, that there is no contact algorithm assumed between bricks, this means that the overlap of neighboring units will be visible. On the other hand, the interface model includes a compressive cap, where the compressive inelastic behavior of masonry is lumped.

Tension mode In the tension mode, the exponential softening law is assumed (see fig.(3)). The yield function has the following form

$$f_1(\boldsymbol{\sigma}, \kappa_1) = \sigma - f_t(\kappa_1) \tag{32}$$

where the yield value f_t is defined as

$$f_t = f_{t0} \exp\left(-\frac{f_{t0}}{G_f^I}\kappa_1\right) \tag{33}$$

The f_{t0} represents tensile strength of joint or interface; and G_f^I is mode-I fracture energy. For the tension mode, the associated flow hypothesis is assumed.

Shear mode For the shear mode a Coulomb friction envelope is used. The yield function has the form

$$f_2(\boldsymbol{\sigma}, \kappa_2) = |\tau| + \sigma \tan\phi(\kappa_2) - c(\kappa_2) \tag{34}$$

According to [6] the variations of friction angle ϕ and cohesion c are assumed as

$$c = c_0 \exp\left(-\frac{c_0}{G_f^{II}}\kappa_2\right) \tag{35}$$

$$\tan\phi = \tan\phi_0 + (\tan\phi_r - \tan\phi_0)\left(\frac{c_0 - c}{c_0}\right) \tag{36}$$

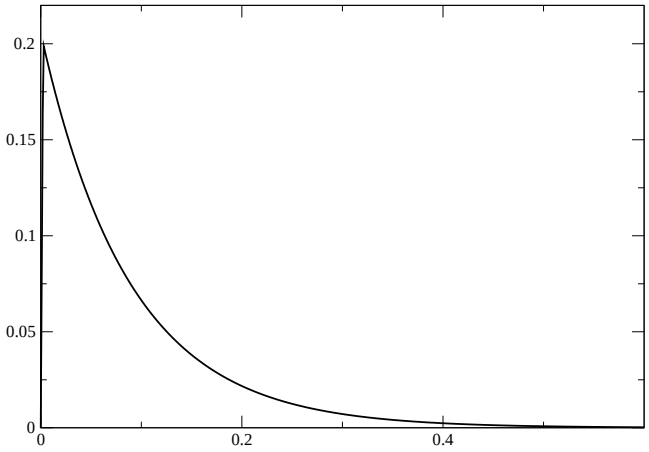

Figure 3: Tensile behavior of proposed model (f_t = 0.2 MPa, $\mathrm{G_f^I}$ = 0.018 N/mm)

where c_0 is initial cohesion of joint, ϕ_0 initial friction angle, ϕ_r residual friction angle, and G_f^{II} fracture energy in mode II failure. A non-associated plastic potential g_2 is considered as

$$g_2 = |\tau| + \sigma \tan \Phi - c \tag{37}$$

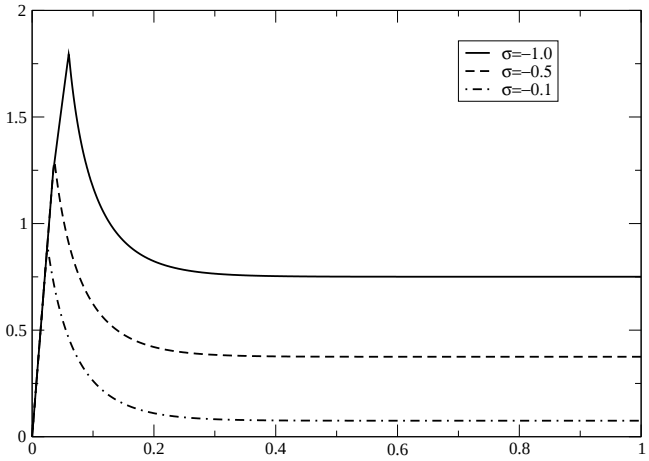

Figure 4: Shear behavior of proposed model for different confinement levels in MPa ($c_0 = 0.8$ MPa, $\tan\phi_0 = 1.0$, $\tan\phi_\mathrm{r} = 0.75$, and $\mathrm{G_f^{II}} = 0.05$ N/mm)

Coupling of tension/shear modes The tension and Coulomb friction modes are coupled with isotropic softening. This means that the percentage of soften-

ing in the cohesion is assumed to be the same as on the tensile strength

$$\dot{\kappa}_1 = \lambda_1 + \frac{G_f^I}{G_f^{II}} \frac{c_0}{f_{t0}} \lambda_2; \quad \dot{\kappa}_2 = \frac{G_f^{II}}{G_f^I} \frac{f_{t0}}{c_0} \lambda_1 + \lambda_2 \tag{38}$$

This follows from (33) and (35). However, in the corner region, when both yield surfaces are activated, such approach will lead to a non-acceptable penalty. For this reason a quadratic combination is assumed

$$\dot{\kappa}_1 = \sqrt{(\lambda_1)^2 + \left(\frac{G_f^I}{G_f^{II}} \frac{c_0}{f_{t0}} \lambda_2 \right)^2}; \quad \dot{\kappa}_2 = \sqrt{\left(\frac{G_f^{II}}{G_f^I} \frac{f_{t0}}{c_0} \lambda_1 \right)^2 + (\lambda_2)^2} \tag{39}$$

Cap mode For the cap mode, an ellipsoid interface model is used. The yield condition is assumed as

$$f_3(\boldsymbol{\sigma}, \kappa_3) = C_{nn}\sigma^2 + C_{ss}\tau^2 + C_n\sigma - \bar{\sigma}^2(\kappa_3) \tag{40}$$

where C_{nn}, C_{ss}, and C_n are material model parameters and $\bar{\sigma}$ is yield value, originally assumed in the following form of hardening/softening law [6]

$$\begin{aligned}
\bar{\sigma}_1(\kappa_3) &= \bar{\sigma}_i + (\bar{\sigma}_p - \bar{\sigma}_i)\sqrt{\frac{2\kappa_3}{\kappa_p} - \frac{\kappa_3^2}{\kappa_p^2}}; \quad \kappa_3 \in (0, \kappa_p) \\
\bar{\sigma}_2(\kappa_3) &= \bar{\sigma}_p + (\bar{\sigma}_m - \bar{\sigma}_p)\left(\frac{\kappa_3 - \kappa_p}{\kappa_m - \kappa_p} \right)^2; \quad \kappa_3 \in (\kappa_p, \kappa_m) \\
\bar{\sigma}_3(\kappa_3) &= \bar{\sigma}_r + (\bar{\sigma}_m - \bar{\sigma}_r)\exp\left(m\frac{\kappa_3 - \kappa_m}{\bar{\sigma}_m - \bar{\sigma}_r} \right); \quad \kappa_3 \in (\kappa_m, \infty)
\end{aligned} \tag{41}$$

with $m = 2(\bar{\sigma}_m - \bar{\sigma}_p)/(\kappa_m - \kappa_p)$. The hardening/softening law (41) is shown in fig.(5). Note that the curved diagram is a C^1 continuous $\sigma - \kappa_3$ relation. The energy under the load-displacement diagram can be related to a "compressive fracture energy". The original hardening law (41.1) exhibits indefinite slope for $\kappa_3 = 0$, which can cause the problems with numerical implementation. This has been overcomed by replacing this hardening law with parabolic equation given by

$$\bar{\sigma}_1(\kappa_3) = \bar{\sigma}_i - 2 * (\bar{\sigma}_i - \bar{\sigma}_p) * \frac{\kappa_3}{\kappa_p} + (\bar{\sigma}_i - \bar{\sigma}_p)\frac{\kappa_3}{\kappa_p} \tag{42}$$

An associated flow and strain hardening hypothesis are being considered. This yields

$$\dot{\kappa}_3 = \lambda_3 \sqrt{(2C_{nn}\sigma + C_n) * (2C_{nn}\sigma + C_n) + (2C_{ss}\tau) * (2C_{ss}\tau)} \tag{43}$$

The model parameters are summarized in table 7. There is one algorithmic issue, that follows from the model formulation. Since the cap mode hardening/softening is not coupled to hardening/softening of shear and tension modes the it may happen that when the cap and shear modes are activated, the return directions become parallel for both surfaces. This should be avoided by adjusting the input parameters accordingly (one can modify dilatancy angle, for example).

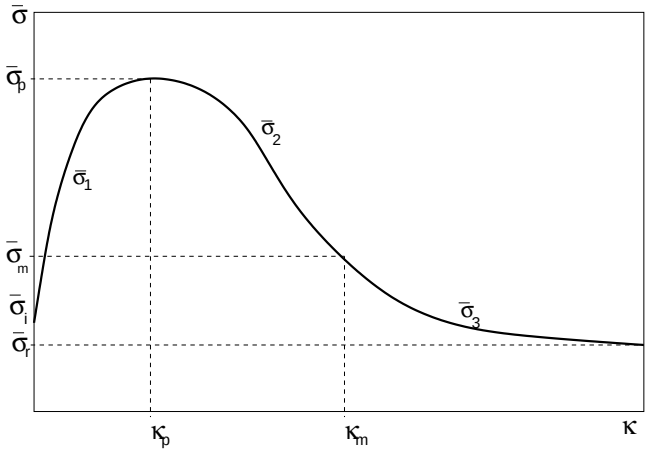

Figure 5: Hardening/softening law for cap mode

Description	Composite plasticity model for masonry
Record Format	**Masonry02** num(in) # d(rn) # E(rn) # n(rn) # ft0(rn) # gfi(rn) # gfii(rn) # kn(rn) # ks(rn) # c0(rn) # tanfi0(rn) # tanfir(rn) # tanpsi(rn) # si(rn) # sp(rn) # sm(rn) # sr(rn) # kp(rn) # km(rn) # kr(rn) # cnn(rn) # css(rn) # cn(rn) #
Parameters	- *num* material model number - *d* material density - *E* Young modulus - *n* Poisson ratio - *ft0* tensile strength - *gfi* fracture energy for mode I - *gfii* fracture energy for mode II - *kn* joint elastic property - *ks* joint elastic property - *c0* initial cohesion - *tanfi0* initial friction angle - *tanfir* residual friction angle - *tanpsi* dilatancy - {*si, sp, sm, sr*} cap parameters $\{\bar{\sigma}_i, \bar{\sigma}_p, \bar{\sigma}_m, \bar{\sigma}_r\}$ - {*kp, km,kr*} cap parameters $\{\kappa_p, \kappa_m, \kappa_r\}$ - *cnn,css,cn* cap mode parametrs
Supported modes	_2dInterface

Table 7: Composite model for masonry - summary.

1.2.5 Nonlinear elasto-plastic material model for concrete plates and shells

Nonlinear elasto-plastic material model with hardening. Takes into account uniaxial stress + transverse shear in concrete layers with transverse stirrups. Can be used only for 2d plates and shells with layered cross section and together with explicit integration method (stiffness matrix is not provided). The model description and parameters are summarized in table 8.

Description	Nonlinear elasto-plastic material model for concrete plates and shells
Record Format	**Concrete2** num$_{(in)}$ # d$_{(rn)}$ # E$_{(rn)}$ # n$_{(rn)}$ # SCCC$_{(rn)}$ # SCCT$_{(rn)}$ # EPP$_{(rn)}$ # EPU$_{(rn)}$ # EOPU$_{(rn)}$ # EOPP$_{(rn)}$ # SHEARTOL$_{(rn)}$ # IS_PLASTIC_FLOW$_{(in)}$ # IFAD$_{(in)}$ # STIRR_E$_{(rn)}$ # STIRR_Ft$_{(rn)}$ # STIRR_A$_{(rn)}$ # STIRR_TOL$_{(rn)}$ # STIRR_EREF$_{(rn)}$ # STIRR_LAMBDA$_{(rn)}$ #
Parameters	- *num* material model number - *d* material density - *E* Young modulus - *n* Poisson ratio - *SCCC* pressure strength - *SCCT* tension strength - *EPP* threshold effective plastic strain for softening in compression - *EPU* ultimate eff. plastic strain - *EOPP* threshold volumetric plastic strain for softening in tension - *EOPU* ultimate volumetric plastic strain - *SHEARTOL* threshold value of the relative shear deformation (psi**2/eef) at which shear is considered in layers. For lower relative shear deformations the transverse shear remains elastic decoupled from bending. default value SHEARTOL = 0.01 - *IS_PLASTIC_FLOW* indicates that plastic flow (not deformation theory) is used in pressure - *IFAD* State variables will not be updated, otherwise update state variables - *STIRR_E* Young modulus of stirrups - *STIRR_R* stirrups uniaxial strength = elastic limit - *STIRR_A* stirrups area/unit length (beam) or /unit area (shell) - *STIRR_TOL* stirrups tolerance of equilibrium in the z direction (=0 no iteration) - *STIRR_EREF* stirrups reference strain rate for Peryzna's material - *STIRR_LAMBDA* coefficient for that material (stirrups) - *SHTIRR_H* isotropic hardening factor for stirrups
Supported modes	3dShellLayer, 2dPlateLayer

Table 8: Nonlinear elasto-plastic material model for concrete - summary.

1.3 Material models for tensile failure

1.3.1 Nonlinear elasto-plastic material model for concrete plates and shells

The description can be found is section 1.2.5.

1.3.2 Smeared rotating crack model

Implementation of smeared rotating crack model. Virgin material is modeled as isotropic linear elastic material (described by Young modulus and Poisson ratio). The onset of cracking begins, when principal stress reaches tensile strength. Further behavior is then determined by softening law, governed by principle of preserving of fracture energy G_f. For large elements, the tension strength can be artificially reduced to preserve fracture energy. Multiple cracks are allowed. The elastic unloading and reloading is assumed. In compression regime, this model correspond to isotropic linear elastic material. The model description and parameters are summarized in table 9.

Description	Rotating crack model for concrete
Record Format	**Concrete3** $d_{(rn)}$ # $E_{(rn)}$ # $n_{(rn)}$ # $Gf_{(rn)}$ # $Ft_{(rn)}$ # $exp\_soft_{(in)}$ # $tAlpha_{(rn)}$ #
Parameters	- *num* material model number - *d* material density - *E* Young modulus - *n* Poisson ratio - *Gf* fracture energy - *Ft* tension strength - *exp_soft* determines the type of softening (0 = exponential, 1 = linear) - *tAlpha* thermal dilatation coefficient
Supported modes	3dMat, PlaneStress, PlaneStrain, 1dMat, 2dPlateLayer, 2dBeamLayer, 3dShellLayer

Table 9: Rotating crack model for concrete - summary.

1.3.3 Smeared rotating crack model with transition to scalar damage - linear softening

Implementation of smeared rotating crack model with transition to scalar damage with linear softening law. Improves the classical rotating model (see section 1.3.2) by introducing the transition to scalar damage model in later stages of tension softening.

Traditional smeared-crack models for concrete fracture are known to suffer by stress locking (meaning here spurious stress transfer across widely opening cracks), mesh-induced directional bias, and possible instability at late stages of the loading process. The combined model keeps the anisotropic character of the rotating crack but it does not transfer spurious stresses across widely open cracks. The new model with transition to scalar damage (RC-SD) keeps the anisotropic character of the RCM but it does not transfer spurious stresses across widely open cracks.

Virgin material is modeled as isotropic linear elastic material (described by Young modulus and Poisson ratio). The onset of cracking begins, when principal stress reaches tensile strength. Further behavior is then determined by **linear** softening law, governed by principle of preserving of fracture energy G_f. For large elements, the tension strength can be artificially reduced to preserve fracture energy. The transition to scalar damage model takes place,

when the softening stress reaches the specified limit. Multiple cracks are allowed. The elastic unloading and reloading is assumed. In compression regime, this model correspond to isotropic linear elastic material. The model description and parameters are summarized in table 10.

Description	Smeared rotating crack model with transition to scalar damage - linear softening
Record Format	**RCSD** $d_{(rn)}$ # $E_{(rn)}$ # $n_{(rn)}$ # $Gf_{(rn)}$ # $Ft_{(rn)}$ # sdtransitioncoeff$_{(rn)}$ # tAlpha$_{(rn)}$ #
Parameters	- *num* material model number - *d* material density - *E* Young modulus - *n* Poisson ratio - *Gf* fracture energy - *Ft* tension strength - *sdtransitioncoeff* determines the transition from RC to SD model. Transition takes plase when ratio of current softening stress to tension strength is less than *sdtransitioncoeff* value - *tAlpha* thermal dilatation coefficient
Supported modes	3dMat, PlaneStress, PlaneStrain, 1dMat, 2dPlateLayer, 2dBeamLayer, 3dShellLayer

Table 10: RC-SD model for concrete - summary.

1.3.4 Smeared rotating crack model with transition to scalar damage - exponential softening

Implementation of smeared rotating crack model with transition to scalar damage with exponential softening law. The description and model summary (table 11) are the same as for the RC-SD model with linear softening law (see section 1.3.3).

