



---

*Research article*

## **Anisotropic peridynamic model—Formulation and implementation**

**Aylin Ahadi\*** and **Jakob Krochmal**

Division of Mechanics, Department of Mechanical Engineering Lund University, P.O. Box 118, S22100 Lund, Sweden

\* **Correspondence:** Email: [aylin.ahadi@mek.lth.se](mailto:aylin.ahadi@mek.lth.se); Tel: +46462223039; Fax: +46462224620.

**Abstract:** In this work, anisotropy is introduced in a peridynamic model. The spherical influence function is replaced by an ellipsoidal influence function. The model is mathematically formulated and implemented into the source files of LAMMPS, extending the program in a straight-forward manner. The extension can be applied to other peridynamic model and here it is introduced into elastoplastic peridynamic model in LAMMPS. The implemented model is tested through simulating beams loaded in compression. The model is found to alter the material behavior in the simulations compared to the original isotropic mode. A clear qualitative and quantitative difference in behavior is shown independently of pre-existing simulation parameters. The model is shown to be internally consistent. Finally we also demonstrated that the model can easily be extended to include several preferable direction opening for application as modeling elasticplastic deformation of anisotropic heterogeneous crystalline matter.

**Keywords:** peridynamics; anisotropy; implementation; PDLAMMPS

---

### **1. Introduction**

Material properties emerge from phenomena on scales ranging from angstroms to meters. A multiscale treatment can provide a basis for an understanding of material behavior on different scales. The traditional multiscale approach couples two models operating at different scales. Alternative modeling strategy is called peridynamics, a novel non-local continuum model approach suggested by Silling, see [1], that models materials on a per-particle basis. This approach continualize the discrete models such as molecular dynamic models, by replacing inhomogeneities present on smaller length scales by an enhanced continuum description on larger length scales. Peridynamic models converge to the classical elastic material model, it can be used as an alternative approach for multiscale modeling as a single multiscale model valid over wide range of length scales, see [2] and [3]. In that way peridynamic can handle length scales beyond those typical for molecular dynamic models, opening

for new possibilities for application to systems at mesoscale and macroscale. Later developments of the original theory, called state based theory of peridynamics, see [4], showed that any constitutive model from the classical continuum theory can be adapted to this state based theory of peridynamics.

The force interactions resulting from the discretized peridynamic equations are very similar to traditional molecular dynamic formulations, and this suggests that peridynamics can be implemented within a molecular dynamics framework code, see [5]. It was shown in [6] showed that peridynamics can be considered as an upscaling of molecular dynamics. The molecular dynamics freeware LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator), see [7], will be used for the simulations since this code also supports the peridynamics framework, see [8]. LAMMPS is a molecular dynamics simulator which is shipped with a peridynamics package PDLAMMPS, developed by Sandia corporation, allowing it to take a peridynamic approach to materials analysis. In addition to the bond based peridynamic model implemented as prototype microelastic brittle model (PMB), see [4] and [5], there are models for the state-based peridynamic linearly elastic (LPS), elastic-plastic (EPS) and viscoelastic (VES), see [5, 9] and [10] solids implemented. Details about the implementation of the elastic-plastic and viscoelastic peridynamic models in PDLAMMPS can be found [11–13].

The peridynamic package in PDLAMMPS allows for modelling material restricted to isotropic behavior of material, limiting the application of the peridynamic theory. This creates a need for extending the existing models to include such behavior. An anisotropy extension of the elasto-plastic peridynamic model is presented in this work. The objective has been to develop anisotropic peridynamic model that easily can be implemented into PDLAMMPS. This was achieved by modifying the spherical influence function in the peridynamic models. A non-spherical influence function was introduced and implemented, i.e., the forces are not modulated based on the internal distance of the acting particles, but based on the direction of their relative position. This way, anisotropic materials can be modeled by aligning the model with coordinate system and thereafter instruct the influence function to modulate the forces differently depending on what direction they have relative to this coordinate system.

The model is introduced into source files of LAMMPS itself, extending the program in a straightforward manner. The influence function being implemented takes internal forces of the object and truncates their magnitudes according to their direction in the coordinate system. The capabilities to represent directional dependent behavior are shown through simulating beams loaded in compression. The model is shown to be internally consistent, i.e., returning the the same results for the elastic-plastic (EPS) model as a spacial case of the newly implemented anisotropic elastic-plastic model (AEP). Clear qualitative differences in behavior is shown independently of pre-existing simulation parameters.

## 2. Theory of peridynamics

In the peridynamic theory the interactions between material points are expressed in terms of bond forces, permitting interaction between particles at a distance, in contrast to the classical continuum mechanics, where interactions between material points are expressed in terms of traction vectors, i.e., we assume a local interaction. Peridynamics treats internal forces within a continuous solid as a network of pair interactions, similar to springs. Each material point interacts with its neighborhood within a sphere of radius  $\delta$ , called the horizon, that serves as an internal length scale in the model.

