# A network evolution model for the anisotropic mullins effect

## Introduction

The constitutive behvior of  filled elastoemrs under large deformation behavior features several inelastic effects such as the  Mullins effect, permanent set, and induced anisotropy.
A Network evolution model was proposed  [1] to describes those the inelastic features. The procedure of implementation into FE softwares were developed later [2], in which  the network evolution model calculates the contribution of the inelastic features based on the free energy function. This code is organized as Table 1.

| | |
|---|---|
| 1. | Problem Description of the Induced Anisotropy |
| 2. | Statistical mechanics of a single chain |
| 3. | Aggregate-polymer debonding |
| 4. | Network rearrangement |
| 5. | Numerical integration |
| 6. | Strain amplification |
| 7. | Three-dimensional generalization |
| 8. | Macroscale behavior |
| 9. | Plot an Example |

**Table 1**  Essential steps in an analytical solution of network evolution model

Here, the network evolution model is coded in Maple (R). An example is also provided to show how the  constitutive modelbehavior and its inelastic features such as induced anisotropy and permanent set has been calculated.

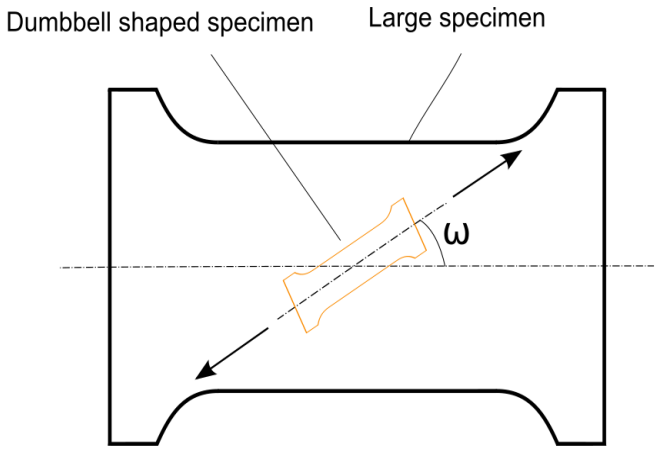
## Initializations

Load packages and define initial data by clicking on the icon below.

# Problem Description of the Induced Anisotropy

Here, the concept of induced anisotropy is coded as the changes in the properties of the material in specific direction $\omega$ after the material was subjected to uniaxial tensions in other directions. To model this, we assume that each time after the eleongation in one direction, the next loading in direction $\omega$ will be appleid on a sample cut from the initial sample in direction $\omega$ as depicted in figure below. In the following experiment, history of streches in different directions stored and used for calculation of constitutive model.

**Note:** This algoritm can be used for any specific sample or test.




Deformation gradiant for sample loaded in direction ($\omega$),

$$\hat{F} := \left( \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \frac{1}{\sqrt{\lambda}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{\lambda}} \end{bmatrix} \right) :$$

$$RM := (Student[LinearAlgebra][RotationMatrix](\text{convert}(\omega \text{ *degrees, radians}), \text{`<,>`}(0,0,1))) :$$

$$FF := unapply(RM^{+}.\hat{F}.RM, \lambda, \text{omega}) :$$

where $RM^{+}$ is the tranpose pf the rotation matrix RM.

As an example, second loading is asumed to be applied in ***y directin.***

$$\omega_0 := 90 :$$

```
F := FF(λ, ω_0);
J := LinearAlgebra[Determinant](F):
```

$$\begin{bmatrix} \frac{1}{\sqrt{\lambda}} & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \frac{1}{\sqrt{\lambda}} \end{bmatrix} \quad \textbf{(3.1)}$$

The history of streches at different directions are stored in two vector ($\boldsymbol{\lambda}_{his}$,$\boldsymbol{\omega}_{his}$). In the case of induced anisotropy, number of loading in differnt directions in the past, n_history, the direction of loading, omega_histroy, and their correponding stretch histrory, lambda_histroy, are given as

```
n_history := 1:
λ_history := [1]:
ω_history := [0]:
```