Description	Smeared rotating crack model with transition to scalar damage - exponential softening
Record Format	**RCSDE** $d_{(rn)}$ # $E_{(rn)}$ # $n_{(rn)}$ # $Gf_{(rn)}$ # $Ft_{(rn)}$ # sdtransitioncoeff$_{(rn)}$ # tAlpha$_{(rn)}$ #

Table 11: RC-SD model for concrete - summary.

1.3.5 Nonlocal smeared rotating crack model with transition to scalar damage

Implementation of nonlocal version of smeared rotating crack model with transition to scalar damage. Improves the classical rotating model (see section 1.3.2) by introducing the transition to scalar damage model in later stages of tension softening. The improved RC-SD (see section 1.3.3) is further extended to a nonlocal formulation, which not only acts as a powerful localization limiter but

also alleviates mesh-induced directional bias. A special type of material instability arising due to negative shear stiffness terms in the rotating crack model is resolved by switching to SD mode. A bell shaped nonlocal averaging function is used.

Virgin material is modeled as isotropic linear elastic material (described by Young modulus and Poisson ratio). The onset of cracking begins, when principal stress reaches tensile strength. Further behavior is then determined by **exponential** softening law.

The transition to scalar damage model takes place, when the softening stress reaches the specified limit or when the loss of material stability due to negative shear stiffness terms that may arise in the standard RCM formulation, which takes place when the ratio of minimal shear coefficient in stiffness to bulk material shear modulus reaches the limit.

Multiple cracks are allowed. The elastic unloading and reloading is assumed. In compression regime, this model correspond to isotropic linear elastic material. The model description and parameters are summarized in table 12.

Description	Nonlocal smeared rotating crack model with transition to scalar damage for concrete
Record Format	**RCSDNL** $d_{(rn)}$ # $E_{(rn)}$ # $n_{(rn)}$ # $Ft_{(rn)}$ # sdtransitioncoeff$_{(rn)}$ # sdtransitioncoeff2$_{(rn)}$ # $r_{(rn)}$ # tAlpha$_{(rn)}$ #
Parameters	- *num* material model number - *d* material density - *E* Young modulus - *n* Poisson ratio - *ef* deformation corresponding to fully open crack - *Ft* tension strength - *sdtransitioncoeff* determines the transition from RC to SD model. Transition takes place when ratio of current softening stress to tension strength is less than *sdtransitioncoeff* value - *sdtransitioncoeff2* determines the transition from RC to SD model. Transition takes place when ratio of current minimal shear stiffness term to virgin shear modulus is less than *sdtransitioncoeff2* value - *r* parameter specifying the width of nonlocal averaging zone - *tAlpha* thermal dilatation coefficient - *regionMap* map indicating the regions (currently region is characterized by cross section number) to skip for nonlocal avaraging. The elements and corresponding IP are not taken into account in nonlocal averaging process if corresponding regionMap value is nonzero.
Supported modes	3dMat, PlaneStress, PlaneStrain, 1dMat, 2dPlateLayer, 2dBeamLayer, 3dShellLayer

Table 12: RC-SD-NL model for concrete - summary.

1.3.6 Isotropic damage model for tensile failure

This isotropic damage model assumes that the stiffness degradation is isotropic, i.e., stiffness moduli corresponding to different directions decrease proportionally and independently of the loading direction. The damaged stiffness tensor is expressed as $\boldsymbol{D} = (1-\omega)\boldsymbol{D}_e$ where ω is a scalar damage variable and $\boldsymbol{D}_e$ is the elastic stiffness tensor. The damage evolution law is postulated in an explicit form, relating the damage variable ω to the largest previously reached equivalent strain level, κ.

The equivalent strain, $\tilde{\varepsilon}$, is a scalar measure derived from the strain tensor. The choice of the specific expression for the equivalent strain affects the shape of the elastic domain in the strain space and plays a similar role to the choice of a yield condition in plasticity. The following definitions of equivalent strain are currently supported:

- **Mazars** (1984) definition based on norm of positive part of strain:

$$\tilde{\varepsilon} = \sqrt{\sum_{I=1}^{3} \langle \varepsilon_I \rangle^2}$$

  where $\langle \varepsilon_I \rangle$ are positive parts of principal values of the strain tensor $\boldsymbol{\varepsilon}$.

- Definition derived from the **Rankine** criterion of maximum principal stress:

$$\tilde{\varepsilon} = \frac{1}{E} \sqrt{\sum_{I=1}^{3} \langle \bar{\sigma}_I \rangle^2}$$

  where $\langle \bar{\sigma}_I \rangle$ are the positive parts of principal values of the effective stress tensor $\bar{\boldsymbol{\sigma}} = \boldsymbol{D}_e : \boldsymbol{\varepsilon}$.

- **Energy** norm scaled by Young's modulus to obtain a strain-like quantity:

$$\tilde{\varepsilon} = \sqrt{\frac{\boldsymbol{\varepsilon} : \boldsymbol{D}_e : \boldsymbol{\varepsilon}}{E}}$$

- **Modified Mises** definition, proposed by de Vree et al. (1995):

$$\tilde{\varepsilon} = \frac{(k-1) I_{1\varepsilon}}{2k(1-2\nu)} + \frac{1}{2k} \sqrt{\frac{(k-1)^2}{(1-2\nu)^2} I_{1\varepsilon}^2 + \frac{12k J_{2\varepsilon}}{(1+\nu)^2}}$$

  where

$$I_{1\varepsilon} = \sum_{I=1}^{3} \varepsilon_I$$

  is the first strain invariant (trace of the strain tensor),

$$J_{2\varepsilon} = \frac{1}{2} \sum_{I=1}^{3} \varepsilon_I^2 - \frac{1}{6} I_{1\varepsilon}^2$$

  is the second deviatoric strain invariant, and k is a model parameter that corresponds to the ratio between the uniaxial compressive strength f_c and uniaxial tensile strength f_t.

Note that all these definitions are based on the three-dimensional description of strain (and stress). If they are used in a reduced problem, the strain components that are not explicitly provided by the finite element approximation are computed from the underlying assumptions and used in the evaluation of equivalent strain. For instance, in a plane-stress analysis, the out-of-plane component of normal strain is calculated from the assumption of zero out-of-plane normal stress (using standard Hooke's law).

Since the growth of damage usually leads to softening and may induce localization of the dissipative process, attention should be paid to proper regularization. The most efficient approach is based on a nonlocal formulation; see Section 1.3.7. If the model is kept local, the damage law should be adjusted according to the element size, in the spirit of the crack-band approach. When done properly, this ensures a correct dissipation of energy in a localized band of cracking elements, corresponding to the fracture energy of the material. For various numerical studies, it may be useful to specify the parameters of the damage law directly, independently of the element size. One should be aware that in this case the model would exhibit pathological sensitivity to the size of finite elements if the mesh is changed.

The following damage laws are currently implemented:

- **Cohesive crack with exponential softening** postulates a relation between the normal stress σ transmitted by the crack and the crack opening w in the form

$$\sigma = f_t \exp\left(-\frac{w}{w_f}\right)$$

  Here, f_t is the tensile strength and w_f is a parameter with the dimension of length (crack opening), which controls the ductility of the material. In fact, $w_f = G_f/f_t$ where G_f is the mode-I fracture energy. In the context of the crack-band approach, the crack opening w corresponds to the inelastic (cracking) strain ε_c multiplied by the effective thickness h of the crack band. The effective thickness h is estimated by projecting the finite element onto the direction of the maximum principal strain (and stress) at the onset of damage. The inelastic strain ε_c is the difference between the total strain ε and the elastic strain σ/E. For the damage model we obtain

$$\varepsilon_c = \varepsilon - \frac{\sigma}{E} = \varepsilon - (1-\omega)\varepsilon = \omega\varepsilon$$

  and thus $w = h\varepsilon_c = h\omega\varepsilon$. Substituting this into the cohesive law and combining with the stress-strain law for the damage model, we get a nonlinear equation

$$(1-\omega)E\varepsilon = f_t \exp\left(-\frac{h\omega\varepsilon}{w_f}\right)$$

  For a given strain ε, the corresponding damage variable ω can be solved from this equation by Newton iterations. It can be shown that the solution exists and is unique for every $\varepsilon \geq \varepsilon_0$ provided that the element size h does not exceed the limit size $h_{max} = w_f/\varepsilon_0$. For larger elements, a local snapback in the stress-strain diagram would occur, which is not admissible. In terms of the material properties, h_{max} can be expressed as EG_f/f_t^2, which is related to Irwin's characteristic length.

The derivation has been performed for monotonic loading and uniaxial tension. Under general conditions, ε is replaced by the internal variable κ, which represents the maximum previously reached level of equivalent strain.

In the list of input variables, the tensile strength f_t is not specified directly but through the corresponding strain at peak stress, $\varepsilon_0 = f_t/E$, denoted by keyword *e0*. Another input parameter is the characteristic crack opening w_f, denoted by keyword *wf*.

- **Cohesive crack with linear softening** is based on the same correspondence between crack opening and inelastic strain, but the cohesive law is assumed to have a simpler linear form

$$\sigma = f_t \left(1 - \frac{w}{w_f}\right)$$

The relation between damage and strain can then be described explicitly by the formula

$$\omega = \frac{1 - \dfrac{\varepsilon_0}{\varepsilon}}{1 - \dfrac{h\varepsilon_0}{w_f}}$$

and no iteration is needed for damage evaluation. Parameter w_f, denoted again by keyword *wf*, has now the meaning of crack opening at complete failure (zero cohesive stress) and is related to fracture energy by a modified formula $w_f = 2G_f/f_t$. The expression for maximum element size, $h_{max} = w_f/\varepsilon_0$, remains the same as for cohesive law with exponential softening, but in terms of the material properties it is now translated as $h_{max} = 2EG_f/f_t^2$.

- **Linear softening stress-strain law** works directly with strain and does not make any adjustment for the element size. The specified parameters ε_0 and ε_f, denoted by keywords *e0* and *ef*, have the meaning of (equivalent) strain at peak stress and at complete failure. The linear relation between stress and strain on the softening branch is obtained with the damage law

$$\omega = \frac{\varepsilon_f}{\varepsilon_f - \varepsilon_0}\left(1 - \frac{\varepsilon_0}{\varepsilon}\right)$$

Again, to cover general conditions, ε is replaced by κ.

- **Exponential softening stress-strain law** also uses two parameters ε_0 and ε_f, denoted by keywords *e0* and *ef*, but leads to a modified dependence of damage on strain:

$$\omega = 1 - \frac{\varepsilon_0}{\varepsilon}\exp\left(-\frac{\varepsilon - \varepsilon_0}{\varepsilon_f - \varepsilon_0}\right)$$

- **Mazars stress-strain law** uses three parameters, ε_0, A_t and B_t, denoted by keywords *e0*, *At* and *Bt*, and the dependence of damage on strain is given by

$$\omega = 1 - \frac{(1 - A_t)\varepsilon_0}{\varepsilon} - A_t \exp\left(B_t(\varepsilon - \varepsilon_0)\right)$$

- **Smooth exponential stress-strain law** uses two parameters, ε_0 and M_d, denoted by keywords *e0* and *md*, and the dependence of damage on strain is given by

$$\omega = 1 - \exp\left(-\left(\frac{\varepsilon}{\varepsilon_0}\right)^{M_d}\right)$$

  This leads to a stress-strain curve that immediately deviates from linearity (has no elastic part) and smoothly changes from hardening to softening, with tensile strength

$$f_t = E\varepsilon_0 \left(\mathrm{e}M_d\right)^{-1/M_d}$$

Note that parameter *damlaw* determines which type of damage law should be used, but the adjustment for element size is done only if parameter *wf* is specified for *damlaw*=0 or *damlaw*=1. For other values of *damlaw*, or if parameter *ef* is specified instead of *wf*, the stress-strain curve does not depend on element size and the model would exhibit pathological sensitivity to the mesh size. These cases are intended to be used in combination with a nonlocal formulation.

The model parameters are summarized in Table 13.

1.3.7 Nonlocal isotropic damage model for tensile failure

Nonlocal version of isotropic damage model from Section 1.3.6. The nonlocal averaging acts as a powerful localization limiter. In the standard version of the model, damage is driven by the nonlocal equivalent strain $\bar{\varepsilon}$, defined as a weighted average of the local equivalent strain:

$$\bar{\varepsilon}(\boldsymbol{x}) = \int_V \alpha(\boldsymbol{x}, \boldsymbol{\xi})\tilde{\varepsilon}(\boldsymbol{\xi})\, \mathrm{d}\boldsymbol{\xi}$$

In the "undernonlocal" formulation, the damage-driving variable is a combination of local and nonlocal equivalent strain, $m\bar{\varepsilon} + (1-m)\tilde{\varepsilon}$, where m is a parameter between 0 and 1. (If $m > 1$, the formulation is called "overnonlocal"; this case is useful for nonlocal plasticity but not for nonlocal damage.)

Instead of averaging the equivalent strain, one can average the compliance variable γ, directly related to damage according to the formula $\gamma = \omega/(1-\omega)$.

The weight function α contains a certain parameter with the dimension of length, which is in general called the characteristic length. Its specific meaning depends on the type of weight function. The following functions are currently supported:

- Truncated quartic spline, also called the bell-shaped function,

$$\alpha_0(s) = \left\langle 1 - \frac{s^2}{R^2} \right\rangle^2$$

  where R is the interaction radius (characteristic length) and s is the distance between the interacting points. This function is exactly zero for $s \geq R$, i.e., it has a bounded support.

- Gaussian function

$$\alpha_0(s) = \exp\left(-\frac{s^2}{R^2}\right)$$

which is theoretically nonzero for an arbitrary large s and thus has an unbounded support. However, in the numerical implementation the value of α_0 is considered as zero for $s > 2.5R$.

- Exponential function
$$\alpha_0(s) = \exp\left(-\frac{s}{R}\right)$$
which also has an unbounded support, but is considered as zero for $s > 6R$. This function is sometimes called the Green function, because in 1D it corresponds to the Green function of the Helmholtz-like equation used by implicit gradient approaches.

- Piecewise constant function
$$\alpha_0(s) = \begin{cases} 1 & \text{if } s \leq R \\ 0 & \text{if } s > R \end{cases}$$
which corresponds to uniform averaging over a segment, disc or ball of radius R.

- Function that is constant over the finite element in which point $\boldsymbol{x}$ is located, and is zero everywhere else. Of course, this is not a physically objective definition of nonlocal averaging, since it depends on the discretization. However, this kind of averaging was proposed in a boundary layer by Prof. Bažant and was implemented into OOFEM for testing purposes.

The above functions depend only on the distance s between the interacting points and are not normalized. If the normalizing condition
$$\int_{V_\infty} \alpha(\boldsymbol{x}, \boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{\xi} = 1$$
is imposed in an infinite body V_∞, it is sufficient to scale α_0 by a constant and set
$$\alpha(\boldsymbol{x}, \boldsymbol{\xi}) = \frac{\alpha_0(\|\boldsymbol{x} - \boldsymbol{\xi}\|)}{V_{r\infty}}$$
where
$$V_{r\infty} = \int_{V_\infty} \alpha_0(\|\boldsymbol{\xi}\|) \, \mathrm{d}\boldsymbol{\xi}$$
Constant $V_{r\infty}$ can be computed analytically depending on the specific type of weight function and the number of spatial dimensions in which the analysis is performed. Since the factor $1/V_{r\infty}$ can be incorporated directly in the definition of α_0, this case is referred to as "no scaling".

If the body of interest is finite (or even semi-infinite), the averaging integral can be performed only over the domain filled by the body, and the volume contributing to the nonlocal average at a point $\boldsymbol{x}$ near the boundary is reduced as compared to points $\boldsymbol{x}$ far from the boundary or in an infinite body. To make sure that the normalizing condition
$$\int_{V} \alpha(\boldsymbol{x}, \boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{\xi} = 1$$

holds for the specific domain V, different approaches can be used. The standard approach defines the nonlocal weight function as

$$\alpha(\boldsymbol{x}, \boldsymbol{\xi}) = \frac{\alpha_0(\|\boldsymbol{x} - \boldsymbol{\xi}\|)}{V_r(\boldsymbol{x})}$$

where

$$V_r(\boldsymbol{x}) = \int_V \alpha_0(\|\boldsymbol{x} - \boldsymbol{\xi}\|) \, \mathrm{d}\boldsymbol{\xi}$$

According to the approach suggested by Borino, the weight function is defined as

$$\alpha(\boldsymbol{x}, \boldsymbol{\xi}) = \frac{\alpha_0(\|\boldsymbol{x} - \boldsymbol{\xi}\|)}{V_{r\infty}} + \left(1 - \frac{V_r(\boldsymbol{x})}{V_{r\infty}}\right) \delta(\boldsymbol{x} - \boldsymbol{\xi})$$

where δ is the Dirac distribution. One can also say that the nonlocal variable is evaluated as

$$\bar{\varepsilon}(\boldsymbol{x}) = \frac{1}{V_{r\infty}} \int_V \alpha_0(\|\boldsymbol{x} - \boldsymbol{\xi}\|) \tilde{\epsilon}(\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{\xi} + \left(1 - \frac{V_r(\boldsymbol{x})}{V_{r\infty}}\right) \tilde{\varepsilon}(\boldsymbol{x})$$

The term on the right-hand side after the integral is a multiple of the local variable, and so it can be referred to as the local complement.