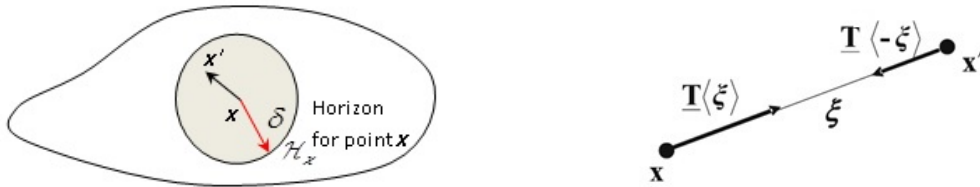
Let  $\mathbf{x}$  be the position vector of the studied point and  $\mathbf{x}'$  the position vector of a point in its neighborhood, see Figure 1. The relative position of the two particles in the reference configuration is denoted by  $\xi$  and the relative displacement vector with  $\eta$ ,

$$\xi = \mathbf{x}' - \mathbf{x} \quad , \quad \eta = \mathbf{u}' - \mathbf{u} \quad (1)$$

where  $\mathbf{u}'$  and  $\mathbf{u}$  are the displacement vectors of the two points.  $H_x$  is a sphere whose center is the studied point  $\mathbf{x}$ . The radius  $\delta$  is the horizon of  $\mathbf{x}$  and its horizon is the distance limit across which a pair of material points can interact, and  $\delta$  can be interpreted as the length scale in the model, defined in the reference configuration in contrast to the cut-off radius in molecular dynamics, which is defined in the deformed configuration. The point  $\mathbf{x}$  interacts directly with the points in the sphere. The interactions with other points are considered negligible. In state-based peridynamics the equation of motion of a material point at  $\mathbf{x}$  then reads

$$\rho(\mathbf{x})\ddot{\mathbf{u}}(\mathbf{x}, t) = \int_{H_x} \{\mathbf{T}[\mathbf{x}, t] \langle \mathbf{x}' - \mathbf{x} \rangle - \mathbf{T}[\mathbf{x}', t] \langle \mathbf{x} - \mathbf{x}' \rangle\} dV_{x'} + \mathbf{b}(\mathbf{x}, t) \quad (2)$$

Here,  $\mathbf{T}[\mathbf{x}, t]$  is a force vector state field,  $\mathbf{b}(\mathbf{x}, t)$  is the external body force density field,  $\rho(\mathbf{x})$  is the mass density in the reference configuration and  $\mathbf{u}(\mathbf{x}, t)$  is the displacement field. To ensure consistency with basic physical principles, it will be required that  $\mathbf{T}[\mathbf{x}, t]$  satisfy balance of linear momentum for any bounded body. A limiting process  $\delta \rightarrow 0$  produces a classical constitutive model for the Piola stress, see [14]. The force vector state field  $\mathbf{T}[\mathbf{x}, t]$  contains entirely the constitutive model. The force vector state  $\mathbf{T}[\mathbf{x}, t] = t\mathbf{M}$  can be interpreted in term of mapping the bond between two point to a force per volume, where  $t$  is the force scalar state and  $\mathbf{M}$  is the deformed state.



**Figure 1.** Schematic description of peridynamic body and a force vector state field.

We introduce a nonnegative scalar state defined on  $H_x$  by  $x\langle\xi\rangle = \|\xi\|$ . The elastic-plastic force scalar state  $t$  is split into two parts, the volumetric and deviatoric scalar force states

$$t = t_i + t_d = -\frac{3K\theta}{m}\omega x + \alpha\omega(e_d - e_{dp}) \quad (3)$$

where  $K$  is the bulk modulus,  $\alpha$  is material constants,  $e_d, e_{pd}$  are the deviatoric and plastic extensions of the material, respective, whereas  $\omega$  is the influence function. The weighted volume  $m$  is defines as

$$m(\mathbf{x}) = \int_{H_x} \omega\langle\xi\rangle x\langle\xi\rangle x\langle\xi\rangle dV_\xi \quad (4)$$

The dilation  $\theta$  is

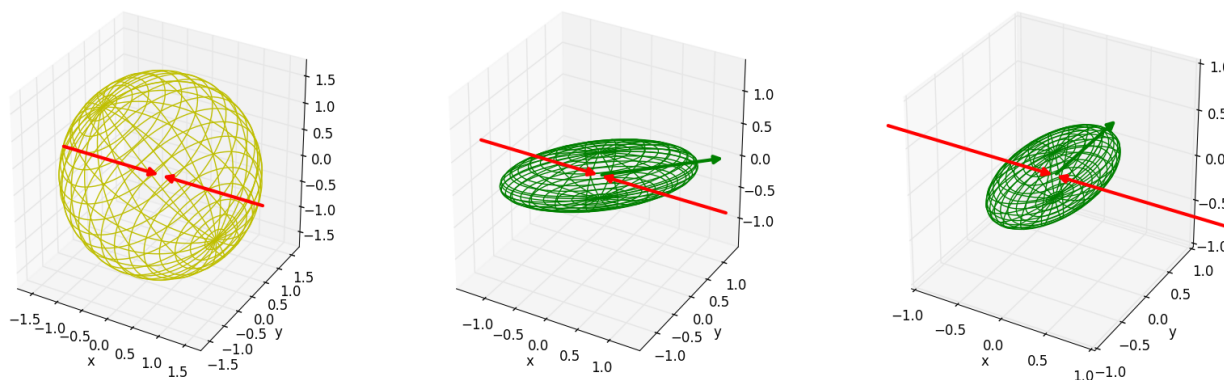
$$\theta(\mathbf{x}, t) = \frac{3}{m(\mathbf{x})} \int_{H_x} \omega\langle\xi\rangle x\langle\xi\rangle e(\mathbf{x}, t) dV_\xi \quad (5)$$

$e(\mathbf{x}, t)$  is extension scalar field defined as  $e(\mathbf{x}, t)\langle\xi\rangle = \|\xi + \eta\| - \|\xi\|$ .