# Statistical mechanics of a single chain

According to [1], the probability of existence of a polymer chain in which none of the segments between numbers 1 and n are attached to an aggregate surface, is given by:

```
A := -alpha*Rbar^2/n - kappa*sqrt(alpha/Pi)*(2*sqrt(n)
    *exp(-alpha*Rbar^2/n) + 2*Rbar*sqrt(Pi*alpha)*(erf(Rbar
    *sqrt(alpha/n)) - erf(Rbar*sqrt(alpha)))) - 2*exp(-alpha
    *Rbar^2)):
P := unapply(kappa*sqrt(alpha/Pi/n)*exp(A), n, Rbar);
```

$$(n, \bar{R}) \quad \textbf{(4.1)}$$

$$\rightarrow \kappa \sqrt{\frac{\alpha}{\pi n}}$$

$$e^{-\frac{\alpha \bar{R}^2}{n}} - \kappa \sqrt{\frac{\alpha}{\pi}} \left( 2\sqrt{n}\, e^{-\frac{\alpha \bar{R}^2}{n}} + 2\bar{R}\sqrt{\pi\alpha}\left(\operatorname{erf}\left(\bar{R}\sqrt{\frac{\alpha}{n}}\right)\right.\right.$$

$$-\operatorname{erf}\left(\bar{R}\sqrt{\alpha}\right)\Big) - 2\,e^{-\alpha \bar{R}^2}\Bigg)$$

In this equation,

$\kappa$= the average area of active adsorption sites available for one bond

$\bar{R}$=normalized end-to-end distance with respect to the segment length (R/l)

and

for the valence angle $\theta = 70.5°$.

```
alpha := ((1-cos(θ)) / (1+cos(θ)))^2;
θ := convert(70.5*degrees, radians): alpha := 0.27:
```

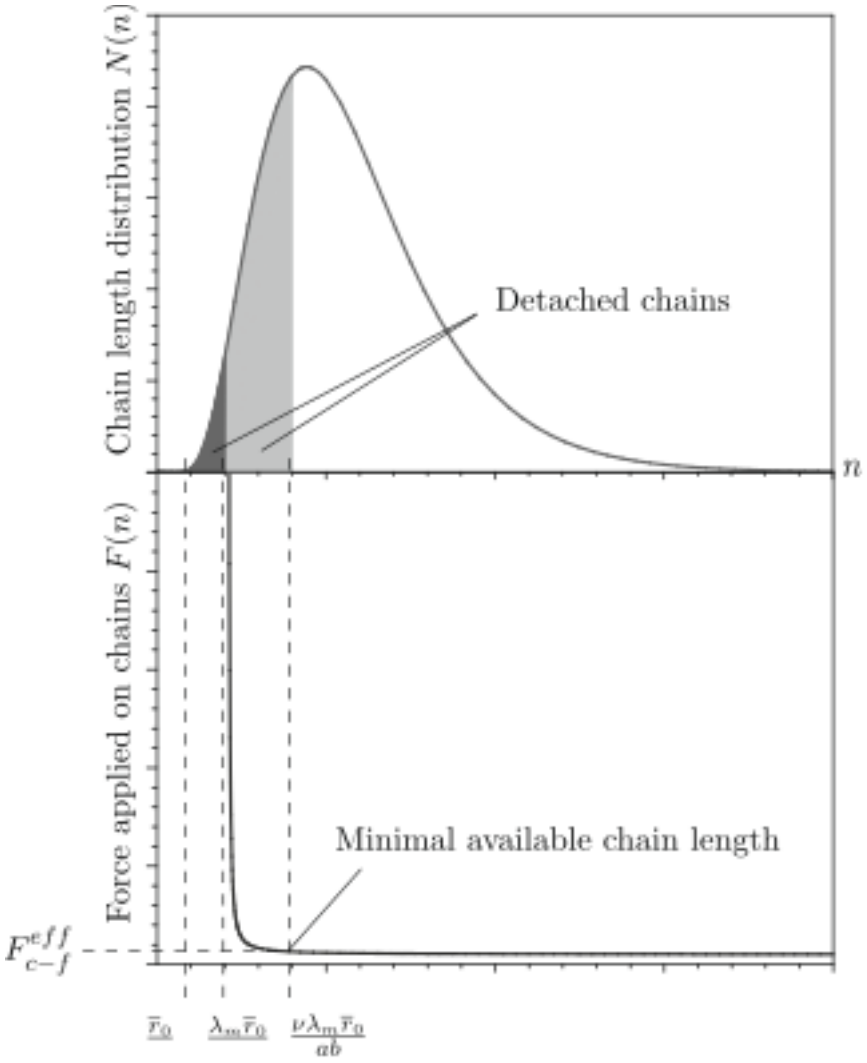
$$\frac{(1-\cos(\theta))^2}{(1+\cos(\theta))^2} \qquad \textbf{(4.2)}$$