The model parameters are summarized in Tables 14 and 15.

Description	Isotropic damage model for concrete in tension
Record Format	**idm1** (in) # d(rn) # E(rn) # n(rn) # [tAlpha(rn) #] [equivstraintype(in) #] [k(rn) #] [damlaw(in) #] e0(rn) # [wf(rn) #] [ef(rn) #] [At(rn) #] [Bt(rn) #] [md(rn) #] [maxOmega(rn) #]
Parameters	- material number - *d* material density - *E* Young's modulus - *n* Poisson's ratio - *tAlpha* thermal expansion coefficient - *equivstraintype* allows to choose from different definitions of equivalent strain: 0 - Mazars (default) 1 - Rankine 2 - scaled energy norm 3 - modified Mises - *k* ratio between uniaxial compressive and tensile strength, needed only if equivstraintype=3, default value 1 - *damlaw* allows to choose from different damage laws: 0 - exponential softening (default) with parameters e0 and wf | ef 1 - linear softening with parameters e0 and wf | ef 2 - bilinear softening (not implemented yet) 3 - Hordijk softening (not implemented yet) 4 - Mazars damage law with parameters At and Bt 5 - smooth stress-strain curve with parameters e0 and md - *e0* strain at peak stress (for damage laws 0,1,2,3), limit elastic strain (for damage law 4), characteristic strain (for damage law 5) - *wf* parameter controling ductility, has the meaning of crack opening (for damage laws 0 and 1) - *ef* parameter controling ductility, has the meaning of strain (for damage laws 0 and 1) - *At* parameter of Mazars damage law, used only by law 4 - *Bt* parameter of Mazars damage law, used only by law 4 - *md* exponent, used only by damage law 5, default value 1 - *maxOmega* maximum damage, used for convergence improvement (its value is between 0 and 0.999999 (default), and it affects only the secant stiffness but not the stress)
Supported modes	3dMat, PlaneStress, PlaneStrain, 1dMat
Features	Adaptivity support

Table 13: Isotropic damage model for tensile failure – summary.

Description	Nonlocal isotropic damage model for concrete in tension
Record Format	**idmnl1** (in) # d(rn) # E(rn) # n(rn) # [tAlpha(rn) #] [equivstraintype(in) #] [k(rn) #] [damlaw(in) #] e0(rn) # [ef(rn) #] [At(rn) #] [Bt(rn) #] [md(rn) #] r(rn) # [regionMap(ia) #] [wft(in) #] [averagingType(in) #] [m(rn) #] [scaling(in) #] [averagedVar(in) #] [maxOmega(rn) #]
Parameters	- material number - d material density - E Young's modulus - n Poisson's ratio - *tAlpha* thermal expansion coefficient - *equivstraintype* allows to choose from different definitions of equivalent strain, same as for the local model; see Table 13 - k ratio between uniaxial compressive and tensile strength, needed only if equivstraintype=3, default value 1 - *damlaw* allows to choose from different damage laws, same as for the local model; see Table 13 (note that parameter *wf* cannot be used for the nonlocal model) - *e0* strain at peak stress (for damage laws 0,1,2,3), limit elastic strain (for damage law 4), characteristic strain (for damage law 5) - *ef* strain parameter controling ductility, has the meaning of strain (for damage laws 0 and 1), the tangent modulus just after the peak is $E_t = -f_t/(\varepsilon_f - \varepsilon_0)$ - *At* parameter of Mazars damage law, used only by law 4 - *Bt* parameter of Mazars damage law, used only by law 4 - *md* exponent, used only by damage law 5, default value 1 - r nonlocal characteristic length R; its meaning depends on the type of weight function (e.g., interaction radius for the quartic spline) - *regionMap* map indicating the regions (currently region is characterized by cross section number) to skip for nonlocal avaraging. The elements and corresponding IP are not taken into account in nonlocal averaging process if corresponding regionMap value is nonzero. - *wft* selects the type of nonlocal weight function: 1 - default, quartic spline (bell-shaped function with bounded support) 2 - Gaussian function 3 - exponential function (Green function in 1D) 4 - uniform averaging up to distance R 5 - uniform averaging over one finite element — *continued in Table 15* —

Table 14: Nonlocal isotropic damage model for tensile failure – summary.

Description	Nonlocal isotropic damage model for concrete in tension
	- *averagingType* activates a special averaging procedure, default value 0 does not change anything, value 1 means averaging over one finite element (equivalent to *wft*=5, but kept here for compatibility with previous version) - m multiplier for overnonlocal or undernonlocal formulation, which use m-times the local variable plus $(1-m)$-times the nonlocal variable, default value 1 - *scaling* selects the type of scaling of the weight function (e.g. near a boundary): 1 - default, standard scaling with integral of weight function in the denominator 2 - no scaling (the weight function normalized in an infinite body is used even near a boundary) 3 - Borino scaling (local complement) - *averagedVar* selects the variable to be averaged, default value 1 corresponds to equivalent strain, value 2 activates averaging of compliance variable - *maxOmega* maximum damage, used for convergence improvement (its value is between 0 and 0.999999 (default), and it affects only the secant stiffness but not the stress)
Supported modes	3dMat, PlaneStress, PlaneStrain, 1dMat
Features	Adaptivity support

Table 15: Nonlocal isotropic damage model for tensile failure – continued.

1.3.8 MDM - Anisotropic damage model

Local formulation The concept of isotropic damage is appropriate for materials weakened by voids, but if the physical source of damage is the initiation and propagation of microcracks, isotropic stiffness degradation can be considered only as a first rough approximation. More refined damage models take into account the highly oriented nature of cracking, which is reflected by the anisotropic character of the damaged stiffness or compliance matrices.

A number of anisotropic damage formulations have been proposed in the literature. Here we use a model outlined by M. Jirásek in [5], which is based on the principle of energy equivalence and on the construction of the inverse integrity tensor by integration of a scalar over all spatial directions. Since the model uses certain concepts from the microplane theory, it is called the microplane-based damage model (MDM).

The general structure of the MDM model is schematically shown in Fig. 6 and the basic equations are summarized in Table 16. Here, $\boldsymbol{\varepsilon}$ and $\boldsymbol{\sigma}$ are the (nominal) second-order strain and stress tensors with components ε_{ij} and σ_{ij}; $\boldsymbol{e}$ and $\boldsymbol{s}$ are first-order strain and stress tensors with components e_i and s_i, which characterize the strain and stress on "microplanes" of different orientations given by a unit vector $\boldsymbol{n}$ with components n_i; ψ is a dimensionless compliance parameter that is a scalar but can have different values for different directions $\boldsymbol{n}$; the symbol δ denotes a virtual quantity; and a sumperimposed tilde denotes an effective quantity, which is supposed to characterize the state of the intact material between defects such as microcracks or voids.

Table 16: Basic equations of microplane-based anisotropic damage model

$\tilde{\boldsymbol{e}} = \tilde{\boldsymbol{\varepsilon}} \cdot \boldsymbol{n}$	$\boldsymbol{s}^T = \psi \boldsymbol{s}$	$\boldsymbol{s} = \boldsymbol{\sigma} \cdot \boldsymbol{n}$
$\tilde{\boldsymbol{\sigma}} : \delta\tilde{\boldsymbol{\varepsilon}} = \dfrac{3}{2\pi} \displaystyle\int_\Omega \boldsymbol{s}^T \cdot \delta\tilde{\boldsymbol{e}} \, \mathrm{d}\Omega$	$\delta \boldsymbol{s} \cdot \boldsymbol{e} = d\boldsymbol{s}^T \cdot \tilde{e}$	$\delta\boldsymbol{\sigma} : \boldsymbol{\varepsilon} = \dfrac{3}{2\pi} \displaystyle\int_\Omega \delta\boldsymbol{s} \cdot \boldsymbol{e} \, \mathrm{d}\Omega$
$\tilde{\boldsymbol{\sigma}} = \dfrac{3}{2\pi} \displaystyle\int_\Omega (\boldsymbol{s}^T \otimes \boldsymbol{n})_{\mathrm{sym}} \, \mathrm{d}\Omega$	$\boldsymbol{e} = \psi \tilde{\boldsymbol{e}}$	$\boldsymbol{\varepsilon} = \dfrac{3}{2\pi} \displaystyle\int_\Omega (\boldsymbol{e} \otimes \boldsymbol{n})_{\mathrm{sym}} \, \mathrm{d}\Omega$

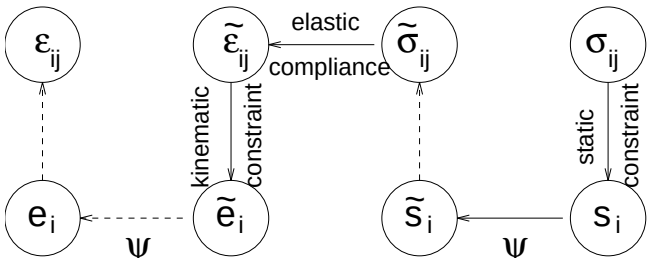

Figure 6: Structure of microplane-based anisotropic damage model

Combining the basic equations, it is possible to show that the components

of the damaged material compliance tensor are given by

$$C_{ijkl} = M_{pqij} M_{rskl} C^e_{pqrs} \tag{44}$$

where C^e_{pqrs} are the components of the elastic material compliance tensor,

$$M_{ijkl} = \tfrac{1}{4} \left(\psi_{ik}\delta_{jl} + \psi_{il}\delta_{jk} + \psi_{jk}\delta_{il} + \psi_{jl}\delta_{ik} \right) \tag{45}$$

are the components of the so-called damage effect tensor, and

$$\psi_{ij} = \frac{3}{2\pi} \int_\Omega \psi \, n_i n_j \, \mathrm{d}\Omega \tag{46}$$

are the components of the second-order inverse integrity tensor. The integration domain Ω is the unit hemisphere. In practice, the integral over the unit hemisphere is evaluated by summing the contribution from a finite number of directions, according to one of the numerical integration schemes that are used by microplane models.

The scalar variable ψ characterizes the relative compliance in the direction given by the vector $\boldsymbol{n}$. If ψ is the same in all directions, the inverse integrity tensor evaluated from (46) is equal to the unit second-order tensor (Kronecker delta) multiplied by ψ, the damage effect tensor evaluated from (45) is equal to the symmetric fourth-order unit tensor multiplied by ψ, and the damaged material compliance tensor evaluated from (44) is the elastic compliance tensor multiplied by ψ^2. The factor multiplying the elastic compliance tensor in the isotropic damage model is $1/(1-\omega)$, and so ψ corresponds to $1/\sqrt{1-\omega}$. In the initial undamaged state, $\psi = 1$ in all directions. The evolution of ψ is governed by the history of the projected strain components. In the simplest case, ψ is driven by the normal strain $e_N = \boldsymbol{\varepsilon}_{ij} n_i n_j$. Analogy with the isotropic damage model leads to the damage law

$$\psi = f(\kappa) \tag{47}$$

and loading-unloading conditions

$$g(e_N, \kappa) \equiv e_N - \kappa \leq 0, \qquad \dot{\kappa} \geq 0, \qquad \dot{\kappa} g(e_N, \kappa) = 0 \tag{48}$$

in which κ is a history variable that represents the maximum level of normal strain in the given direction ever reached in the previous history of the material. An appropriate modification of the exponential softening law leads to the damage law

$$f(\kappa) = \begin{cases} 1 & \text{if } \kappa \leq e_0 \\ \sqrt{\frac{\kappa}{e_0} \exp\left(\frac{\kappa - e_0}{e_f - e_0} \right)} & \text{if } \kappa > e_0 \end{cases} \tag{49}$$

where e_0 is a parameter controlling the elastic limit, and $e_f > e_0$ is another parameter controlling ductility. Note that softening in a limited number of directions does not necessarily lead to softening on the macroscopic level, because the response in the other directions remains elastic. Therefore, e_0 corresponds to the elastic limit but not to the state at peak stress.

If the MDM model is used in its basic form described above, the compressive strength turns out to depend on the Poisson ratio and, in applications to

concrete, its value is too low compared to the tensile strength. The model is designed primarily for tensile-dominated failure, so the low compressive strength is not considered as a major drawback. Still, it is desirable to introduce a modification that would prevent spurious compressive failure in problems where moderate compressive stresses appear. The desired effect is achieved by redefining the projected strain e_N as

$$e_N = \frac{\boldsymbol{\varepsilon}_{ij} n_i n_j}{1 - \dfrac{m}{E e_0} \sigma_{kk}} \tag{50}$$

where m is a nonnegative parameter that controls the sensitivity to the mean stress, σ_{kk} is the trace of the stress tensor, and the normalizing factor Ee_0 is introduced in order to render the parameter m dimensionless. Under compressive stress states (characterized by $\sigma_{kk} < 0$), the denominator in (50) is larger than 1, and the projected strain is reduced, which also leads to a reduction of damage. A typical recommended value of parameter m is 0.05.

Nonlocal formulation Nonlocal formulation of the MDM model is based on the averaging of the inverse integrity tensor. This roughly corresponds to the nonlocal isotropic damage model with averaging of the compliance variable $\gamma = \omega/(1-\omega)$, which does not cause any spurious locking effects. In equation (45) for the evaluation of the damage effect tensor, the inverse integrity tensor is replaced by its weighted average with components

$$\bar{\psi}_{ij}(\boldsymbol{x}) = \int_V \alpha(\boldsymbol{x}, \boldsymbol{\xi}) \psi_{ij}(\boldsymbol{\xi}) \mathrm{d}\boldsymbol{\xi} \tag{51}$$

By fitting a wide range of numerical results, it has been found that the parameters of the nonlocal MDM model can be estimated from the measurable material properties using the formulas

$$\lambda_f = \frac{E G_f}{R f_t^2} \tag{52}$$

$$\lambda = \frac{\lambda_f}{1.47 - 0.0014 \lambda_f} \tag{53}$$

$$e_0 = \frac{f_t}{(1-m) E (1.56 + 0.006\lambda)} \tag{54}$$

$$e_f = e_0 [1 + (1-m)\lambda] \tag{55}$$

where E is Young's modulus, G_f is the fracture energy, f_t is the uniaxial tensile strength, m is the compressive correction factor, typically chosen as $m = 0.05$, and R is the radius of nonlocal interaction reflecting the internal length of the material.

Input Record The model description and parameters are summarized in table 17.

Description	MDM Anisotropic damage model
Record Format Parameters	Common parameters **mdm** d$_{(rn)}$ # nmp$_{(ins)}$ # talpha$_{(rn)}$ # parmd$_{(rn)}$ # nonloc$_{(in)}$ # formulation$_{(in)}$ # mode$_{(in)}$ # -*num* material model number - *D* material density - *nmp* number of microplanes used for hemisphere integration, supported values are 21,28, and 61 - *talpha* thermal dillatation coeff - *parmd* - *nonloc* - *formulation* - *mode*
Nonlocal variant I Additional params	 r$_{(rn)}$ # efp$_{(rn)}$ # ep$_{(rn)}$ # -*r* nonlocal interaction radius -*efp* $\varepsilon_f p$ is a model parameter that controls the post-peak slope $\varepsilon_f p = \varepsilon_f - \varepsilon_0$, where ε_f is strain at zero stress level. -*ep* max effective strain at peak ε_0
Nonlocal variant II Additional params	 r$_{(rn)}$ # gf$_{(rn)}$ # ft$_{(rn)}$ # -*r* nonlocal intraction radius -*gf* fracture energy -*ft* tensile strength
Local variant I Additional params	 efp$_{(rn)}$ # ep$_{(rn)}$ # -*efp* $\varepsilon_f p$ is a model parameter that controls the post-peak slope $\varepsilon_f p = \varepsilon_f - \varepsilon_0$, where ε_f is strain at zero stress level. -*ep* max effective strain at peak ε_0
Local variant II Additional params	 gf$_{(rn)}$ # ep$_{(rn)}$ # -*gf* fracture energy -*ep* max effective strain at peak ε_0
Supported modes Features	3dMat, PlaneStress Adaptivity support

Table 17: MDM model - summary.

1.4 Material models specific to concrete

1.4.1 Mazars damage model for concrete

This isotropic damage model assumes that the stiffness degradation is isotropic, i.e., stiffness moduli corresponding to different directions decrease proportionally and independently of direction of loading. It introduces two damage parameters ω_t and ω_c that are computed from the same equivalent strain using two different damage functions g_t and g_c. The g_t is identified from the uniaxial tension tests, while g_c from compressive test. The damage parameter for general stress states ω is obtained as a linear combination of ω_t and ω_c: $\omega = \alpha_t g_t + \alpha_c g_c$, where the coefficients α_t and α_c take into account the character of the stress state. The damaged stiffness tensor is expressed as $\boldsymbol{D} = (1-\omega)\boldsymbol{D}^e$. Damage evolution law is postulated in an explicit form, relating damage parameter and scalar measure of largest reached strain level in material, taking into account the principle of preserving of fracture energy G_f. The equivalent strain, i.e., a scalar measure of the strain level is defined as norm from positive principal strains. The model description and parameters are summarized in table 18.