### 3. Anisotropic influence function

The influence function  $\omega$  in the peridynamic theory is used to weight the contribution of all the bonds. It regulates how the particles interact with each other and was first presented as a boolean function returning a value of either 0 or 1 for each bond, modulating the forces by simply setting them to zero in certain instances and leaving them unchanged in others, and were first introduced in the modern state-based formulation of peridynamics by. Different material behavior can be modeled by letting the influence function modulate the forces acting on particles in the simulation based only on their distance from each other. Such an influence function is called spherical, as forces have the same magnitude in every direction from the particle. In [15], it was also demonstrated that influence functions can be used to modulate nonlocal effects within a peridynamic model independently of the peridynamic horizon. Details on spherical influence functions, can be found in [6] and [15].

In this work, a non-spherical influence function, see Figure 2, was implemented in a peridynamic model—the forces are not modulated based on the internal distance of the acting particles, but based on the direction of their relative position as seen relative to some coordinate system. This way, anisotropic materials can be modeled by aligning the model with coordinate system and thereafter let the influence function to modulate the forces differently depending on what direction they have relative to this coordinate system. This isotropic model is modified to include anisotropy by introducing the ellipsoidal influence function and the rest of the model is unchanged, i.e., the same flow rule as in the original models is used.



**Figure 2.** Left: Isotropic influence function; Middle: Anisotropic influence function, major axis of the ellipsoid oriented at a  $45^\circ$  angle relative to the loading direction; Right: Anisotropic influence function, major axis perpendicular to the loading direction. Green arrow indicates the proffered direction and red line indicate the loading direction.

The influence function being implemented takes internal forces of the object and truncates their magnitudes according to their direction in the coordinate system as follows: if the force  $\mathbf{f}$  is directed exactly along a predetermined preferred direction  $\mathbf{u}$ , then  $\mathbf{f}$  will not be modulated at all. If the force is instead directed along a second direction  $\mathbf{v}$ , orthogonal to  $\mathbf{u}$ , the force vector will be multiplied by a predetermined scalar  $a < 1$ . If the force is directed along a third direction  $\mathbf{w}$ , orthogonal to both  $\mathbf{u}$ ,  $\mathbf{v}$ ,

then the force vector will be multiplied by a second predetermined scalar  $b < 1$ . If  $\mathbf{f}$  instead points in any other direction, then  $\mathbf{f}$  will be modulated such that its magnitude will be the distance from the center of an ellipsoid to its edge, said ellipsoid being defined as being centered in the origin, having a major axis with length  $\|\mathbf{f}\|$  pointing in the direction of the preferred direction  $\mathbf{u}$  and semi-axes of length  $a \cdot \|\mathbf{f}\|$ ,  $b \cdot \|\mathbf{f}\|$ . Finally, the semi-axes are rotated about the vector  $\mathbf{u}$  by some angle  $\theta$ . Thus, the only calculation to consider is that of finding the distance from the origin of a cartesian 3-d coordinate system to any ellipsoid centered in said origin.

### 3.1. Algorithm Design

The approach taken is return the distance to the edge of an *ellipsoid* with the major semi-axis being the length of the incoming force vector and having a predefined direction, the other semi-axes having a lengths some fraction of the major semi-axis. The pre-defined direction of the major semi-axis we call the *preferred* direction, and should correspond to the preferred direction of deformation of the solid being modelled. The ellipsoids are defined by inputting the preferred direction along with real numbers  $< 1$  for the two minor semi-axes and a rotation about its own major semi-axis defined in radians. The algorithm calculates extrinsic rotations to turn the ellipsoid so that its major semi-axis points in the preferred direction of the solid, and then rotates it intrinsically (that is, rotates it around its own major semi-axis) to match the input rotation. This results in a  $3 \times 3$  rotation matrix  $M$ . The LU factorization of this matrix is then calculated, and the results stored in a single  $3 \times 6$  matrix. It is this LU factorization that is used in the calculations once the simulation is run. The distance from the center of the ellipsoid to its surface is then calculated as explained in the next section for each bond interaction in the simulation. As the ellipsoid is scaled such that the longest such distance is along its major semi-axis and is the exact length of the force vector  $\mathbf{f}$  being subjected to the function, it is obvious that the distance  $d \leq \|\mathbf{f}\|$  depends on the direction of  $f$  relative to the pre-defined preferred direction of the material. This will lead to larger internal forces in the preferred direction of the material and thus smaller deformations and stresses, along with larger deformations and stresses in other directions. A material can have more than one preferred direction—the algorithm simulates this by storing several ellipsoids, running the scaling routine on each one and then returning the largest value. An even more useful algorithm would weight the ellipsoids somehow, and use a smarter selection of which ellipsoid should be used to modulate the force vector.

### 3.2. Mathematics

The mathematical problem to be solved can be stated as follows: given an arbitrarily rotated ellipsoid in euclidean 3d-space, with arbitrarily chosen semi-axes, find where an arbitrary vector  $\mathbf{v}$  intersects the surface of the ellipsoid if the length of  $\mathbf{v}$  is also the length of the major semi-axis of the ellipsoid, thus ensuring that it will always intersect the ellipsoid. The method to do this is comprises two steps. First change basis so that the semi-axes of the ellipsoid becomes parallel with the basis vectors of the new coordinate system. Then normalize  $\mathbf{v}$  and then using the equation for an ellipsoid find a scalar  $d$  such that  $d \cdot \mathbf{v}$  lies on the surface of the ellipsoid. More detail regarding the mathematics can be found in [16].