# PP Netowrk

It host all damages, The damages are considered to be resulted from two simoltanous mechanisms

## 1.Aggregate-polymer debonding

In the course of deformation, polymer chains begin to slide on or debond from the aggregates. This debonding starts with the shortest chain and gradually involves longer and longer chains.




The length of the shortest available chain in the deformed subnetwork is then obtained by

$$n_{cf} := \frac{\nu \cdot \overset{d}{\lambda}_{\max} \cdot R0}{a\,b} :$$

In aforementioned equation, $\nu > 1$ denotes a sliding ratio and is a material parameter and $\overset{d}{\lambda}_{\max}$ denotes the maximal micro-stretch reached in direction ***d***. upper bond of the relative length is $n_{\max}$ (material parameter).

Accordingly, the set of available relative lengths of chains bounded to aggregates in the direction d can be expressed by

$$DD\left(\overset{d}{\lambda}_{\max}\right) = \left\{ n \,\middle|\, \nu \frac{\overset{d}{\lambda}_{\max} R0}{a\,b} \le n \le n_{\max} \right\}$$

$$DD := unapply\left(n_{cf}..n\_max, \overset{d}{\lambda}_{\max}\right);$$

$$a := \frac{\alpha^2}{\cos\left(\frac{\theta}{2}\right)} : a := 2.44 :$$

$$b := \frac{\cos\left(\frac{\theta}{2}\right)^2}{\alpha^2} : b := 0.33 :$$

$$y1 \rightarrow \frac{\nu\, y1\, R0}{a\, b} ..n\_max \qquad \textbf{(5.1.1)}$$

## 2. Network rearrangement

The concept of chain rearrangement in a rubber network suggests that the detachment of chains from the aggregate surface does not necessarily result in the complete loss of their role in the network entropic energy, but it may also lead to the activation of some new segments. Thus, all in all, one can assume that the total number of active segments remains constant. This assumption yields

$$\hat{N}\left(n, \bar{r}\right) = N_0 \Phi\left(\overset{d}{\lambda}_m\right) \quad P\left(n, \bar{r}\right) = N_0 \Phi\left(\overset{d}{\lambda}_m\right) \hat{P}(n)$$

where $\Phi(\overset{d}{\lambda}_{\max})$ ,

$$\text{Phi} := unapply(1 / int(N^* P(N, R0), N = DD(x)), x) :$$

$$\hat{N} := unapply\left(N0 \cdot \text{Phi}\left(\overset{d}{\lambda}_{\max}\right), \overset{d}{\lambda}_{\max}\right);$$

$$y1 \rightarrow N0 \Big/ \left( \int_{1.241927472\, \nu\, y1\, R0}^{n\_max} \right. \qquad \textbf{(5.2.1)}$$

$$0.2931615071\, N \kappa \sqrt{\frac{1}{N}}\; e^{-\frac{0.27\, R0^2}{N}} - 0.2931615071\, \kappa \left( 2\sqrt{N}\, e^{-\frac{0.27\, R0^2}{N}} + 1.841988074\, R0 \left( \operatorname{erf}\left( 0.5196152423\, R0 \sqrt{\frac{1}{N}} \right) - \operatorname{erf}(0.5196152423\, R0) \right) - 2\, e^{-0.27\, R0^2} \right) dN$$

In this equation, We consider R0 and N0 as material constants.
So, we can calculate the energy of a subnetwork of PP network in direction ***d*** as:

$$\overset{d}{\Psi} = N_0\, \Phi\left( \overset{d}{\lambda}_m \right) \int_{DD\left( \overset{d}{\lambda}_{\max} \right)} \hat{P}(n)\, \psi_c\left( n, \bar{R}\, \overset{d}{\lambda} \right) dn$$