1.4.2 Nonlocal Mazars damage model for concrete

The nonlocal variant of Mazars damage model for concrete. Model based on nonlocal averaging of equivalent strain. The nonlocal averaging acts as a powerful localization limiter. The bell-shaped nonlocal averaging function is used. The model description and parameters are summarized in table 19.

1.4.3 CebFip78 model for concrete creep with aging

Implementation of aging viscoelastic model for concrete creep according to the CEB-FIP Model Code. The model parameters are summarized in table 20.

1.4.4 Double-power law model for concrete creep with aging

Implementation of aging viscoelastic model for concrete creep with compliance function given by the double-power law. The model parameters are summarized in table 21.

1.4.5 B3 model for concrete creep with aging

Model B3 is an aging viscoelastic model for concrete creep and shrinkage, developed by Prof. Bažant and coworkers. In OOFEM it is implemented in two different forms. Material "B3mat" uses an approximation by an aging Maxwell chain. This is an older version, kept in OOFEM for compatibility. A more recent implementation is referred to as "B3solidmat" and exploits a non-aging Kelvin chain combined with the solidification theory. It is also extended to the so-called microprestress solidification theory (MPS), which takes into account the effects of variable humidity (and in the future also of variable temperature) on creep. The underlying rheological chain consists of three major components. The solidifying Kelvin chain represents short-term creep; it is serially coupled with a non-aging elastic spring that reflects instantaneous deformation. Long-term creep is captured by an aging dashpot with viscosity dependent on the

Description	Mazars damage model for concrete
Record Format	**mazarsmodel** $d_{(rn)}$ # $E_{(rn)}$ # $n_{(rn)}$ # $e0_{(rn)}$ # $ac_{(rn)}$ # [$bc_{(rn)}$ #] [$beta_{(rn)}$ #] $at_{(rn)}$ # [$bt_{(rn)}$ #] [$hreft_{(rn)}$ #] [$hrefc_{(rn)}$ #] [$version_{(in)}$ #] [$tAlpha_{(rn)}$ #] [equivstrain-$type_{(in)}$ #] [$maxOmega_{(rn)}$ #]
Parameters	- *num* material model number - *d* material density - *E* Young modulus - *n* Poisson ratio - *e0* max effective strain at peak - *ac*,*bc* material parameters related to the shape of uniaxial compression curve (A sample set used by Saouridis is $A_c = 1.34, B_c = 2537$ - *beta* coefficient reducing the effect of damage under response under shear. Default value set to 1.06 - *at*, [*bt*] material parameters related to the shape of uniaxial tension curve. Meaning dependent on *version* parameter. - *hreft*, *hrefc* reference characteristic lengths for tension and compression. The material parameters are specified for element with these characteristic lengths. The current element then will have the same COD (Crack Opening Displacement) as reference one. - *version* Model variant. if 0 specified, the original form $g_t = 1.0 - (1.0 - A_t) * \varepsilon_0/\kappa - A_t * \exp(-B_t * (\kappa - \varepsilon_0))$; of tension damage evolution law is used, if equal 1, the modified law used which asymptotically tends to zero $g_t = 1.0 - (\varepsilon_0/\kappa) * \exp((\varepsilon_0 - \kappa)/A_t)$ - *tAlpha* thermal dilatation coefficient - *equivstraintype* see Table 13 - *maxOmega* limit maximum damage, use for convergency improvement
Supported modes	3dMat, PlaneStress, PlaneStrain, 1dMat

Table 18: Mazars damage model – summary.

microprestress, the evolution of which is affected by changes of temperature and humidity.

The model description and parameters are summarized in table 22 for "B3mat" and in table 23 for "B3solidmat". Since some model parameters are determined from the composition and strength using empirical formulae, it is necessary to use the specified units (e.g. compressive strength always in MPa, irrespectively of the units used in the simulation for stress).

Description	Nonlocal Mazars damage model for concrete
Record Format	**mazarsmodelnl** r(rn) # E(rn) # n(rn) # e0(rn) # ac(rn) # bc(rn) # beta(rn) # version(in) # at(rn) # [bt(rn) #] r(rn) # tAlpha(rn) #
Parameters	- *num* material model number - *d* material density - *E* Young modulus - *n* Poisson ratio - *maxOmega* limit maximum damage, use for convergency improvement - *tAlpha* thermal dilatation coefficient - *version* Model variant. if 0 specified, the original form $g_t = 1.0 - (1.0 - A_t) * \varepsilon_0/\kappa - A_t * \exp(-B_t * (\kappa - \varepsilon_0))$; of tension damage evolution law is used, if equal 1, the modified law used which asymptotically tends to zero $g_t = 1.0 - (\varepsilon_0/\kappa) * \exp((\varepsilon_0 - \kappa)/A_t)$ - *ac*,*bc* material parameters related to the shape of uniaxial compression curve (A sample set used by Saouridis is $A_c = 1.34, B_c = 2537$ - *at*, [*bt*] material parameters related to the shape of uniaxial tension curve. Meaning dependent on *version* parameter. - *beta* coefficient reducing the effect of damage under response under shear. Default value set to 1.06 - *r* parameter specifying the width of nonlocal averaging zone
Supported modes	3dMat, PlaneStress, PlaneStrain, 1dMat

Table 19: Nonlocal Mazars damage model – summary.

Description	CebFip78 model for concrete creep with aging
Record Format	**CebFip78** n(rn) # relMatAge(rn) # E28(rn) # fibf(rn) # kap_a(rn) # kap_c(rn) # kap_tt(rn) # u(rn) #
Parameters	- *num* material model number - *E28* Young modulus at age of 28 days [MPa] - *n* Poisson ratio - *fibf* basic creep coefficient - *kap_a* coefficient of hydrometric conditions - *kap_c* coefficient of type of cement - *kap_tt* coeficient of temperature effects - *u* surface imposed to environment $[mm^2]$; temporary here; should be in crosssection level - *relmatage* relative material age
Supported modes	3dMat, PlaneStress, PlaneStrain, 1dMat, 2dPlateLayer,2dBeamLayer, 3dShellLayer

Table 20: CebFip78 material model – summary.

Description	Double-power law model for concrete creep with aging
Record Format	**doublepowerlaw** n(rn) # relMatAge(rn) # E28(rn) # fi1(rn) # m(rn) # n(rn) # alpha(rn) #
Parameters	- *num* material model number - *E28* Young modulus at age of 28 days [MPa] - *n* Poisson ratio - *fibf* basic creep coefficient - *m* coefficient - *n* coefficient - *alpha* coeficient - *relmatage* relative material age
Supported modes	3dMat, PlaneStress, PlaneStrain, 1dMat, 2dPlateLayer,2dBeamLayer, 3dShellLayer

Table 21: Double-power law model – summary.

Description	B3 material model for concrete aging
Record Format	**B3mat** n(rn) # relMatAge(rn) # fc(rn) # cc(rn) # w/c(rn) # a/c(rn) # t0(rn) # alpha1(rn) # alpha2(rn) # ks(rn) # hum(rn) # vs(rn) # noshrinkage(in) #
Parameters	- *num* material model number - *n* Poisson ratio - *relmatage* relative material age - *fc* 28-day mean cylinder compression strength in MPa - *cc* cement content of concrete in kg/m^3 - *w/c* ratio (by weight) of water to cementitious material - *a/c* ratio (by weight) of aggregate to cement - *t0* age when drying begins [day] - *alpha1* shrinkage parameter – influence of cement type - *alpha2* shrinkage parameter – influence of curing type - *ks* cross-section shape factor - *hum* relative humidity of the environment - *vs* volume to surface ratio [m] - *noshrinkage* shrinkage flag (0 = shrinkage included, 1 = shrinkage ignored)
Supported modes	3dMat, PlaneStress, PlaneStrain, 1dMat, 2dPlateLayer,2dBeamLayer, 3dShellLayer

Table 22: B3 creep and shrinkage model – summary.

Description	B3solid material model for concrete creep
Record Format	**B3solidmat** d(rn) # n(rn) # talpha(rn) # mode(in) # EmoduliMode(in) # Microprestress(in) # shm(in) # begoftimeofinterest(rn) # endoftimeofinterest(rn) # timefactor(rn) # relMatAge(rn) # fc(rn) # cc(rn) # w/c(rn) # a/c(rn) # t0(rn) # q1(rn) # q2(rn) # q3(rn) # q4(rn) # c0(rn) # c1(rn) # tS0(rn) # w_h(rn) # ncoeff(rn) # a(rn) # ks(rn) # alpha1(rn) # alpha2(rn) # hum(rn) # vs(rn) # q5(rn) # kt(rn) # EpsSinf(rn) # es0(rn) # r(rn) # rprime(rn) # at(rn) # kSh(rn) # inithum(rn) # finalhum(rn) #
Parameters	- *num* material model number - *d* material density - *n* Poisson ratio - *talpha* coefficient of thermal expansion - *mode* optional parameter; if *mode* = 0 (default), parameters $q1 - q4$ are predicted from composition of the concrete mixture (parameters fc, cc, w/c, a/c and t0 need to be specified). Otherwise values of parameters $q1 - q4$ are expected. - *EmoduliMode* optional parameter; analysis of retardation spectrum (= 0, default value) or least-squares method (= 1) is used for evaluation of Kelvin units moduli - *Microprestress* 0 = basic creep; 1 = drying creep (must be run as a staggered problem with preceding analysis of humidity diffusion. Parameter *shm* must be equal to 3. The following parameters must be specified: *c0*, *c1*, *tS0*, *w_h*, *ncoeff*, *a*) - *shm* shrinkage mode; 0 = no shrinkage; 1 = average shrinkage (the following parameters must be specified: *ks*, *alpha1*, *alpha2*, *hum* and *vs* for *mode* = 0 and *ks*, *q5*, *kt* and *EpsSinf* for *mode* = 1); 2 = point shrinkage (needed: *es0*, *r*, *rprime*, *at*); 3 = point shrinkage based on MPS theory (needed: parameter *kSh* or value of *kSh* can be approximately determined if following parameters are given: *inithum*, *finalhum*, *alpha1* and *alpha2*) - *begoftimeofinterest*, *endoftimeofinterest* optional parameters; lower and upper boundary of time interval with good approximation of the compliance function [day] - *timefactor* scaling factor transforming the simulation time units into days - *relMatAge* relative material age [day]

	- *fc* 28-day mean cylinder compression strength [MPa] - *cc* cement content of concrete mixture [kg/m^3] - *w/c* water to cement ratio (by weight) - *a/c* aggregate to cement ratio (by weight) - *t0* age of concrete when drying begins [day] - *q1*, *q2*, *q3*, *q4* parameters of B3 model for basic creep - *c0* MPS theory parameter [MPa^{-1} day^{-1}] - *c1* MPS theory parameter [MPa] - *tS0* MPS theory parameter - time when drying begins - *w_h*, *ncoeff*, *a* sorption isotherm parameters obtained from experiments [Pedersen, 1990] - *ks* cross section shape factor - *alpha1* optional shrinkage parameter - influence of cement type (optional parameter, default value is 1.0) - *alpha2* optional shrinkage parameter - influence of curing type (optional parameter, default value is 1.0) - *hum* relative humidity of the environment [-] - *vs* volume to surface ratio [m] - *q5*, *kt*, *EpsSinf* parameters of B3 model for drying creep - *es0* final shrinkage at material point - *at* coefficient relating stress-induced thermal strain and shrinkage - *rprime*, *r* coefficients - *kSh* influences magnitude of shrinkage in MPS theory - *inithum*, *finalhum* if provided, approximate value of *kSh* can be computed
Supported modes	3dMat, PlaneStress, PlaneStrain, 1dMat, 2dPlateLayer, 2dBeamLayer, 3dShellLayer

Table 23: B3solid creep and shrinkage model – summary.

1.4.6 Microplane model M4

Model M4 covers inelastic behavior of concrete under complex triaxial stress states. It is based on the microplane concept and can describe softening. However, objectivity with respect to element size is not ensured – the parameters need to be manually adjusted to the element size. Since the tangent stiffness matrix is not available, elastic stiffness is used. This can lead to a very slow convergence when used within an implicit approach. The model parameters are summarized in table 24.

Description	M4 material model
Record Format	**microplane_m4** $\text{nmp}_{(\text{in})}$ # $\text{c3}_{(\text{rn})}$ # $\text{c20}_{(\text{rn})}$ # $\text{k1}_{(\text{rn})}$ # $\text{k2}_{(\text{rn})}$ # $\text{k3}_{(\text{rn})}$ # $\text{k4}_{(\text{rn})}$ # $\text{E}_{(\text{rn})}$ # $\text{n}_{(\text{rn})}$ #
Parameters	- *nmp* number of microplanes, supported values are 21, 28 and 61 - *n* Poisson ratio - *E* Young modulus - *c3*,*c20*, *k1*, *k2*, *k3*, *k4* model parameters
Supported modes	3dMat

Table 24: Microplane model M4 – summary.

1.4.7 Damage-plastic model for concrete

This model, developed by Grassl and Jirásek for failure of concrete under general triaxial stress, is described in detail in [3]. It belongs to the class of damage-plastic models with yield condition formulated in terms of the effective stress $\bar{\boldsymbol{\sigma}} = \boldsymbol{D}_e : (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_p)$. The stress-strain law is postulated in the form

$$\boldsymbol{\sigma} = (1-\omega)\bar{\boldsymbol{\sigma}} = (1-\omega)\boldsymbol{D}_e : (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_p) \tag{56}$$

where $\boldsymbol{D}_e$ is the elastic stiffness tensor and ω is a scalar damage parameter. The plastic part of the model consists of a three-invariant yield condition, nonassociated flow rule and pressure-dependent hardening law. For simplicity, damage is assumed to be isotropic. In contrast to pure damage models with damage driven by the total strain, here the damage is linked to the evolution of plastic strain.

The **yield surface** is described in terms of the cylindrical coordinates in the principal effective stress space (Haigh-Westergaard coordinates), which are the volumetric effective stress $\bar{\sigma}_{\text{V}} = I_1(\bar{\boldsymbol{\sigma}})/3$, the norm of the deviatoric effective stress $\bar{\rho} = \sqrt{2J_2(\bar{\boldsymbol{\sigma}})}$, and the Lode angle θ defined by the relation

$$\cos 3\theta = \frac{3\sqrt{3}}{2}\frac{J_3}{J_2^{3/2}} \tag{57}$$

where J_2 and J_3 are the second and third deviatoric invariants. The yield function

$$\begin{aligned} f_{\text{p}}(\bar{\sigma}_{\text{V}}, \bar{\rho}, \bar{\theta}; \kappa_{\text{p}}) \quad = \quad & \left([1 - q_{\text{h}}(\kappa_{\text{p}})] \left(\frac{\bar{\rho}}{\sqrt{6}\bar{f}_c} + \frac{\bar{\sigma}_{\text{V}}}{\bar{f}_c} \right)^2 + \sqrt{\frac{3}{2}} \frac{\bar{\rho}}{\bar{f}_c} \right)^2 + \\ & + m_0 q_{\text{h}}^2(\kappa_{\text{p}}) \left(\frac{\bar{\rho} r(\bar{\theta})}{\sqrt{6}\bar{f}_c} + \frac{\bar{\sigma}_{\text{V}}}{\bar{f}_c} \right) - q_{\text{h}}^2(\kappa_{\text{p}}) \end{aligned} \tag{58}$$

depends on the effective stress (which enters in the form of cylindrical coordinates) and on the hardening variable $\kappa_{\rm p}$ (which enters through a dimensionless variable $q_{\rm h}$). Parameter $\bar{f}_c$ is the uniaxial compressive strength. Note that, under uniaxial compression characterized by axial stress $\bar{\sigma} < 0$, we have $\bar{\sigma}_{\rm V} = \bar{\sigma}/3$, $\bar{\rho} = -\sqrt{2/3}\,\bar{\sigma}$ and $\bar{\theta} = 60^o$. The yield function then reduces to $f_{\rm p} = (\bar{\sigma}/\bar{f}_c)^2 - q_{\rm h}^2$. This means that function $q_{\rm h}$ describes the evolution of the uniaxial compressive yield stress normalized by its maximum value, $\bar{f}_c$.

The evolution of the yield surface during hardening is presented in Fig. 7. The parabolic shape of the meridians (Fig. 7a) is controlled by the hardening variable $q_{\rm h}$ and the friction parameter m_0. The initial yield surface is closed, which allows modeling of compaction under highly confined compression. The initial and intermediate yield surfaces have two vertices on the hydrostatic axis but the ultimate yield surface has only one vertex on the tensile part of the hydrostatic axis and opens up along the compressive part of the hydrostatic axis. The deviatoric sections evolve as shown in Fig. 7b, and their final shape at full hardening is a rounded triangle at low confinement and almost circular at high confinement. The shape of the deviatoric section is controlled by the Willam-Warnke function

$$r(\theta) = \frac{4(1-e^2)\cos^2\theta + (2e-1)^2}{2(1-e^2)\cos\theta + (2e-1)\sqrt{4(1-e^2)\cos^2\theta + 5e^2 - 4e}} \tag{59}$$

The eccentricity parameter e that appears in this function, as well as the friction parameter m_0, are calibrated from the values of uniaxial and equibiaxial compressive strengths and uniaxial tensile strength.