The first step is accomplished as follows: let  $\mathbf{a} = (x_a, y_a, z_a)$  be the vector that constitutes the major semi-axis of the ellipsoid. Now, calculate how to rotate the basis vector  $\mathbf{e}_x$  so that  $(c \cdot \mathbf{e}_x) = \mathbf{a}$  for an

arbitrary scalar  $c$ :

$$Y = \arcsin\left(\frac{y_a}{\|\mathbf{a}\|}\right), \quad P = \arcsin\left(\frac{z_a}{\|\mathbf{a}\|}\right) \quad (6)$$

where  $P$  is the “pitch” and  $Y$  is the “yaw” of  $\mathbf{a}$ , ie the angles of  $\mathbf{a}$  with respect to the basis of the coordinate system (“pitch” is what angle an airplane flying along  $\mathbf{e}_x$  would have to have with the ground and “yaw” what angle it would have to turn left-right to start flying along  $\mathbf{a}$ ). The ellipsoid is also rotated along its own major semi-axis, that is, around  $\mathbf{a}$ —this angle we call  $R$  for “roll”. These are the so called Eulers angels. We can create a rotation matrix to perform a rotation corresponding to each of these intrinsic rotations. These we call  $M_P, M_Y, M_R$ . An arbitrary rotation  $M$  may be written as a product of three simple rotations:

$$M = M_Y \cdot M_P \cdot M_R \quad (7)$$

where  $M$  is then a rotation matrix such that  $M \cdot \mathbf{e}_x = \mathbf{a}$ . We have

$$M_Y = \begin{bmatrix} \cos(Y) & -\sin(Y) & 0 \\ \sin(Y) & \cos(Y) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (8)$$

$$M_P = \begin{bmatrix} \cos(P) & 0 & \sin(P) \\ 0 & 1 & 0 \\ -\sin(P) & 0 & \cos(P) \end{bmatrix} \quad (9)$$

$$M_R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(R) & \sin(R) \\ 0 & \sin(R) & \cos(R) \end{bmatrix} \quad (10)$$

To change  $\mathbf{v}$  into the desired basis, we simply do  $\mathbf{v}' = M^{-1}\mathbf{v}$ . Observe that the rotation matrix only rotates vectors, it cannot scale them. Thus,  $\det(M) = 1$ .

For the second step of the calculation, we note the general equation for an origin-centred ellipsoid with its semi-axes on the basis vectors:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1 \quad (11)$$

where  $(x, y, z)$  is some point on the surface of the ellipsoid with major semi-axis  $a$  and the minor semi-axes  $b$  and  $c$ . We have changed basis into one where our ellipsoid is oriented in this fashion and are looking for the point where  $\mathbf{v}$  intersects the ellipsoid to calculate the distance  $d$  to it from the origin. Let  $\mathbf{w} = \frac{\mathbf{v}}{\|\mathbf{v}\|} = (w_x, w_y, w_z)$ .  $\mathbf{w}$  is a vector pointing in the direction of  $\mathbf{v}$  with length 1, and the sought distance  $d$  is a scalar such that  $d \cdot \mathbf{w}$  is a point on the surface of the ellipsoid found by traveling the direction of  $\mathbf{v}$ , i.e., the intersection of  $\mathbf{v}$  with the ellipsoid surface. Thus,  $d$  must be some multiple of  $\|\mathbf{w}\|$  that satisfies the ellipsoid equation, so that

$$\frac{(d \cdot w_x)^2}{a^2} + \frac{(d \cdot w_y)^2}{b^2} + \frac{(d \cdot w_z)^2}{c^2} = 1 \quad (12)$$

From the last equation the distance  $d$  to the surface of the ellipsoid can be calculated as:

$$d = \frac{1}{\sqrt{\frac{w_x^2}{a^2} + \frac{w_y^2}{b^2} + \frac{w_z^2}{c^2}}} \quad (13)$$

### 3.3. Implementation procedure

A c-style struct is created from user input and saved, comprising:

1. The preferred direction  $\mathbf{a} = (a_x, a_y, a_z)$  for the modeled material as a vector of doubles. This is the vector that points in the direction of the major semi-axis of the ellipsoid used in the calculations.
2. Two numbers  $b$  and  $c$  such that  $\|\mathbf{b}\| = b \cdot \|\mathbf{a}\|$  and  $\|\mathbf{c}\| = c \cdot \|\mathbf{a}\|$ , if the other two semi-axes are thought to point along vectors  $\mathbf{b}$  and  $\mathbf{c}$ , represented as doubles.
3. A number  $r$  representing the intrinsic rotation of the ellipsoid in radians.
4. A double called “weight”, intended to use if extending this modification to allow for more specific handling of several ellipsoids.

This struct is then used to calculate the Euler angles  $Y, P, R$  as described in the section above. These angles are used to construct the rotation matrices  $M_Y, M_P, M_R$ . The total rotation matrix  $M = M_Y \cdot M_P \cdot M_R$  is then constructed. The matrices  $L, U$  such that  $L \cdot U = M$  are calculated and saved as a vector of vectors of doubles in the struct. This struct is saved in a vector of similar structs. All of this is done upon initialization of the simulation.