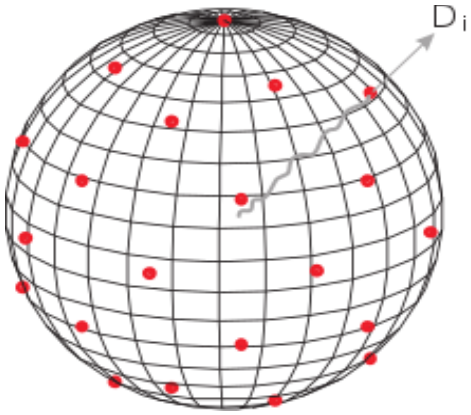
# Numerical integration

For 3D generalization, we need to sum microscopic strain energies of all active chains available within the network which can be calculated by integration over the unit sphere.
Integeration over the unit sphere can be carried out numerically by

$$\Psi_0 = \frac{1}{A_s} \int_s W_0^d\, du^d = \sum_{i=1}^{k} W_0^{d_i} w_i$$

where $\omega_i$: are the weight factors corresponding to the collocation directions $d_i$: (i =1.. k). A set of k = 21 integration points on the half-sphere is chosen

```
Int_data := Matrix(readdata("intpoints.txt", 4)) :
x_c, y_c, z_c, w_c := Column(Int_data, [1..4]) :
Num_c := Dimension(w_c) :
```




## Strain amplification

In a filler-reinforced rubber network, the stretch of the polymer chains between aggregates (micro-stretch) generally exceeds the stretch applied to the rubber matrix (macrostretch).
This strain amplification concept is based on the fact that filler aggregates are considerably stiffer than the polymer chains connecting them together.

$$\chi := unapply(\lambda\,(1 - Cc^{Pp}) + Cc^{Pp}, \lambda);$$
$$Pp := \frac{1}{3}:$$
$$Cc := 0.2:$$
$$\lambda \rightarrow \lambda\,(1 - Cc^{Pp}) + Cc^{Pp} \qquad \textbf{(7.1)}$$

## Three-dimensional generalization

- Calculation of parametrs in direction $d$

In order impliment model, we should calculate stretch 'lambda_i' and maximum stretch 'lambda_i_m' in the integration directions.

$$d := Vector([x, y, z]):$$
$$DOD := unapply(OuterProductMatrix(d, d), x, y, z):$$
$$\lambda\_d := unapply(\mathrm{sqrt}(Transpose(d).F.F.d), \lambda, x, y, z):$$
$$\lambda\_i := seq(\lambda\_d(\lambda, x_c[j], y_c[j], z_c[j]),\ j = 1..Num_c):$$

$$\lambda\_di\_max := unapply(\mathrm{sqrt}(Transpose(d).FF(\lambda m, \Omega).FF(\lambda m, \Omega).d), \lambda m, \Omega, x, y, z):$$

**if** $n\_history \geq 1$ **then**

$\lambda\_i\_max := seq(\max(1, seq(evalf(\lambda\_di\_max(\lambda\_history[k], \omega\_history[k], x_c[j], y_c[j], z_c[j])), k = 1..n\_history)), j = 1..Num_c)$ :

**else**

$\lambda\_i\_max := seq(1, j = 1..Num_c)$ :

**end if**:

$\lambda\_di\_m := seq(\lambda\_di\_max(\lambda m, \omega_0, x_c[j], y_c[j], z_c[j]), j = 1..Num_c)$ :

$\lambda\_i\_m := seq((\max(\lambda\_i\_max[j], \lambda\_i[j], \lambda\_di\_m[j])), j = 1..Num_c)$ :

# Macroscale behavior

Based on [2], The constitutive equation for the first Piola–Kirchhoff stress tensor P can be written as.

$$P = \sum_{i=1}^{k} (P_{cc}(\mathrm{D}_i) + P_{pp}(\mathrm{D}_i)) \frac{w_i}{d_i} J^{-\frac{1}{3}} \mathbf{F} : (\mathrm{D}_i \times \mathrm{D}_i)$$

where

$$P_{cc}(x) = N_c \sqrt{bn_c}\, L^{-1}\left(\frac{\chi}{\sqrt{bn_c}}\right),$$

$$P_{pp}(x) = \frac{\bar{R}}{a} \frac{N_0}{1 - C^p} \Phi\left(\frac{\chi}{\lambda_m}\right) \int_{DD\left(\frac{\chi}{\lambda_{max}}\right)} \hat{P}(n)\, L^{-1}\left(\frac{\bar{R}\frac{\chi}{\lambda}}{abn}\right) \mathrm{d}n$$