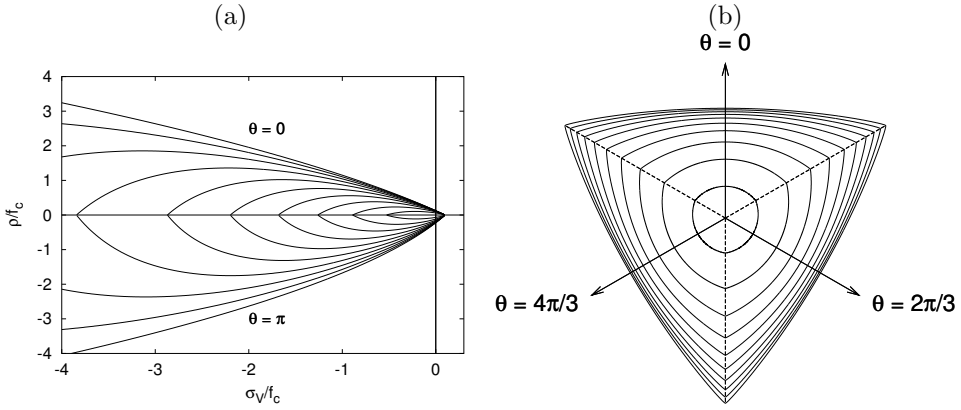

Figure 7: Evolution of the yield surface during hardening: a) meridional section, b) deviatoric section for a constant volumetric effective stress of $\bar{\sigma}_{\rm V} = -\bar{f}_c/3$

The maximum size of the elastic domain is attained when the variable $q_{\rm h}$ is equal to one (which is its maximum value, as follows from the hardening law, to be specified in (64)). The yield surface is then described by the equation

$$f_{\rm p}\left(\bar{\sigma}_{\rm V}, \bar{\rho}, \bar{\theta}; 1\right) \equiv \frac{3}{2}\frac{\bar{\rho}^2}{\bar{f}_c^2} + m_0\left(\frac{\bar{\rho}}{\sqrt{6}\bar{f}_c} r(\bar{\theta}) + \frac{\bar{\sigma}_{\rm V}}{\bar{f}_c}\right) - 1 = 0 \tag{60}$$

The **flow rule**

$$\dot{\boldsymbol{\varepsilon}}_{\rm p} = \dot{\lambda}\frac{\partial g_{\rm p}}{\partial \bar{\sigma}} \tag{61}$$

is non-associated, which means that the yield function f_p and the plastic potential

$$g_\mathrm{p}(\bar{\sigma}_\mathrm{V}, \bar{\rho}; \kappa_\mathrm{p}) = \left([1 - q_\mathrm{h}(\kappa_\mathrm{p})] \left(\frac{\bar{\rho}}{\sqrt{6}\bar{f}_c} + \frac{\bar{\sigma}_\mathrm{V}}{\bar{f}_c} \right)^2 + \sqrt{\frac{3}{2}} \frac{\bar{\rho}}{\bar{f}_c} \right)^2 + \\ + q_\mathrm{h}^2(\kappa_\mathrm{p}) \left(\frac{m_0 \bar{\rho}}{\sqrt{6}\bar{f}_c} + \frac{m_\mathrm{g}(\bar{\sigma}_\mathrm{V})}{\bar{f}_c} \right) \tag{62}$$

do not coincide and, therefore, the direction of the plastic flow $\partial g_\mathrm{p}/\partial \bar{\boldsymbol{\sigma}}$ is not normal to the yield surface. The ratio of the volumetric and the deviatoric parts of the flow direction is controled by function m_g, which depends on the volumetric stress and is defined as

$$m_\mathrm{g}(\bar{\sigma}_\mathrm{V}) = A_\mathrm{g} B_\mathrm{g} \bar{f}_c \exp \frac{\bar{\sigma}_\mathrm{V} - \bar{f}_t/3}{B_\mathrm{g} \bar{f}_c} \tag{63}$$

where A_g and B_g are model parameters that are determined from certain assumptions on the plastic flow in uniaxial tension and compression.

The dimensionless variable q_h that appears in the yield function (58) is a function of the hardening variable κ_p. It controls the size and shape of the yield surface and, thereby, of the elastic domain. The **hardening law** is given by

$$q_\mathrm{h}(\kappa_\mathrm{p}) = \begin{cases} q_{\mathrm{h}0} + (1 - q_{\mathrm{h}0}) \kappa_\mathrm{p} ({\kappa_\mathrm{p}}^2 - 3\kappa_\mathrm{p} + 3) & \text{if } \kappa_\mathrm{p} < 1 \\ 1 & \text{if } \kappa_\mathrm{p} \geq 1 \end{cases} \tag{64}$$

The initial inclination of the hardening curve (at $\kappa_\mathrm{p} = 0$) is positive and finite, and the inclination at peak (i.e., at $\kappa_\mathrm{p} = 1$) is zero.

The evolution law for the hardening variable,[2]

$$\dot{\kappa}_\mathrm{p} = \frac{\|\dot{\boldsymbol{\varepsilon}}_\mathrm{p}\|}{x_\mathrm{h}(\bar{\sigma}_\mathrm{V})} (2 \cos \bar{\theta})^2 \tag{65}$$

sets the rate of the hardening variable equal to the norm of the plastic strain rate scaled by a hardening ductility measure

$$x_\mathrm{h}(\bar{\sigma}_\mathrm{V}) = \begin{cases} A_\mathrm{h} - (A_\mathrm{h} - B_\mathrm{h}) \exp(-R_\mathrm{h}(\bar{\sigma}_\mathrm{V})/C_\mathrm{h}) & \text{if } R_\mathrm{h}(\bar{\sigma}_\mathrm{V}) \geq 0 \\ E_\mathrm{h} \exp(R_\mathrm{h}(\bar{\sigma}_\mathrm{V})/F_\mathrm{h}) + D_\mathrm{h} & \text{if } R_\mathrm{h}(\bar{\sigma}_\mathrm{V}) < 0 \end{cases} \tag{66}$$

The dependence of the scaling factor x_h on the volumetric effective stress $\bar{\sigma}_\mathrm{V}$ is constructed such that the model response is more ductile under compression. The variable

$$R_\mathrm{h}(\bar{\sigma}_\mathrm{V}) = -\frac{\bar{\sigma}_\mathrm{V}}{\bar{f}_c} - \frac{1}{3} \tag{67}$$

is a linear function of the volumetric effective stress. Model parameters $A_\mathrm{h}, B_\mathrm{h}, C_\mathrm{h}$ and D_h are calibrated from the values of strain at peak stress under uniaxial tension, uniaxial compression and triaxial compression, whereas the parameters

$$E_\mathrm{h} = B_\mathrm{h} - D_\mathrm{h} \tag{68}$$

$$F_\mathrm{h} = \frac{(B_\mathrm{h} - D_\mathrm{h}) C_\mathrm{h}}{B_\mathrm{h} - A_\mathrm{h}} \tag{69}$$

[2] In the original paper [3], equation (65) was written with $\cos^2 \bar{\theta}$ instead of $(2 \cos \bar{\theta})^2$, but all the results presented in that paper were computed with OOFEM using an implementation based on (65).

are determined from the conditions of a smooth transition between the two parts of equation (66) at $R_{\mathrm{h}} = 0$.

For the present model, the **evolution of damage** starts after full saturation of plastic hardening, i.e., at $\kappa_{\mathrm{p}} = 1$. This greatly facilitates calibration of model parameters, because the strength envelope is fully controled by the plastic part of the model and damage affects only the softening behavior. In contrast to pure damage models, damage is assumed to be driven by the plastic strain, more specifically by its volumetric part, which is closely related to cracking. To slow down the evolution of damage under compressive stress states, the damage-driving variable κ_{d} is not set equal to the volumetric plastic strain, but it is defined incrementally by the rate equation

$$\dot{\kappa}_{\mathrm{d}} = \begin{cases} 0 & \text{if } \kappa_{\mathrm{p}} < 1 \\ \mathrm{Tr}(\dot{\boldsymbol{\varepsilon}}_{\mathrm{p}})/x_{\mathrm{s}}\left(\bar{\sigma}_{\mathrm{V}}\right) & \text{if } \kappa_{\mathrm{p}} \geq 1 \end{cases} \tag{70}$$

where

$$x_{\mathrm{s}}(\bar{\sigma}_{\mathrm{V}}) = \begin{cases} 1 + A_{\mathrm{s}} R_{\mathrm{s}}^2(\bar{\sigma}_{\mathrm{V}}) & \text{if } R_{\mathrm{s}}(\bar{\sigma}_{\mathrm{V}}) < 1 \\ \\ 1 - 3A_{\mathrm{s}} + 4A_{\mathrm{s}}\sqrt{R_{\mathrm{s}}(\bar{\sigma}_{\mathrm{V}})} & \text{if } R_{\mathrm{s}}(\bar{\sigma}_{\mathrm{V}}) \geq 1 \end{cases} \tag{71}$$

is a softening ductility measure. Parameter A_{s} is determined from the softening response in uniaxial compression. The dimensionless variable $R_{\mathrm{s}} = \dot{\varepsilon}_{\mathrm{pV}}^{-}/\dot{\varepsilon}_{\mathrm{pV}}$ is defined as the ratio between the "negative" volumetric plastic strain rate

$$\dot{\varepsilon}_{\mathrm{pV}}^{-} = \sum_{I=1}^{3} \langle -\dot{\varepsilon}_{\mathrm{pI}} \rangle \tag{72}$$

and the total volumetric plastic strain rate $\dot{\varepsilon}_{\mathrm{pV}}$. This ratio depends only on the flow direction $\partial g_{\mathrm{p}}/\partial \bar{\boldsymbol{\sigma}}$, and thus R_{s} can be shown to be a unique function of the volumetric effective stress. In (72), $\dot{\varepsilon}_{\mathrm{pI}}$ are the principal components of the rate of plastic strains and $\langle \cdot \rangle$ denotes the McAuley brackets (positive-part operator). For uniaxial tension, for instance, all three principal plastic strain rates are nonnegative, and so $\dot{\varepsilon}_{\mathrm{pV}}^{-} = 0$, $R_{\mathrm{s}} = 0$ and $x_{\mathrm{s}} = 1$. This means that under uniaxial tensile loading we have $\kappa_{\mathrm{d}} = \kappa_{\mathrm{p}} - 1$. On the other hand, under compressive stress states the negative principal plastic strain rates lead to a ductility measure x_{s} greater than one and the evolution of damage is slowed down. It should be emphasized that the flow rule for this specific model is constructed such that the volumetric part of plastic strain rate at the ultimate yield surface cannot be negative.

The relation between the damage variable ω and the internal variable κ_{d} (maximum level of equivalent strain) is assumed to have the exponential form

$$\omega = 1 - \exp\left(-\kappa_{\mathrm{d}}/\varepsilon_f\right) \tag{73}$$

where ε_f is a parameter that controls the slope of the softening curve. To avoid pathological sensitivity of the results to the size of finite elements, this parameter is adjusted according to the element size. Therefore, another parameter, w_f, is considered as a material property, and $\boldsymbol{\varepsilon}_f$ is evaluated as w_f/h where h is the size of the finite element (projected onto the direction of maximum principal strain).

The damage-plastic model contains 15 parameters, but only 6 of them are actually calibrated for different concrete types, namely Young's modulus E,

Description	Damage-plastic model for concrete
Record Format	**ConcreteDPM** d$_{(rn)}$ # E$_{(rn)}$ # n$_{(rn)}$ # tAlpha$_{(rn)}$ # ft$_{(rn)}$ # fc$_{(rn)}$ # wf$_{(rn)}$ # ecc$_{(rn)}$ # kinit$_{(rn)}$ # Ahard$_{(rn)}$ # Bhard$_{(rn)}$ # Chard$_{(rn)}$ # Dhard$_{(rn)}$ # Asoft$_{(rn)}$ # helem$_{(rn)}$ # dilation$_{(rn)}$ # yieldtol$_{(rn)}$ # newtoniter$_{(in)}$ #
Parameters	- d material density - E Young modulus - n Poisson ratio - *tAlpha* thermal dilatation coefficient - *ft* uniaxial tensile strength - *fc* uniaxial compressive strength - *wf* parameter w_f that controls the slope of the softening branch (serves for the evaluation of $\varepsilon_f = w_f/h$ to be used in (73)) - *ecc* eccentricity parameter e from (59), optional, default value 0.525 - *kinit* parameter $q_{\mathrm{h}0}$ from (64), optional, default value 0.1 - *Ahard* parameter A_{h} from (66), optional, default value 0.08 - *Bhard* parameter B_{h} from (66), optional, default value 0.003 - *Chard* parameter C_{h} from (66), optional, default value 2 - *Dhard* parameter D_{h} from (66), optional, default value 10^{-6} - *Asoft* parameter A_{s} from (71), optional, default value 15 - *helem* element size h, optional (if not specified, the actual element size is used) - *dilation* dilation factor (ratio between lateral and axial plastic strain rates in the softening regime under uniaxial compression), optional, default value -0.85 - *yieldtol* tolerance for the implicit stress return algorithm, optional, default value 10^{-10} - *newtoniter* maximum number of iterations in the implicit stress return algorithm, optional, default value 100
Supported modes	3dMat

Table 25: Damage-plastic model for concrete – summary.

Poisson's ratio ν, tensile strength f_{t}, compressive strength f_{c}, parameter ε_{f} of the damage law (73), and parameter A_{s} in the ductility measure (71) of the damage model. The remaining parameters can be set to their default values specified in [3].

The model parameters are summarized in table 25. Note that it is possible to specify the "size" of finite element, h, which (if specified) replaces the actual element size in the relation between the material parameter w_f and the parameter $\varepsilon_f = w_f/h$ that is used in (73). The usual approach is to consider h as the actual element size (evaluated automatically by OOFEM), in which case the optional parameter h is missing (or is set to 0., which has the same effect in the code). However, for various studies of mesh sensitivity it is useful to have

the option of setting ε_f to a fixed value, independent of the element size. This can be achieved by specifying h as an input "material" value. For instance, if h is set to 1, then the parameter ε_f used in (73) will have the same value as the input parameter w_f, independently of the actual element size.

1.5 Orthotropic damage model with fixed crack orientations for composites

The model is designed for transversely isotropic elastic material defined by five elastic material constants. Typical example is a carbon fiber tow. Axis 1 represents the material principal direction. The orthotropic material constants are defined as

$$\nu_{12} = \nu_{13},\ \nu_{21} = \nu_{31},\ \nu_{23} = \nu_{32},\ E_{22} = E_{33} \tag{74}$$

$$G_{12} = G_{13} = G_{21} = G_{31}, G_{23} = G_{32} \tag{75}$$

$$\frac{\nu_{12} = \nu_{13}}{E_{11}} = \frac{\nu_{21} = \nu_{31}}{E_{22}},\ \frac{\nu_{31} = \nu_{21}}{E_{33}} = \frac{\nu_{13} = \nu_{12}}{E_{11}} \tag{76}$$

Material orientation on a finite element can be specified with *lcs* optional parameter. If unspecified, material orientation is the same as the global coordinate system. *lcs* array contains six numbers, where the first three numbers represent a directional vector of the local x-axis, and the next three numbers represent a directional vector of the local y-axis with the reference to the global coordinate system. The composite material is extended to 1D and is also suitable for beams and trusses. In such particular case, the *lcs* has no effect and the 1D element orientation is aligned with the global xx component.

The index $p,\ p \in \{11, 22, 33, 23, 31, 12\}$ symbolizes six components of stress or strain vectors. The linear softening occurs after reaching a critical stress $f_{p,0}$, see Fig. 8. Orientation of cracks is assumed to be orthogonal and aligned with an orientation of material axes [1, pp.236]. The transverse isotropy is generally lost upon fracture, material becomes orthotropic and six damage parameters d_p are introduced.