For every time step in the simulation, the influence function is called with a force vector  $\mathbf{v} = (x, v_y, v_z)$  as input. The vector of structs is iterated over, using each LU factorization to calculate the distance from the center of the ellipsoid it represents to its edge in the direction of  $\mathbf{v}$ . These distances are temporarily saved in a vector of doubles. The largest number in this vector is then found and returned. If no preferred directions are input the vector of structs will be empty, and the influence function will instead return  $\sqrt{v_x^2 + v_y^2 + v_z^2} = \|\mathbf{v}\|$ , indicating an isotropic material.

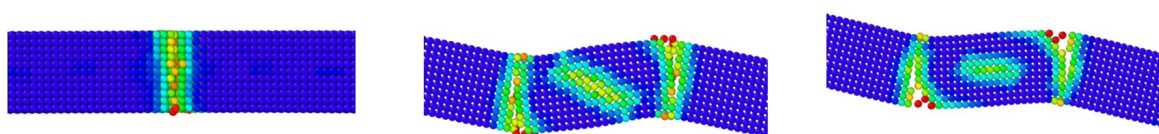
## 4. Results

To explore the performance of the anisotropic influence function being implemented is tested by conducting a compression test on a beam. A beam with a cross-section of  $11 \times 11$  particles and a length of 100 particles was constructed in PDLAMMPS, aligned along the x axis. The length of the beam is 30 cm and the cross section is  $3 \times 3$  cm. The beam was loaded in compression by prescribing the displacement at the two end surfaces. The ends of the beam were set to move toward the center of the beam at 0.2 cm/s. The tests were run for a variable amount of time steps, enough for the beam to buckle and crack. In all simulation the damage parameter was set to 0.1 and the horizon set to 1.0 cm.

As the beam is loaded during the simulation, the bonds between the particles may break when they are stretched beyond a given limit. The damage associated with each particle can be computed in LAMMPS and represented by taking value between 0 and 1. The upper limit correspond to a complete broken bond that can never recover. In the results presented below the damage parameter and the stress in axial direction are compared for different simulation.

In the initial test simulation the major axes and the semi-axes were set to be equal, i.e., the model should return the same result as the original isotropic influence function in PDLAMMPS. The obtained stresses were accurate to six digits of precision when comparing the two models, the anisotropic influence function with the initial isotropic influence function. The next step of testing the internal consistency is to achieve identical results when given different ellipsoids with the same proportions, i.e., doubling the major axis but proportioning the semi-axes in the same way. This is to ensure that the non-spherical influence function modulates the force depending only on its direction.

The initiation of the slip band of peridynamic nanobeam with the original isotropic model and anisotropic model with two preferable directions are seen in Figure 3. With the isotropic model necking band is in the vertical direction, whereas when introducing anisotropic model with preferable direction, this can be controlled in the desirable direction.



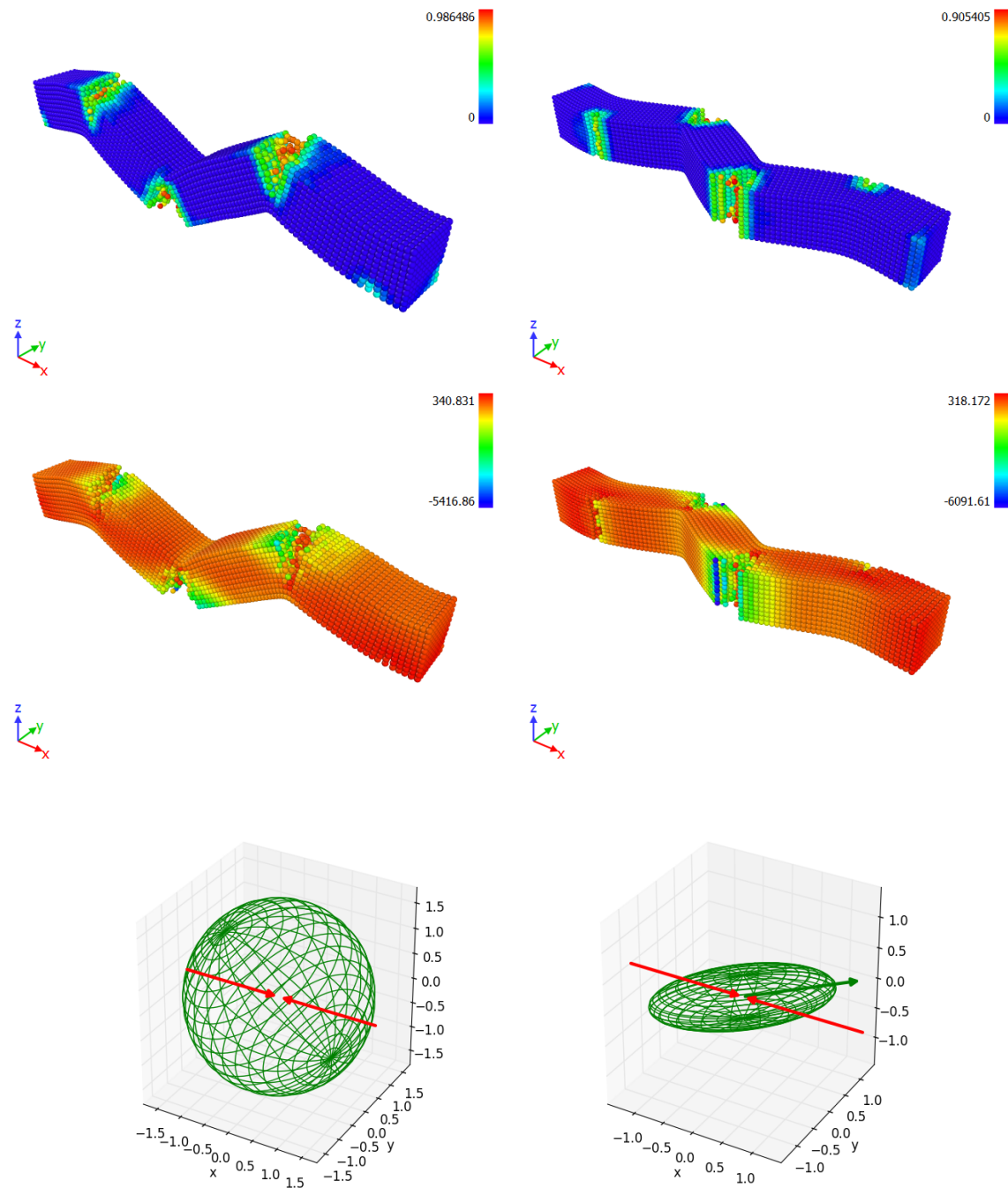
**Figure 3.** Deformation of the nanobeam using peridynamics, original isotropic model (left), anisotropic model with preferable direction oriented 45 (middle) and anisotropic model with preferable direction along the beam (right).