Inverse Langevin function, a proper approximation approach for the inverse Langevin function can be chosen depending on the elongation range of polymer chains, Padé approximants show better agreement with the exact values when $\upsilon < 1.04$. Generally in filled elastomers the initial guess is $\nu \sim 1.01$

**if** $\nu < 1.04$ **then**

$L_{inv} := (x) \rightarrow \frac{3 \cdot x}{1 - x^3}$ :

**else**

$$L_{inv} := (x) \to 3 \cdot x + \frac{9}{5} \cdot x^3 + \frac{297}{175} \cdot x^5 + \frac{1539}{875} \cdot x^7 + \frac{126117}{67375} \cdot x^9 + \frac{43733439}{21896875} \cdot x^{11} + \frac{231321177}{109484375} \cdot x^{13} + \frac{20495009043}{9306171875} \cdot x^{15} + \frac{1073585186448381}{476522530859375} \cdot x^{17} + \frac{4387445039583}{1944989921875} \cdot x^{19} :$$

**end if:**

```
L__inv := proc (x) options operator, arrow; 3*x/(1-x^3) end proc
```

$$x \to \frac{3\,x}{1 - x^3} \qquad \textbf{(9.1)}$$

$$P_c := seq\left( Nc * \sqrt{b * nc} * L_{inv}\left( \frac{\chi(\lambda\_i[j])}{\sqrt{b * nc}} \right) , j = 1 .. Num_c \right) :$$

$$P_p := seq\left( \frac{R0}{a} \cdot \frac{N0}{1 - Cc^{Pp}} \cdot \mathrm{Phi}(\lambda\_i\_m[j]) \cdot int\left( L_{inv}\left( \frac{R0 * \lambda\_i[j]}{a \cdot b \cdot n} \right) * P(n, R0), n = DD(\lambda\_i\_m[j]) \right), j = 1 .. Num_c \right) :$$

Due to the incompressibility of elastomers, one can use the following lagrange multiplier, p, to satisfy the icompressibility condition

$$T = \frac{\partial \Psi_m}{\partial F} - p\,F^{-T} = \frac{\partial \Psi_{pp}}{\partial F} + \frac{\partial \Psi_{cc}}{\partial F} - p\,F^{-T}$$

For uniaxial test sample in ω direction, we have $T_{33} = 0$,

$$DOD\_p := F.DOD(x, y, z) :$$
$$p1 := unapply(DOD\_p[3, 3] \cdot F[3, 3], x, y, z) :$$
$$DD\_n := unapply\left( eval\left( (RM^{+} . (F.DOD(x, y, z) - p1(x, y, z) \cdot MatrixInverse(F^{+}) ) . RM)[1, 1], \omega = -\omega_0 \right), x, y, z \right) :$$
$$DOD\_i := seq\left( DD\_n\left( x_c[j], y_c[j], z_c[j] \right) , j = 1 .. Num_c \right) :$$

where DOD_p represents the tensor $\mathbf{F} : (\mathrm{D}_i \times \mathrm{D}_i)$, pl the lagrange ultiplier p, DD_n is the 1,1 component of the tensor $\mathbf{F} : (\mathrm{D}_i \times \mathrm{D}_i)$ at the current loading direction in the global coordinates. Here, DOD_i represents the stretch field of the current sample which is stretched in direction D.

Summation of caculated stress in different direction

Error, (in LinearAlgebra:-&x) invalid input: LinearAlgebra:-`&x` expects its 1st argument, V1, to be of type Vector(3) but received D__i

$$P\_n := seq\left(\left((P_c[k] + P_p[k]) \cdot \frac{w_c[k]}{\chi(\lambda\_i[k])} \cdot J^{-\frac{1}{3}} \cdot DOD\_i[k]\right), k = 1..Num_c\right):$$

$$Pt := \sum_{i=1}^{Num_c} P\_n[i]:$$

$$Fist\_piola := unapply(Pt, \lambda, \lambda m):$$

where P_n[i] reresents the stress component in each integration direction.