The compliance material matrix $\mathbf{H}$, in the secant form and including damage parameters, reads

$$\begin{bmatrix} \frac{1}{(1-d_{11})E_{11}} & -\frac{\nu_{21}}{E_{22}} & -\frac{\nu_{31}}{E_{33}} & 0 & 0 & 0 \\ -\frac{\nu_{12}}{E_{11}} & \frac{1}{(1-d_{22})E_{22}} & -\frac{\nu_{32}}{E_{33}} & 0 & 0 & 0 \\ -\frac{\nu_{13}}{E_{11}} & -\frac{\nu_{23}}{E_{22}} & \frac{1}{(1-d_{33})E_{33}} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{(1-d_{23})G_{23}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{(1-d_{31})G_{31}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{(1-d_{12})G_{12}} \end{bmatrix} \tag{77}$$

Damage occurs when any out of six stress tensor components exceeds a given strength $f_{p,0}$

$$|\sigma_p| \geq |f_{p,0}| \tag{78}$$

Positive and negative ultimate strengths can be generally different but share the same damage variable. At the point of damage initiation, see Fig. 8, one evaluates $\varepsilon_{p,E}$ and characteristic element length l_p, generally different for each damage mode. Given the fracture energy $G_{F,p}$, the maximum strain at zero stress $\varepsilon_{p,0}$ is computed

$$\varepsilon_{p,0} = \varepsilon_{p,E} + \frac{2G_{F,p}}{f_{p,0}l_p} \tag{79}$$

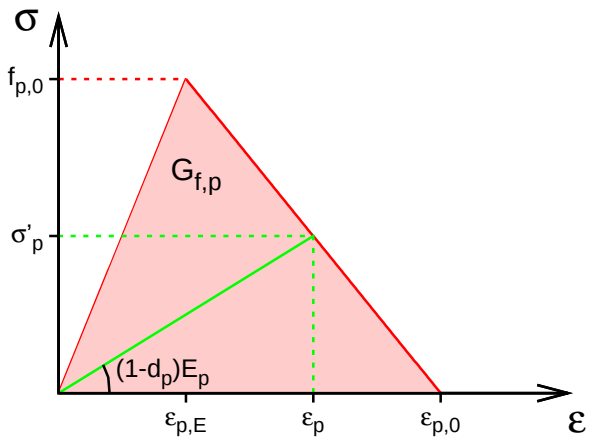

Figure 8: Implemented stress-strain evolution with damage for 1D case. Tension and compression are separated, but sharing the same damage parameter.

The point of damage initiation is never reached exactly, one needs to interpolate between the previous equilibrated step and current step to achieve objectivity.

The evolution of damage d_p is based on the evolution of corresponding strain ε_p. A maximum achieved strain is stored in the variable κ_p. If $\varepsilon_p > \kappa_p$ the damage may grow so the corresponding damage variable d_p may increase. Desired stress σ'_p is evaluated from the actual strain ε_p

$$\sigma'_p = f_{p,0}\frac{\varepsilon_{p,0} - \varepsilon_p}{\varepsilon_{p,0} - \varepsilon_{p,E}} \tag{80}$$

and the calculation of damage variables d_p stems from Eq. 77, for example

$$d_{11} = 1 - \frac{\sigma'_{11}}{E_{11}\left(\varepsilon_{11} + \frac{\nu_{21}}{E_{22}}\sigma_{22} + \frac{\nu_{31}}{E_{33}}\sigma_{33}\right)} \tag{81}$$

$$d_{12} = 1 - \frac{\sigma'_{12}}{G_{12}\varepsilon_{12}} \tag{82}$$

Damage is always controled not to decrease. Fig. 9 shows a typical performance for this damage model in one direction.

The damage initiation is based on a trial stress. It becomes necessary for higher precision to skip a few first iteration, typically 5, and then to introduce damage. A parameter *afterIter* is designed for this purpose and *MinIter* forces a solver always to proceed certain amount of iterations.

allowSnapBack skips the checking of sufficient fracture energy for each direction. If not specified, all directions are checked to prevent snap-back which dissipates incorrect amount of energy.

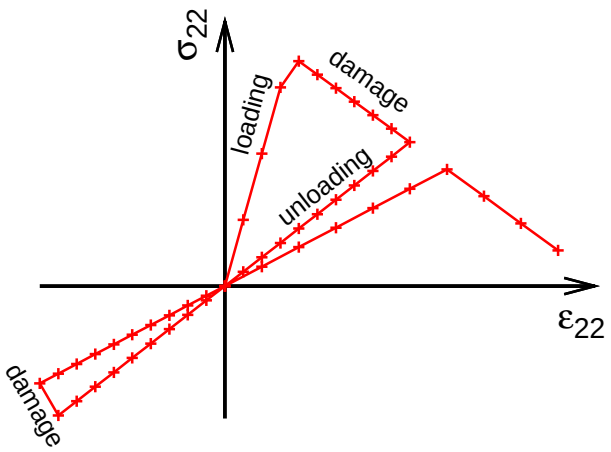

Figure 9: Typical loading/unloading material performance for homogenized stress and strain in the direction '2'. Note that one damage parameter is common for both tension and compression.

1.6 Orthotropic elastoplastic model with isotropic damage

This model combines orthotropic elastoplasticity with isotropic damage. Material orthotropy is described by the fabric tensor, i.e., a symmetric second-order tensor with principal directions aligned with the axes of orthotropy and principal values normalized such that their sum is 3. Elastic constants as well as coefficients that appear in the yield condition are linked to the principal values of the fabric tensor and to porosity. The yield condition is piecewise quadratic, with different parameters in the regions of positive and negative volumetric strain.

1.6.1 Local formulation

The basic equations include an additive decomposition of total strain into elastic (reversible) part and plastic (irreversible) part

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_{\mathrm{e}} + \boldsymbol{\varepsilon}_{\mathrm{p}}, \tag{83}$$

the stress strain law

$$\boldsymbol{\sigma} = (1-\omega)\,\bar{\boldsymbol{\sigma}} = (1-\omega)\,\boldsymbol{D} : \boldsymbol{\varepsilon}_{\mathrm{e}}, \tag{84}$$

the yield function

$$f(\bar{\boldsymbol{\sigma}}, \kappa) = \sqrt{\bar{\boldsymbol{\sigma}} : \boldsymbol{F} : \bar{\boldsymbol{\sigma}}} - \sigma_Y(\kappa). \tag{85}$$

loading-unloading conditions

$$f(\bar{\boldsymbol{\sigma}}, \kappa) \le 0 \qquad \dot{\lambda} \ge 0 \qquad \dot{\lambda} f(\bar{\boldsymbol{\sigma}}, \kappa) = 0, \tag{86}$$

evolution law for plastic strain

$$\dot{\boldsymbol{\varepsilon}}_{\mathrm{p}} = \dot{\lambda} \frac{\partial f}{\partial \bar{\boldsymbol{\sigma}}}, \tag{87}$$

Description	Orthotropic damage model with fixed crack orientations for composites
Record Format	**compdammat** num$_{(in)}$ # d$_{(rn)}$ # Exx$_{(rn)}$ # EyyEzz$_{(rn)}$ # nuxynuxz$_{(rn)}$ # nuyz$_{(rn)}$ # GxyGxz$_{(rn)}$ # Tension_f0_Gf$_{(ra)}$ # Compres_f0_Gf$_{(ra)}$ # [afterIter$_{(in)}$ #] [allowSnapBack$_{(()}$ #ia)]
Parameters	- *num* material model number - *d* material density - *Exx* Young's modulus for principal direction xx - *EyyEzz* Young's modulus in orthogonal directions to the principal direction xx - *nuxynuxz* Poisson's ratio in xy and xz directions - *nuyz* Poisson's ratio in yz direction - *GxyGxz* shear modulus in xy and xz directions - *Tension_f0_Gf* array with six pairs of positive numbers. Each pair describes maximum stress in tension and fracture energy for each direction (xx, yy, zz, yz, zx, xy) - *Compres_f0_Gf* array with six pairs of numbers. Each pair describes maximum stress in compression (given as a negative number) and positive fracture energy for each direction (xx, yy, zz, yz, zx, xy)
Supported modes	3dMat, 1dMat - *afterIter* how many iterations must pass until damage is computed from strains, zero is default. User must ensure that the solver proceeds the minimum number of iterations. - *allowSnapBack* array to skip checking for snap-back. The array members are 1-6 for tension and 7-12 for compression components.

Table 26: Orthotropic damage model with fixed crack orientations for composites – summary.

the incremental definition of cumulated plastic strain

$$\dot{\kappa} = \|\dot{\boldsymbol{\varepsilon}}_{\mathrm{p}}\|, \tag{88}$$

the law governing the evolution of the damage variable

$$\omega(\kappa) = \omega_c(1 - \mathrm{e}^{-a\kappa}), \tag{89}$$

and the hardening law

$$\sigma_Y(\kappa) = 1 + \sigma_H(1 - \mathrm{e}^{-s\kappa}). \tag{90}$$

In the equations above, $\bar{\boldsymbol{\sigma}}$ is the effective stress tensor, $\boldsymbol{D}$ is the elastic stiffness tensor, f is the yield function, λ is the consistency parameter (plastic multiplier), ω is the damage variable, σ_Y is the yield stress and s, a, σ_H and ω_c are positive material parameters. Material anisotropy is characterized by the second-order positive definite fabric tensor

$$\boldsymbol{M} = \sum_{i=1}^{3} m_i(\boldsymbol{m}_i \otimes \boldsymbol{m}_i), \tag{91}$$

normalized such that $\mathrm{Tr}(\boldsymbol{M}) = 3$, m_i are the eigenvalues and $\boldsymbol{m}_i$ the eigenvectors. The eigenvectors of the fabric tensor determine the directions of material orthotropy and the components of the elastic stiffness tensor $\boldsymbol{D}$ are linked to eigenvalues of the fabric tensor. In the coordinate system aligned with m_i, $i = 1, 2, 3$, the stiffness can be presented in Voigt (engineering) notation as

$$\boldsymbol{D} = \begin{bmatrix} \frac{1}{E_1} & -\frac{\nu_{12}}{E_1} & -\frac{\nu_{13}}{E_1} & 0 & 0 & 0 \\ -\frac{\nu_{21}}{E_2} & \frac{1}{E_2} & -\frac{\nu_{23}}{E_2} & 0 & 0 & 0 \\ -\frac{\nu_{31}}{E_3} & -\frac{\nu_{32}}{E_3} & \frac{1}{E_3} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G_{23}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G_{13}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G_{12}} \end{bmatrix}^{-1}, \tag{92}$$

where $E_i = E_0 \rho^k m_i^{2l}$, $G_{ij} = G_0 \rho^k m_i^l m_j^l$ and $\nu_{ij} = \nu_0 \frac{m_i^l}{m_j^l}$. Here, E_0, G_0 and ν_0 are elastic constants characterizing the compact (poreless) material, ρ is the volume fraction of solid phase and k and l are dimensionless exponents.

Similar relations as for the stiffness tensor are also postulated for the components of a fourth-order tensor $\boldsymbol{F}$ that is used in the yield condition. The yield condition is divided into tensile and compressive parts. Tensor $\boldsymbol{F}$ is different in each part of the effective stress space. This tensor is denoted $\boldsymbol{F}^+$ in tensile part, characterized by $\hat{\boldsymbol{N}} : \bar{\boldsymbol{\sigma}} \leq 0$ and $\boldsymbol{F}^-$ in compressive part, characterized by $\hat{\boldsymbol{N}} : \bar{\boldsymbol{\sigma}} \leq 0$, where

$$\hat{\boldsymbol{N}} = \frac{\sum_{i=1}^{3} m_i^{-2q}}{\sqrt{\sum_{i=1}^{3} m_i^{-4q}}} (\boldsymbol{m}_i \otimes \boldsymbol{m}_i) \tag{93}$$

$$\boldsymbol{F}^{\pm} = \begin{bmatrix} \frac{1}{\left(\sigma_1^{\pm}\right)^2} & -\frac{\chi_{12}^{\pm}}{\left(\sigma_1^{\pm}\right)^2} & -\frac{\chi_{13}^{\pm}}{\left(\sigma_1^{\pm}\right)^2} & 0 & 0 & 0 \\ -\frac{\chi_{21}^{\pm}}{\left(\sigma_2^{\pm}\right)^2} & \frac{1}{\left(\sigma_2^{\pm}\right)^2} & -\frac{\chi_{23}^{\pm}}{\left(\sigma_2^{\pm}\right)^2} & 0 & 0 & 0 \\ -\frac{\chi_{31}^{\pm}}{\left(\sigma_3^{\pm}\right)^2} & -\frac{\chi_{32}^{\pm}}{\left(\sigma_3^{\pm}\right)^2} & \frac{1}{\left(\sigma_3^{\pm}\right)^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\tau_{23}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\tau_{13}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{\tau_{12}} \end{bmatrix}. \tag{94}$$

In the equation above $\sigma_i^{\pm} = \sigma_0^{\pm} \rho^p m_i^{2q}$ is uniaxial yield stress along the i-th principal axis of orthotropy, $\tau_{ij} = \tau_0 \rho^p m_i^q m_j^q$ is the shear yield stress in the plane of orthotropy and $\chi_{ij}^{\pm} = \chi_0^{\pm} \frac{m_i^{2q}}{m_j^{2q}}$ is the so-called interaction coefficient, p and q are dimensionless exponents and parameters with subscript 0 are related to a fictitious material with zero porosity. The yield surface is continuously differentiable if the parameters values are constrained by the condition

$$\frac{\chi_0^- + 1}{(\sigma_0^-)^2} = \frac{\chi_0^+ + 1}{(\sigma_0^+)^2}. \tag{95}$$

The model description and parameters are summarized in table 27.

Description	Anisotropic elastoplastic model with isotropic damage
Record Format	**TrabBone3d** (in) # d(rn) # eps0(rn) # nu0(rn) # mu0(rn) # expk(rn) # expl(rn) # m1(rn) # m2(rn) # rho(rn) # sig0pos(rn) # sig0neg(rn) # chi0pos(rn) # chi0neg(rn) # tau0(rn) # plashardfactor(rn) # expplashard(rn) # expdam(rn) # critdam(rn) #
Parameters	- material number - *d* material density - *eps0* Young modulus (at zero porosity) - *nu0* Poisson ratio (at zero porosity) - *mu0* shear modulus of elasticity (at zero porosity) - *m1* first eigenvalue of the fabric tensor - *m2* second eigenvalue of the fabric tensor - *rho* volume fraction of solid phase - *sig0pos* yield stress in tension - *sig0neg* yield stress in compression - *tau0* yield stress in shear - *chi0pos* interaction coefficient in tension - *plashardfactor* hardening parameter - *expplashard* exponent in hardening law - *expdam* exponent in damage law - *critdam* critical damage - *expk* exponent k in the expression for elastic stiffness - *expl* exponent l in the expression for elastic stiffness - *expq* exponent q in the expression for tensor $\boldsymbol{F}$ - *expp* exponent p in the expression for tensor $\boldsymbol{F}$
Supported modes	3dMat

Table 27: Anisotropic elastoplastic model with isotropic damage - summary.

1.6.2 Nonlocal formulation

The model is regularized by the over-nonlocal formulation with damage driven by a combination of local and nonlocal cumulated plastic strain

$$\hat{\kappa} = (1-m)\kappa + m\bar{\kappa}, \tag{96}$$

where m is a dimensionless material parameter (typically $m > 1$) and

$$\bar{\kappa}(x) = \int_V \alpha(x,s)\kappa(s)\,\mathrm{d}s \tag{97}$$

is the nonlocal cumulated plastic strain. The nonlocal weight function is defined as

$$\alpha(x,s) = \frac{\alpha_0(\|x-s\|)}{\int_V \alpha_0(\|x-t\|)\,\mathrm{d}t} \tag{98}$$

where

$$\alpha_0(r) = \begin{cases} \left(1-\frac{r^2}{R^2}\right)^2 & \text{if } r \le R \\ 0 & \text{if } r > R \end{cases} \tag{99}$$

Parameter R is related to the internal length of the material. The model description and parameters are summarized in table 28.

Description	Nonlocal anisotropic elastoplastic model with isotropic damage
Record Format	**TrabBoneNL3d** (in) # d(rn) # eps0(rn) # nu0(rn) # mu0(rn) # expk(rn) # expl(rn) # m1(rn) # m2(rn) # rho(rn) # sig0pos(rn) # sig0neg(rn) # chi0pos(rn) # chi0neg(rn) # tau0(rn) # plashardfactor(rn) # expplashard(rn) # expdam(rn) # critdam(rn) # m(rn) # R(rn) #
Parameters	- material number - *d* material density - *eps0* Young modulus (at zero porosity) - *nu0* Poisson ratio (at zero porosity) - *mu0* shear modulus (at zero porosity) - *m1* first eigenvalue of the fabric tensor - *m2* second eigenvalue of the fabric tensor - *rho* volume fraction of the solid phase - *sig0pos* yield stress in tension - *tau0* yield stress in shear - *chi0pos* interaction coefficient in tension - *chi0neg* interaction coefficient in compression - *plashardfactor* hardening parameter - *expplashard* exponent in the hardening law - *expdam* exponent in the damage law - *critdam* critical damage - *expk* exponent k in the expression for elastic stiffness - *expl* exponent l in the expression for elastic stiffness - *expq* exponent q in the expression for tensor $\boldsymbol{F}$ - *expp* exponent p in the expression for tensor $\boldsymbol{F}$ - m over-nonlocal parameter - R nonlocal interaction radius
Supported modes	3dMat

Table 28: Nonlocal formulation of anisotropic elastoplastic model with isotropic damage – summary.

2 Material Models for Transport Problems

2.1 Isotropic linear material

Linear isotropic material model for transport problems. The model parameters are summarized in table 29.

Description	Linear isotropic elastic material
Record Format	**IsoHeat** num$_{(in)}$ # d$_{(rn)}$ # k$_{(rn)}$ # c$_{(rn)}$ #
Parameters	- *num* material model number
	- *d* material density
	- *k* Conductivity
	- *c* Specific heat capacity
Supported modes	_2dHeat

Table 29: Linear Isotropic Material - summary.