Figures 4–7 consists of a series of tests with ellipsoid influence functions, where the major axes of the ellipsoids point in different directions in the  $x$ – $y$ -plane of the simulation. Figure 4 shows result for the damage parameter and the stress in the loading direction for simulations with spherical influence functions, to use as a baseline to inspect the ellipsoid influence functions against and result with an ellipsoid influence function, where the major axis of the ellipsoid is directed at a  $45^\circ$  angle from the  $x$ -axis. With the isotropic model necking band is in the vertical direction, whereas when introducing anisotropic model with preferable direction, this can be controlled in the desirable direction. The beam acts qualitatively differently when there is an ellipsoid influence function present compared to the spherical case. With all the ellipsoid influence functions the beam bends and cracks in the  $x$ – $y$ -plane rather than in the  $x$ – $z$ -plane, most probably corresponding to the fact that all the ellipsoids are oriented in, and thus give rise to anisotropy in, said  $x$ – $y$ -plane. The approach taken here is replace the sphere with an ellipsoid by setting the distance to the edge of the ellipsoid with the major semi-axis being the length of the incoming force vector and having a predefined direction, the other semi-axes having a lengths some fraction of the major semi-axis. The pre-defined direction of the major semi-axis correspond to the preferred direction of deformation of the solid being modeled. Since the preferable direction in Figure 4 is  $45^\circ$  angle from the  $x$ -axis, the maximum stress in the  $x$ -direction is lower compared to the isotropic model.

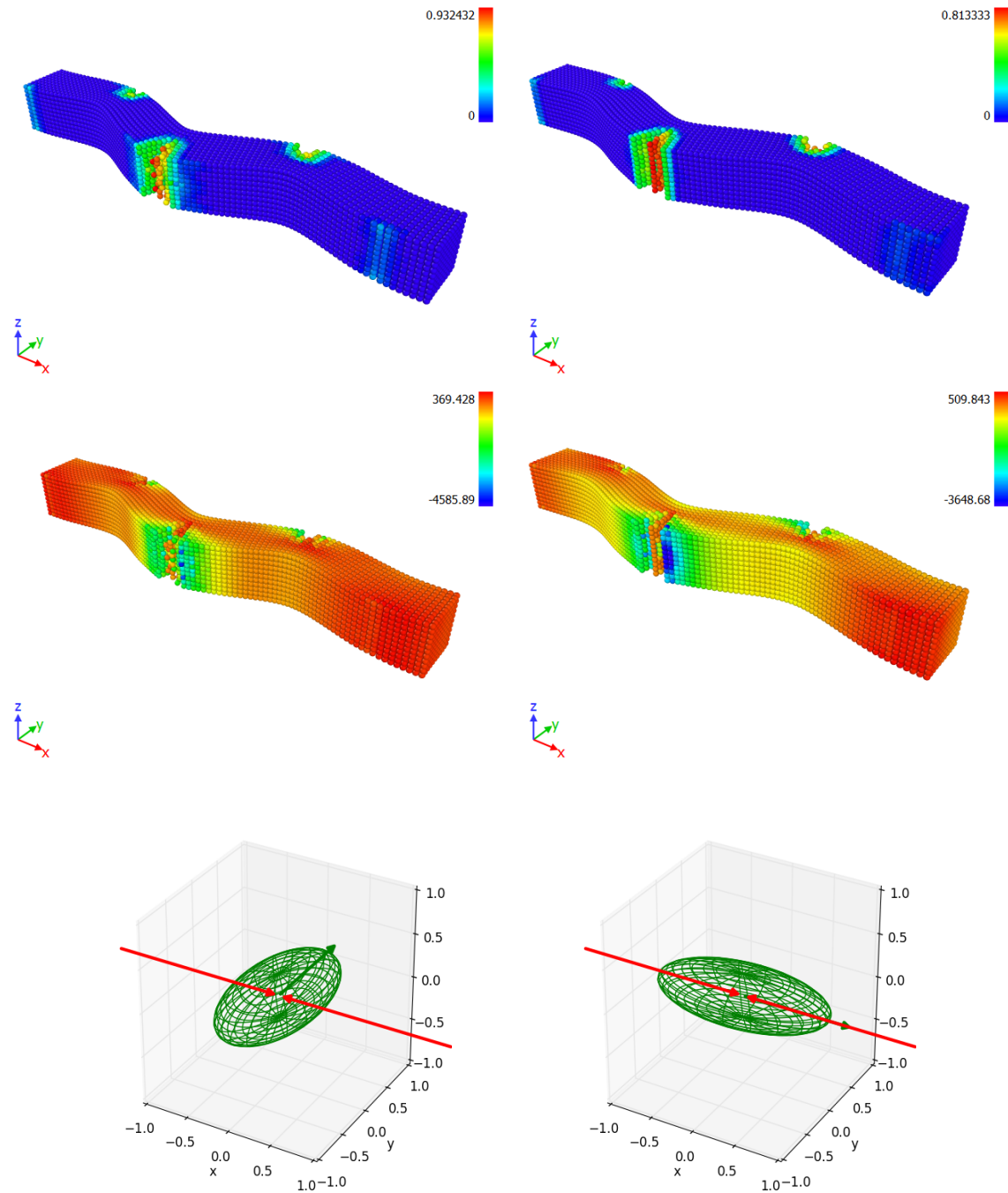
Figure 5 shows results with an ellipsoid influence function, where the major axis of the ellipsoid orthogonal and parallel to the loading direction along  $x$ -axis. The occurring slip bands are orthogonal to the the loading direction leading to much higher damage value for the case when the major axis of the ellipsoid orthogonal to the loading direction along  $x$ -axis. Comparing the axial stress it is clearly seen that higher stress level is reached when the major axis of the ellipsoid is oriented along the loading