# Plot an Example

Here we used the data from tabel 3 reference [2] to illustrate capability of this algorithm.
In this part we assumed material parameters as bellow an ploted stress and strain graph.

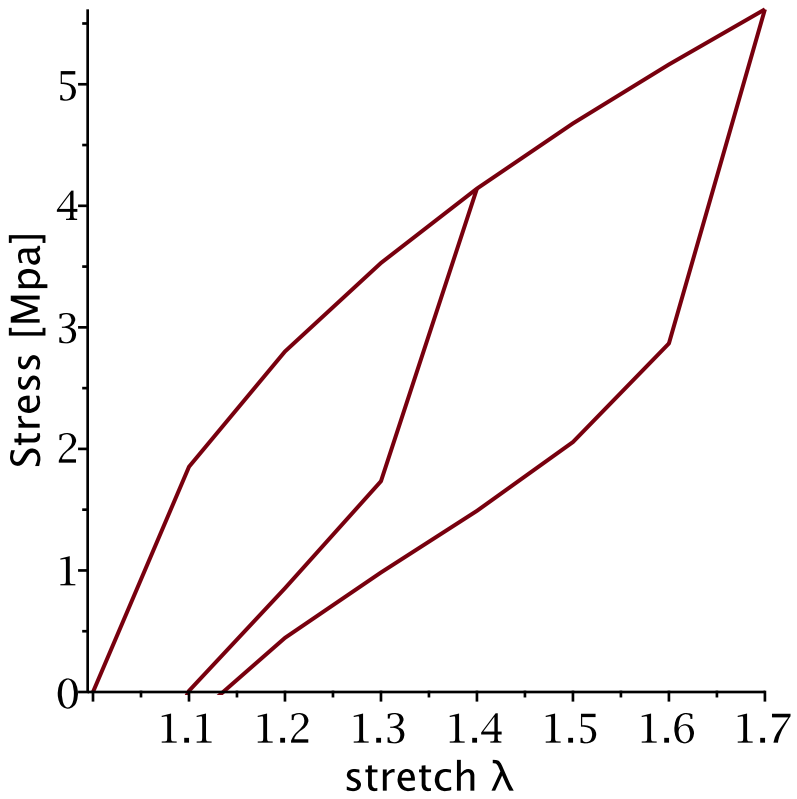
```
κ := 14.824:
ν := 1.0065:
n_max := 100:
R0 := 6.406:
Nc := 1.8141:
nc := 100:
N0 := 2.835:
```

```
plot_interval := 0.1:
Final_stretch := 1.7:
sigma_max := Fist_piola(Final_stretch, Final_stretch):
unloading_lim_1 := 1.7:
unloading_lim_2 := 1.4:

F1[1] := plot({seq([1 + x*plot_interval, evalf(Fist_piola(1
    + x*plot_interval, 1 + x*plot_interval))], x = 0
    ..(Final_stretch-1) / plot_interval)}, x = 1..Final_stretch,
    y = 0..sigma_max):
F1[2] := plot({seq([1 + x*plot_interval, evalf(Fist_piola(1
    + x*plot_interval, unloading_lim_1))], x = 0
```

```
    ..(unloading_lim_1-1) / plot_interval)}, x = 1
    ..Final_stretch, y = 0..sigma_max) :
F1[3] := plot({seq([1 + x* plot_interval, evalf(Fist_piola(1
    + x* plot_interval, unloading_lim_2))], x = 0
    ..(unloading_lim_2-1) / plot_interval)}, x = 1
    ..Final_stretch, y = 0..sigma_max) :
```

```
display(F1[kk] $ kk = 1..3, labels = ["stretch λ",
    "Stress [Mpa]"], labeldirections = ["horizontal",
    "vertical"], labelfont = ["HELVETICA", 12]) ;
```




## Refrences


[1] Mikhail Itskov. "A network evolution model for the anisotropic Mullins effect in carbon black filled rubbers." International Journal of Solids and Structures 46, no. 16 (2009): 2967-2977.

[2] Vu Ngoc Khiêm, Uwe Navrath, and Mikhail Itskov. "Network evolution model of anisotropic stress softening in filled rubber-like materials: Parameter identification and finite element implementation." Journal of Mechanics of Materials and Structures 7, no. 8 (2013): 861-885.