2.2 Material for cement hydration

CemhydMat represents a hydrating material based on CEMHYD3D model version 3.0, developed at NIST [2]. The model represents a digital hydrating microstructure, driven with cellular automata rules and combined with cement chemistry. Ordinary Portland cement is treated without any difficulties, blended cements are ususally decomposed into hydrating Portland contribution and intert secondary cementitious material. The microstructure size can be from $10 \times 10 \times 10$ to over $200 \times 200 \times 200$ μm. For standard computations the size $50 \times 50 \times 50$ suffices.

Each material instance creates an independent microstructure. It is also possible to enforce having different microstructures in each integration point. The hydrating model is coupled with temperature and averaging over shared elements within one material instance occurs during the solution. Such approach allows domain partitioning to many CemhydMat instances, depending on expected accuracy or computational speed. A more detailed description with engineering examples was published [13]. Table 30 summarizes input parameters.

The input XML file specifies the details about cement and concrete composition. It is possible to start all simulations from the scratch, i.e. with the reconstruction of digital microstructure. Alternatively, the digital microstructure can be provided directly in two files; one for chemical phases, the second for particle's IDs. The XML input file can be created with the CemPy package, obtainable from http://mech.fsv.cvut.cz/~smilauer/index.php?id=software. The CemPy package alleviates tedious preparation of particle size distribution etc.

The linear solver (specified as NonStationaryProblem) performs well when the time integration step is small enough (order of minutes) and heat capacity, conductivity and density remain constant. If not so, use of nonlinear solver is strongly suggested (specified as NlTransientTransportProblem).

Description	Cemhyd hydrating material
Record Format	**CemhydMat** num$_{(in)}$ # d$_{(rn)}$ # k$_{(rn)}$ # c$_{(rn)}$ # file$_{(s)}$ # [eachGP$_{(in)}$ #] [densityType$_{(in)}$ #] [conductivityType$_{(in)}$ #] [capacityType$_{(in)}$ #] [castingtime$_{(ia)}$ #] [nowarnings$_{(ia)}$ #]
Parameters	- *num* material model number - *d* material density - *k* Conductivity - *c* Specific heat capacity - *file* XML input file for cement microstructure and concrete composition - *eachGP* 0 (default) no separate microstructures in each GP, 1 assign separate microstructures to each GP - *densityType* 0 (default) get density from OOFEM input file, 1 get it from XML input file - *conductivityType* 0 (default) get constant conductivity from OOFEM input file, 1 compute as λ = k(1.33 − 0.33DoH) [10] - *capacityType* 0 (default) get capacity, 1 according to Bentz, 2 according to XML and CEMHYD3D routines - *castingtime* optional casting time of concrete, from which hydration takes place - *nowarnings* supresses warnings when material data are out of standard ranges. The array of size 4 represent entries for density, conductivity, capacity, temperature. Nonzero values mean supression.
Supported modes	_2dHeat, _3dHeat

Table 30: Cemhydmat - summary.

2.3 Coupled heat and mass transfer material model

Coupled heat and mass transfer material model. Source: T. Krejci doctoral thesis; Bazant and Najjar, 1972; Pedersen, 1990. Assumptions: water vapor is the only driving mechanism; relative humidity is from range 0.2 - 0.98 (I and II regions). The model parameters are summarized in table 31.

Description	Coupled heat and mass transfer material model
Record Format	**HeMotk** num$_{(in)}$ # d$_{(rn)}$ # a_0$_{(rn)}$ # nn$_{(rn)}$ # phi_c$_{(rn)}$ # delta_wet$_{(rn)}$ # w_h$_{(rn)}$ # n$_{(rn)}$ # a$_{(rn)}$ # latent$_{(rn)}$ # c$_{(rn)}$ # rho$_{(rn)}$ # chi_eff$_{(rn)}$ # por$_{(rn)}$ # rho_gws$_{(rn)}$ #
Parameters	- *num* material model number - *d*, *rho* material density - *a_0* constant (obtained from experiments) a_0 [Bazant and Najjar, 1972] - *nn* constant-exponent (obtained from experiments) n [Bazant and Najjar, 1972] - *phi_c* constant-relative humidity (obtained from experiments) phi_c [Bazant and Najjar, 1972] - *delta_wet* constant-water vapor permeability (obtained from experiments) delta_wet [Bazant and Najjar, 1972] - *w_h* constant water content (obtained from experiments) w_h [Pedersen, 1990] - *n* constant-exponent (obtained from experiments) n [Pedersen, 1990] - *a* constant (obtained from experiments) A [Pedersen, 1990] - *latent* latent heat of evaporation - *c* thermal capacity - *chi_eff* effective thermal conductivity - *por* porosity - *rho_gws* saturation volume density
Supported modes	_2dHeMo

Table 31: Coupled heat and mass transfer material model - summary.

3 Material Models for Fluid Dynamic

3.1 Newtonian fluid

Constitutive model of Newtonian fluid. The model parameters are summarized in table 32.

Description	Newtonian Fluid material
Record Format	**NewtonianFluid** num$_{(in)}$ # d$_{(rn)}$ # mu$_{(rn)}$ #
Parameters	- *num* material model number
	- *d* material density
	- *mu* viscosity
Supported modes	2dFlow

Table 32: Newtonian Fluid material - summary.

3.2 Bingham fluid

Constitutive model of Bingham fluid. This is a constitutive model of non-Newtonian type. The model parameters are summarized in table 33.

In the Bingham model the flow is characterized by following constitutive equation

$$\boldsymbol{\tau} = \boldsymbol{\tau}_0 + \mu\dot{\boldsymbol{\gamma}} \quad \text{if } \dot{\tau} \geq \tau_0 \tag{100}$$

$$\dot{\gamma} = \mathbf{0} \quad \text{if } \dot{\tau} \leq \tau_0 \tag{101}$$

where $\boldsymbol{\tau}$ is the shear stress applied to material, $\dot{\tau} = \sqrt{\boldsymbol{\tau} : \boldsymbol{\tau}}$ is the shear stress measure, $\dot{\boldsymbol{\gamma}}$ is the shear rate, $\boldsymbol{\tau}_0$ is the yield stress, and μ is the plastic viscosity. The parameters for the model can be in general determined using two possibilities: (i) stress controlled rheometer, when the stress is applied to material and shear rate is measured, and (ii) shear rate controlled rheometer, where concrete is sheared and stress is measured. However, most of the widely used tests are unsatisfactory in the sense, that they measure only one parameter. These one-factor tests include slump test, penetrating rod test, and Ve-Be test. Recently, some tests providing two parameters on output have been designed (BTRHEOM, IBB, and BML rheometers). Also a refined version of the standard slump test has been developed for estimating yield stress and plastic viscosity. The test is based on measuring the time necessary for the upper surface of the concrete cone in the slump to fall a distance 100 mm. Semi-empirical models are then proposed for estimating yield stress and viscosity based on measured results. The advantage is, that this test does not require any special equipment, provided that the one for the standard version is available.

In order to avoid numerical difficulties caused by the existence of the sharp angle in material model response at $\tau = \tau_0$, the numerical implementation uses following smoothed relation for viscosity

$$\mu = \mu_0 + \frac{\tau_0}{\dot{\gamma}}(1 - e^{-m\dot{\gamma}}) \tag{102}$$

where m is so called stress growth parameter. The higher value of parameter m, the closer approximation of the original constitutive equation (100) is obtained.

Description	Bingham fluid material
Record Format	**BinghamFluid** num$_{(in)}$ # d$_{(rn)}$ # mu0$_{(rn)}$ # tau0$_{(rn)}$ #
Parameters	- *num* material model number - *d* material density - *mu0* viscosity - *tau0* Yield stress
Supported modes	_2dFlow

Table 33: Bingham Fluid material - summary.

3.3 Two-fluid material

Material coupling the behaviour of two particulat materials based on rule of mixture. The weighting factor is VOF fraction. The model parameters are summarized in table 34.

Description	Two-Fluid material
Record Format	**twofluidmat** num$_{(in)}$ # mat$_{(ia)}$ #
Parameters	- *num* material model number - *mat* integer array contaning two numbers representing numbers of material models of which the receiver is composed. Material with index 0 is a material, that is fully active in a cell with VOF=0, material with index 1 is a material fuully active in a cell with VOF=1.
Supported modes	_2dFlow

Table 34: Two-Fluid material - summary.

4 Material Drivers - Theory & Application

The purpose of this section is to present the theoretical backgroung of some handy general purpose algorithms, that are provided in oofem in the form of general material base classes. They can significantly facilitate the implementation of particular material models that are based on such concepts. Typical example can be a general purpose plasticity class, that implements general stress return and stifness matrix evaluation algorithms, based on provided methods for computing yield functions and corresponding derivatives. Particular models are simply derived from the base classes, inheriting common algorithms.

4.1 Multisurface plasticity driver - MPlasticMaterial class

In this section, a general multisurface plasticity theory with hardening/softening is reviewed. The presented algorithms are implemented in MPlasticMaterial class.

4.1.1 Plasticity overview

Let $\boldsymbol{\sigma}, \boldsymbol{\varepsilon}$, and $\boldsymbol{\varepsilon}^p$ be the stress, total strain, and plastic strain vectors, respectively. It is assumed that the total strain is decomposed into reversible elastic and irreversible plastic parts

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}^e + \boldsymbol{\varepsilon}^p \tag{103}$$

The elastic response is characterized in terms of elastic constitutive matrix $\boldsymbol{D}$ as

$$\boldsymbol{\sigma} = \boldsymbol{D}\boldsymbol{\varepsilon}^e = \boldsymbol{D}(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^p) \tag{104}$$

As long as the stress remains inside the elastic domain, the deformation process is purely elastic and the plastic strain does not change. It is assumed that the elastic domain, denoted as IE is bounded by a composite yield surface. It is defined as

$$IE = \{(\boldsymbol{\sigma}, \boldsymbol{\kappa}) | f_i(\boldsymbol{\sigma}, \boldsymbol{\kappa}) < 0, \text{for all i} \in \{1, \cdots, \text{m}\}\} \tag{105}$$

where $f_i(\boldsymbol{\sigma}, \boldsymbol{\kappa})$ are $m \geq 1$ yield functions intersecting in a possibly non-smooth fashion. The vector $\boldsymbol{\kappa}$ contains internal variables controlling the evolution of yield surfaces (amount of hardening or softening). The evolution of plastic strain $\boldsymbol{\varepsilon}^p$ is expressed in Koiter's form. Assuming the non-associated plasticity, this reads

$$\dot{\boldsymbol{\varepsilon}}^p = \sum_{i=1}^{m} \lambda^i \partial_{\boldsymbol{\sigma}} g_i(\boldsymbol{\sigma}, \boldsymbol{\kappa}) \tag{106}$$

where g_i are plastic potential functions. The λ^i are referred as plastic consistency parameters, which satisfy the following Kuhn-Tucker conditions

$$\lambda^i \geq 0, \ f_i \leq 0, \text{ and } \lambda^i f_i = 0 \tag{107}$$

These conditions imply that in the elastic regime the yield function must remain negative and the rate of the plastic multiplier is zero (plastic strain remains constant) while in the plastic regime the yield function must be equal to zero (stress remains on the surface) and the rate of the plastic multiplier is positive.

The evolution of vector of internal hardening/softening variables $\boldsymbol{\kappa}$ is expressed in terms of a general hardening/softening law of the form

$$\dot{\boldsymbol{\kappa}} = \dot{\boldsymbol{\kappa}}(\boldsymbol{\sigma}, \boldsymbol{\lambda}) \tag{108}$$

where $\boldsymbol{\lambda}$ is the vector of plastic consistency parameters λ_i.

4.1.2 Closest-point return algorithm

Let us assume, that at time t_n the total and plastic strain vectors and internal variables are known

$$\{\boldsymbol{\varepsilon}_n, \boldsymbol{\varepsilon}^p_n, \boldsymbol{\kappa}_n\} \text{ given at } t_n$$

By applying an implicit backward Euler difference scheme to the evolution equations (104 and 106) and making use of the initial conditions the following discrete non-linear system is obtained

$$\begin{aligned}
\boldsymbol{\varepsilon}_{n+1} &= \boldsymbol{\varepsilon}_n + \Delta\boldsymbol{\varepsilon} && (109)\\
\boldsymbol{\sigma}_{n+1} &= \boldsymbol{D}(\boldsymbol{\varepsilon}_{n+1} - \boldsymbol{\varepsilon}^p_{n+1}) && (110)\\
\boldsymbol{\varepsilon}^p_{n+1} &= \boldsymbol{\varepsilon}^p_n + \sum \lambda^i \partial_{\boldsymbol{\sigma}} g_i(\boldsymbol{\sigma}_{n+1}, \boldsymbol{\kappa}_{n+1}) && (111)
\end{aligned}$$

In addition, the discrete counterpart of the Kuhn-Tucker conditions becomes

$$\begin{aligned}
f_i(\boldsymbol{\sigma}_{n+1}, \boldsymbol{\kappa}_{n+1}) &= 0 && (112)\\
\lambda^i_{n+1} &\geq 0 && (113)\\
\lambda^i_{n+1} f_i(\boldsymbol{\sigma}_{n+1}, \boldsymbol{\kappa}_{n+1}) &= 0 && (114)
\end{aligned}$$

In the standard displacement-based finite element analysis, the strain evolution is determined by the displacement increments computed on the structural level. The basic task on the level of a material point is to evaluate the stress evolution generated by strain history. According to this, the strain driven algorithm is assumed, i.e. that the total strain $\boldsymbol{\varepsilon}_{n+1}$ is given. Then, the Kuhn-Tucker conditions determine whether a constraint is active. The set of active constraints is denoted as J_{act} and is defined as

$$J_{act} = \{\beta \in \{1, \cdots, m\} | f_\beta = 0 \ \& \ \dot{f}_\beta = 0\} \tag{115}$$

Let's start with the definition of the residual of plastic flow

$$\boldsymbol{R}_{n+1} = -\boldsymbol{\varepsilon}^p_{n+1} + \boldsymbol{\varepsilon}^p_n + \sum_{j \in J_{act}} \lambda^j_{n+1} \partial_\sigma g_{n+1} \tag{116}$$

By noting that total strain $\boldsymbol{\varepsilon}_{n+1}$ is fixed during the increment we can express the plastic strain increment using (104) as

$$\Delta\boldsymbol{\varepsilon}^p_{n+1} = -\boldsymbol{D}\Delta\boldsymbol{\sigma}_{n+1} \tag{117}$$

The linearization of the plastic flow residual (116) yields[3]

$$\begin{aligned}
&\boldsymbol{R} + \boldsymbol{D}^{-1}\Delta\boldsymbol{\sigma} + \sum \lambda \partial_{\sigma\sigma} g \Delta\boldsymbol{\sigma} + \\
&+ \sum \lambda \partial_{\sigma\kappa} g \cdot (\partial_\sigma \kappa \Delta\boldsymbol{\sigma} + \partial_\lambda \kappa \Delta\lambda) + \sum \Delta\lambda \partial_\sigma g = 0 && (118)
\end{aligned}$$

[3]For brevity, the simplified notation is introduced: $f = f(\boldsymbol{\sigma}, \boldsymbol{\kappa})$, $g = g(\boldsymbol{\sigma}, \boldsymbol{\kappa})$, $\boldsymbol{\kappa} = \boldsymbol{\kappa}(\boldsymbol{\sigma}, \lambda)$, and subscript $n+1$ is omitted.