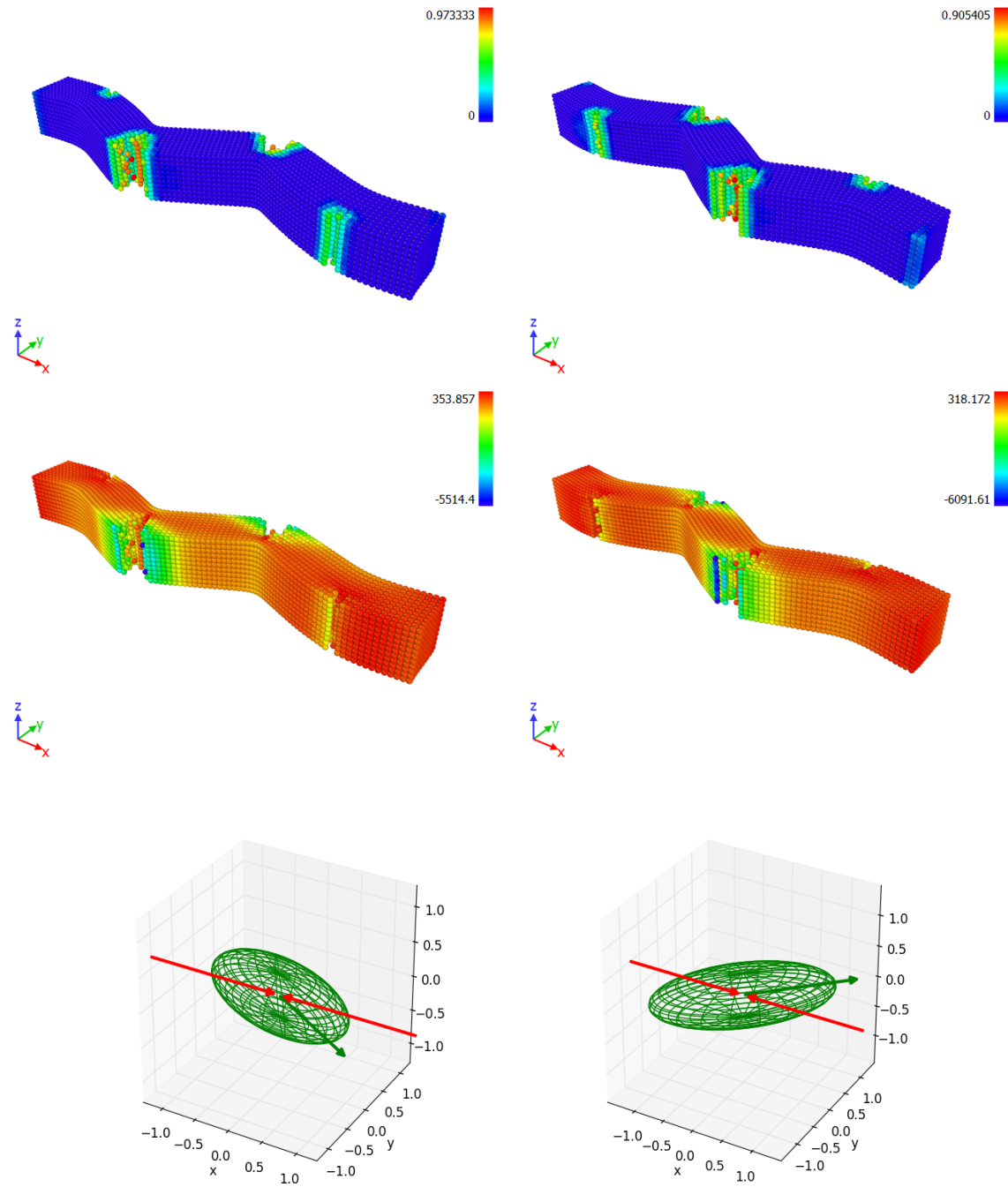
direction. On Figure 6 results with the preferable direction of  $45^\circ$  angle from the x-axis in the x–y plane is compared to simulation with ellipsoid directed at a  $22.5^\circ$  angle from the x-axis in the x–z plane. The axial stress is as expected higher for the ellipsoid directed at a  $22.5^\circ$  angle from the x-axis. The fact that the preferable direction lies in different effects the deformation pattern of the beam.