From the previous equation, the stress increment $\Delta\boldsymbol{\sigma}$ can be expressed as

$$\Delta\boldsymbol{\sigma} = -\boldsymbol{H}^{-1}\left(\boldsymbol{R} + \sum \Delta\lambda\partial_\sigma g + \sum \lambda\partial_{\sigma\kappa} g\partial_\lambda\kappa\Delta\lambda\right) \tag{119}$$

where $\boldsymbol{H}$ is algorithmic moduli defined as

$$\boldsymbol{H} = \left[\boldsymbol{D}^{-1} + \sum \lambda\partial_{\sigma\sigma} g + \sum \lambda\partial_{\sigma\kappa} g\partial_\sigma\kappa\right] \tag{120}$$

Differentiation of active discrete consistency conditions (112) yields

$$\boldsymbol{f} + \partial_\sigma \boldsymbol{f}\Delta\boldsymbol{\sigma} + \partial_\kappa \boldsymbol{f}(\partial_\sigma\kappa\Delta\boldsymbol{\sigma} + \partial_\lambda\kappa\Delta\lambda) = 0 \tag{121}$$

Finally, by combining equations (119) and (121), one can obtain expression for incremental vector of consistency parameters $\Delta\boldsymbol{\lambda}$

$$\left[\boldsymbol{V}^T\boldsymbol{H}^{-1}\boldsymbol{U} - \partial_\kappa \boldsymbol{f}\partial_\lambda\boldsymbol{\kappa}\right]\Delta\boldsymbol{\lambda} = \boldsymbol{f} - \boldsymbol{V}^T\boldsymbol{H}^{-1}\boldsymbol{R} \tag{122}$$

where the matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ are defined as

$$\boldsymbol{U} = \left[\partial_\sigma \boldsymbol{g} + \sum \lambda\partial_{\sigma\kappa} g\partial_\lambda\kappa\right] \tag{123}$$

$$\boldsymbol{V} = [\partial_\sigma \boldsymbol{f} + \partial_\kappa \boldsymbol{f}\partial_\sigma\kappa] \tag{124}$$

Before presenting the final return mapping algorithm, the algorithm for determination of the active constrains should be discussed. A yield surface $f_{i,n+1}$ is active if $\lambda^i_{n+1} > 0$. A systematic enforcement of the discrete Kuhn-Tucker condition (112), which relies on the solution of return mapping algorithm, then serves as the basis for determining the active constraints. The starting point in enforcing (112) is to define the trial set

$$J^{trial}_{act} = \{j \in \{1, \cdots, m\} | f^{trial}_{j,n+1} > 0\} \tag{125}$$

where $J_{act} \subseteq J^{trial}_{act}$. Two different procedures can be adopted to determine the final set J_{act}. The conceptual procedure is as follows

- Solve the closest point projection with $J_{act} = J^{trial}_{act}$ to obtain final stresses, along with $\lambda^i_{n+1},\ i \in J^{trial}_{act}$.
- Check the sign of λ^i_{n+1}. If $\lambda^i_{n+1} < 0$, for some $i \in J^{trial}_{act}$, drop the $i-$th constrain from the active set and goto first point. Otherwise exit.

In the procedure 2, the working set J^{trial}_{act} is allowed to change within the iteration process, as follows

- Let $J^{(k)}_{act}$ be the working set at the k-th iteration. Compute increments $\Delta\lambda^{i,(k)}_{n+1},\ i \in J^{(k)}_{act}$.
- Update and check the signs of $\Delta\lambda^{i,(k)}_{n+1}$. If $\Delta\lambda^{i,(k)}_{n+1} < 0$, drop the i-th constrain from the active set $J^{(k)}_{act}$ and restart the iteration. Otherwise continue with next iteration.

If the consistency parameters $\Delta\lambda^i$ can be shown to increase monotonically within the return mapping algorithm, the the latter procedure is preferred since it leads to more efficient computer implementation.

The overall algorithm is convergent, first order accurate and unconditionally stable. The general algorithm is summarized in table (4.1.2).

1. Elastic predictor

   (a) Compute Elastic predictor (assume frozen plastic flow)
   $\boldsymbol{\sigma}_{n+1}^{trial} = \boldsymbol{D}\left(\boldsymbol{\varepsilon}_{n+1} - \boldsymbol{\varepsilon}_n^p\right)$
   $f_{i,n+1}^{trial} = f_i(\boldsymbol{\sigma}_{n+1}^{trial}, \boldsymbol{\kappa}_n),$ for i $\in \{1, \cdots, \mathrm{m}\}$

   (b) Check for plastic processes IF $f_{i,n+1}^{trial} \leq 0$ for all $i \in \{1, \cdots, m\}$ THEN:

   Trial state is the final state, EXIT.

   ELSE:

   $J_{act}^{(0)} = \{i \in \{1, \cdots, m\} | f_{i,n+1}^{trial} > 0\}$
   $\boldsymbol{\varepsilon}_{n+1}^{p(0)} = \boldsymbol{\varepsilon}_n^p,\ \boldsymbol{\kappa}_{n+1}^{(0)} = \boldsymbol{\kappa}_n,\ \lambda_{n+1}^{i(0)} = 0$

   ENDIF

2. Plastic Corrector

   (c) Evaluate plastic strain residual

   $\boldsymbol{\sigma}_{n+1}^{(k)} = \boldsymbol{D}\left(\boldsymbol{\varepsilon}_{n+1} - \boldsymbol{\varepsilon}_{n+1}^{p(k)}\right)$
   $\boldsymbol{R}_{n+1}^{(k)} = -\boldsymbol{\varepsilon}_{n+1}^{p(k)} + \boldsymbol{\varepsilon}_n^p + \sum \lambda_{n+1}^{i(k)} \partial_\sigma g_i$

   (d) Check convergence

   $f_{i,n+1}^{(k)} = f_i(\boldsymbol{\sigma}_{n+1}^{(k)}, \boldsymbol{\kappa}_{n+1}^{(k)})$
   if $f_{i,n+1}^{(k)} < TOL$, for all $i \in J_{act}^{(k)}$ and $\|\boldsymbol{R}_{n+1}^{(k)}\| < TOL$ then EXIT

   (e) Compute consistent moduli

   $\boldsymbol{G} = \left[\boldsymbol{V}^T \boldsymbol{H}^{-1} \boldsymbol{U} - \partial_\kappa \boldsymbol{f} \partial_\lambda \boldsymbol{\kappa}\right]^{-1}$

   (f) Obtain increments to consistency parameter

   $\Delta\boldsymbol{\lambda}_{n+1}^{(k)} = \boldsymbol{G}\{\boldsymbol{f} - \boldsymbol{V}^T \boldsymbol{H}^{-1} \boldsymbol{R}\}_{n+1}^{(k)}$
   If using procedure 2 to determine active constrains, then update the active set and restart iteration if necessary

   (g) Obtain increments of plastic strains and internal variables

   $\Delta\boldsymbol{\varepsilon}_{n+1}^{p(k)} = \boldsymbol{D}^{-1}\left\{\boldsymbol{R}_{n+1}^{(k)} + \sum \Delta\lambda_{n+1}^{i(k)} \partial_\sigma g_{n+1}^{(k)} + \sum \lambda_{n+1}^{i(k)} \partial_{\sigma\kappa} g_{n+1}^{(k)} \partial_\lambda \kappa \Delta\lambda_{n+1}^{i(k)}\right\}$
   $\Delta\boldsymbol{\kappa}_{n+1}^{(k)} = \dot{\boldsymbol{\kappa}}(\boldsymbol{\sigma}^{(k)_{n+1}}, \boldsymbol{\lambda}_{n+1}^k)$

   (h) Update state variables

   $\boldsymbol{\varepsilon}_{n+1}^{p(k+1)} = \boldsymbol{\varepsilon}_{n+1}^{p(k)} + \Delta\boldsymbol{\varepsilon}_{n+1}^{p(k)}$
   $\boldsymbol{\kappa}_{n+1}^{(k+1)} = \boldsymbol{\kappa}_{n+1}^{(k)} + \Delta\boldsymbol{\kappa}_{n+1}^{(k)}$
   $\lambda_{n+1}^{i(k+1)} = \lambda_{n+1}^{i(k)} + \Delta\boldsymbol{\lambda}_{n+1}^{(k)},\ i \in J_{act}$

   (i) Set k=k+1 and goto step (b)

Table 35: General multisurface closest point algorithm

4.1.3 Algorithmic stiffness

Differentiation of the elastic stress-strain relations (110) and the discrete flow rule (111) yields

$$d\boldsymbol{\sigma}_{n+1} \quad = \quad \boldsymbol{D}\left(d\boldsymbol{\varepsilon}_{n+1} - d\boldsymbol{\varepsilon}_{n+1}^p\right) \tag{126}$$

$$d\boldsymbol{\varepsilon}^p_{n+1} = \sum \left(\lambda^i \partial_{\sigma\sigma} g d\boldsymbol{\sigma} + \lambda^i \partial_{\sigma\kappa} g \left(\partial_\sigma \boldsymbol{\kappa} d\boldsymbol{\sigma} + \partial_\lambda \boldsymbol{\kappa} d\lambda^i\right) + d\lambda^i \partial_\sigma g\right) \quad (127)$$

Combining this two equations, one obtains following relation

$$d\boldsymbol{\sigma} = \boldsymbol{\Xi}_{n+1} \left\{ d\boldsymbol{\varepsilon}_{n+1} - \sum \lambda^i \partial_{\sigma\kappa} g \partial_\lambda \boldsymbol{\kappa} d\lambda^i - \sum d\lambda^i \partial_\sigma g \right\} \quad (128)$$

where $\boldsymbol{\Xi}_{n+1}$ is the algorithmic moduli defined as

$$\boldsymbol{\Xi}_{n+1} = \left[\boldsymbol{D}^{-1} + \sum \lambda^i \partial_{\sigma\sigma} g + \sum \lambda \partial_{\sigma\kappa} g \partial_\sigma \boldsymbol{\kappa} \right] \quad (129)$$

Differentiation of discrete consistency condition yields

$$\partial_\sigma f^i d\boldsymbol{\sigma} + \partial_\kappa f^i (\partial_\sigma \boldsymbol{\kappa} d\boldsymbol{\sigma} + \partial_\lambda \boldsymbol{\kappa} d\boldsymbol{\lambda}) = 0 \quad (130)$$

By substitution of (128) into (130) the following relation is obtained

$$d\boldsymbol{\lambda} = \boldsymbol{G} \left\{ \boldsymbol{V} \boldsymbol{\Xi} d\boldsymbol{\varepsilon} \right\} \quad (131)$$

where matrix $\boldsymbol{G}$ is defined as

$$\boldsymbol{G} = \left[\boldsymbol{V}^T \boldsymbol{\Xi} \boldsymbol{U} - \partial_\kappa \boldsymbol{f} \partial_\lambda \boldsymbol{\kappa} \right]^{-1} \quad (132)$$

Finally, by substitution of (132) into (128) one obtains the algorithmic elasto-plastic tangent moduli

$$\frac{\mathrm{d}\boldsymbol{\sigma}}{\mathrm{d}\boldsymbol{\varepsilon}}|_{n+1} = \boldsymbol{\Xi} - \boldsymbol{\Xi} \boldsymbol{U} \left(\boldsymbol{V} \boldsymbol{\Xi} \boldsymbol{U} - [\partial_\kappa f][\partial_\lambda \boldsymbol{\kappa}] \right) \boldsymbol{V} \boldsymbol{\Xi} \quad (133)$$

4.1.4 Implementation of particular models

As follows from previous sections, a new plasticity based class has to provide only some model-specific services. The list of services, that should be implemented includes (for full reference, please consult documentation of MPlasticMaterial class):

- method for computing the value of yield function (computeYieldValueAt service)
- method for computing stress gradients of yield and load functions (method computeStressGradientVector)
- method for computing hardening variable gradients of yield and load functions (method computeKGradientVector)
- methods for computing gradient of hardening variables with respect to stress and plastic multipliers vectors (computeReducedHardeningVarsSigmaGradient and computeReducedHardeningVarsLamGradient methods)
- method for evaluating the increments of hardening variables due to reached state (computeStrainHardeningVarsIncrement)
- methods for computing second order derivatives of load and yield functions (computeReducedSSGradientMatrix and computeReducedSKGradientMatrix methods). Necessary only if consistent stiffness is required.

4.2 Isotropic damage model – IsotropicDamageMaterial class

In this section, the implementation of an isotropic damage model will be described. To cover the various models based on isotropic damage concept, a base class IsotropicDamageMaterial is defined first, declaring the necessary services and providing the implementation of them, which are general. The derived classes then only implement a particular damage-evolution law.

The isotropic damage models are based on the simplifying assumption that the stiffness degradation is isotropic, i.e., stiffness moduli corresponding to different directions decrease proportionally and independently of direction of loading. Consequently, the damaged stiffness matrix is expressed as

$$\boldsymbol{D} = (1-\omega)\boldsymbol{D}_e,$$

where $\boldsymbol{D}_e$ is elastic stiffness matrix of the undamaged material and ω is the damage parameter. Initially, ω is set to zero, representing the virgin undamaged material, and the response is linear-elastic. As the material undergoes the deformation, the initiation and propagation of microdefects decreases the stiffness, which is represented by the growth of the damage parameter ω. For $\omega = 1$, the stiffness completely disappears.

In the present context, the $\boldsymbol{D}$ matrix represents the secant stiffness that relates the total strain to the total stress

$$\boldsymbol{\sigma} = \boldsymbol{D}\boldsymbol{\varepsilon} = (1-\omega)\boldsymbol{D}_e\boldsymbol{\varepsilon}.$$

Similarly to the theory of plasticity, a loading function f is introduced. In the damage theory, it is natural to work in the strain space and therefore the loading function is depending on the strain and on an additional parameter κ, describing the evolution of the damage. Physically, κ is a scalar measure of the largest strain level ever reached. The loading function usually has the form

$$f(\boldsymbol{\varepsilon}, \kappa) = \tilde{\varepsilon}(\boldsymbol{\varepsilon}) - \kappa,$$

where $\tilde{\varepsilon}$ is the equivalent strain, i.e., the scalar measure of the strain level. Damage can grow only if current state reaches the boundary of elastic domain ($f = 0$). This is expressed by the following loading/unloading conditions

$$f \leq 0, \quad \dot{\kappa} \geq 0, \quad \dot{\kappa} f = 0.$$

It remains to link the variable κ to the damage parameter ω. As both κ and ω grow monotonically, it is convenient to postulate an explicit evolution law

$$\omega = g(\kappa).$$

The important advantage of this explicit formulation is that the stress corresponding to the given strain can be evaluated directly, without the need to solve the nonlinear system of equations. For the given strain, the corresponding stress is computed simply by evaluating the current equivalent strain, updating the maximum previously reached equivalent strain value κ and the damage parameter and reducing the effective stress according to $\boldsymbol{\sigma} = (1-\omega)\boldsymbol{D}_e\boldsymbol{\varepsilon}$.

This general framework for computing stresses and stiffness matrix is common for all material models of this type. Therefore, it is natural to introduce

the base class for all isotropic-based damage models which provides the general implementation for the stress and stiffness matrix evaluation algorithms. The particular models then only provide their equivalent strain and damage evolution law definitions. The base class only declares the virtual services for computing equivalent strain and corresponding damage. The implementation of common services uses these virtual functions, but they are only declared at IsotropicDamageMaterial class level and have to be implemented by the derived classes.

Together with the material model, the corresponding status has to be defined, containing all necessary history variables. For the isotropic-based damage models, the only history variable is the value of the largest strain level ever reached (κ). In addition, the corresponding damage level ω will be stored. This is not necessary because damage can be always computed from corresponding κ. The IsotropicDamageMaterialStatus class is derived from StructuralMaterialStatus class. The base class represents the base material status class for all structural statuses. At StructuralMaterialStatus level, the attributes common to all "structural analysis" material models - the strain and stress vectors (both the temporary and non-temporary) are introduced. The corresponding services for accessing, setting, initializing, and updating these attributes are provided. Therefore, only the κ and ω parameters are introduced (both the temporary and non-temporary). The corresponding services for manipulating these attributes are added and services for context initialization, update, and store/restore operations are overloaded, to handle the history parameters properly.

References

[1] Z.P. Bažant, J. Planas: Fracture and Size Effect in Concrete and Other Quasibrittle Materials, CRC Press, 1998.

[2] D.P. Bentz: CEMHYD3D: A Three-Dimensional Cement Hydration and Microstructure Development Modeling Package. Version 3.0., NIST Building and Fire Research Laboratory, Gaithersburg, Maryland, Technical report, 2005.

[3] P. Grassl and M. Jirásek. Damage-plastic model for concrete failure. *International Journal of Solids and Structures*, 43:7166–7196, 2006.

[4] M. Jirásek, Z.P. Bažant: Inelastic analysis of structures, John Wiley, 2001.

[5] M. Jirásek: Comments on microplane theory, Mechanics of Quasi-Brittle Materials and Structures, ed. G. Pijaudier-Cabot, Z. Bittnar, and B. Gérard, Hermès Science Publications, Paris, 1999, pp. 57-77.

[6] B. Lourenco, J.G. Rots: Multisurface Interface Model for Analysis of Masonry Structures, Journal of Engng Mech, vol. 123, No. 7, 1997.

[7] M. Ortiz, E.P. Popov: Accuracy and stability of integration algorithms for elasto-plastic constitutive relations, Int. J. Numer. Methods Engrg, 21, 1561-1576, 1985.

[8] B. Patzák: OOFEM home page, http://www.oofem.org, 2003.

[9] J.C. Simo, T.J.R. Hughes: Computational Inelasticity, Springer, 1998.

[10] J. Ruiz, A. Schindler, R. Rasmussen, P. Kim, G. Chang: Concrete temperature modeling and strength prediction using maturity concepts in the FHWA HIPERPAV software, 7th international conference on concrete pavements, Orlando (FL), USA, 2001.

[11] J.C. Simo, J.G. Kennedy, S. Govindjee: Non-smooth multisurface plasticity and viscoplasticity. Loading/unloading conditions and numerical algorithms, Int. J. Numer. Methods Engrg, 26, 2161-2185, 1988.

[12] J.C. Simo, K.S. Pister: Remarks on rate constitutive equations for finite deformation problems: computational implications, Comp Methods in Applied Mech and Engng, 46, 201-215, 1984.

[13] V. Šmilauer and T. Krejčí, Multiscale Model for Temperature Distribution in Hydrating Concrete, International Journal for Multiscale Computational Engineering, 7 (2), 135-151, 2009.