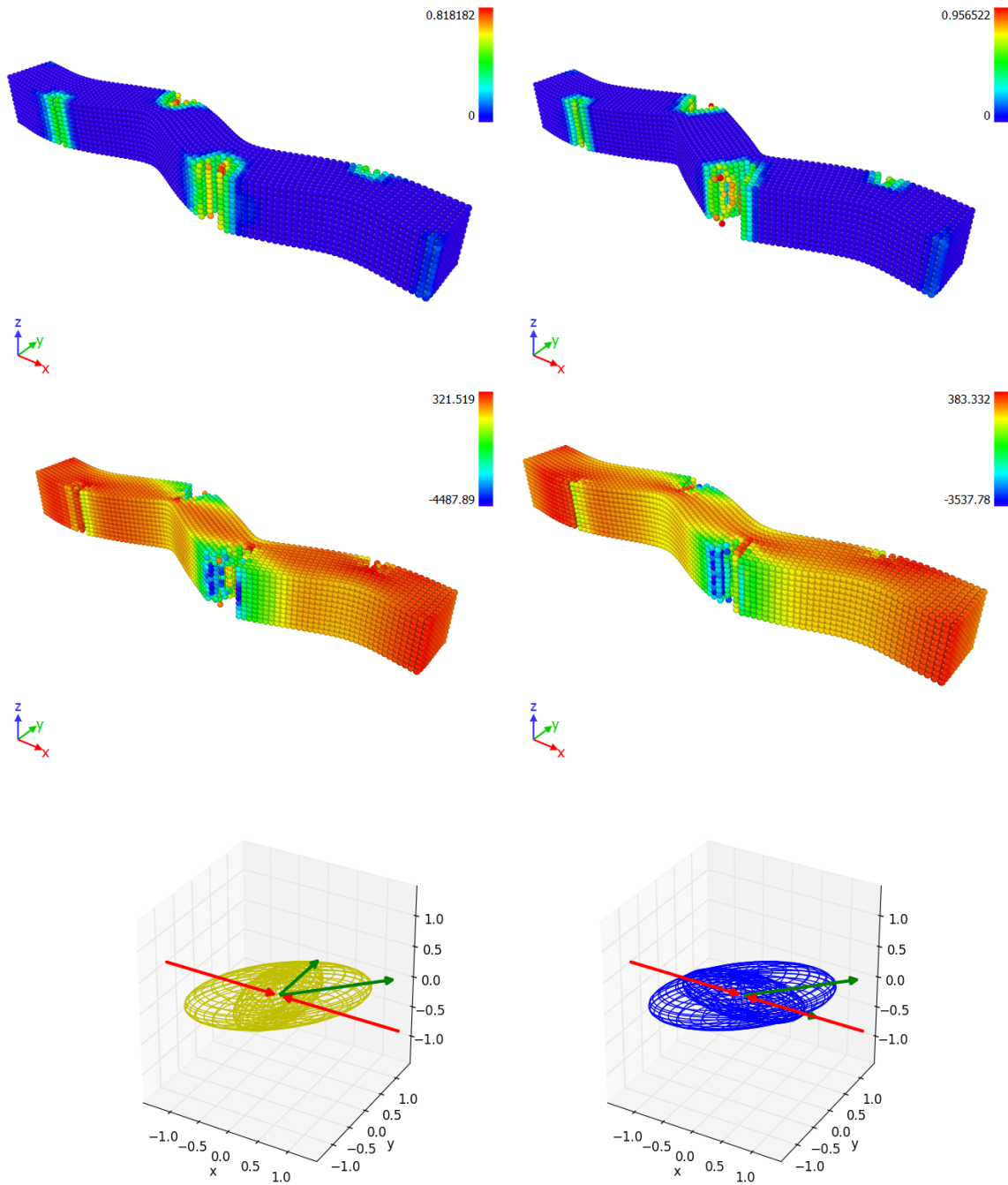
**Figure 4.** Left: Spherical influence function; Right:  $45^\circ$  angle influence function with  $0.5 \times 0.25$  semi-axes.



**Figure 5.** Left:  $45^\circ$  Orthogonal influence function with  $0.5 \times 0.25$  semi-axes; Right: Parallel influence function with  $0.5 \times 0.25$  semi-axes.



**Figure 6.** Left:  $-23.68^\circ$  angle influence function with  $0.5 \times 0.25$  semi-axes; Right: Orthogonal and  $45^\circ$  angle influence functions; Bottom: Parallel and  $45^\circ$  angle influence functions.



**Figure 7.** Left:  $-23.68^\circ$  angle influence function with  $0.5 \times 0.25$  semi-axes; Right: Orthogonal and  $45^\circ$  angle influence functions with  $0.5 \times 0.25$  semi-axes; Bottom: Parallel and  $45^\circ$  angle influence functions.

The implemented model is equipped with probabilities to include several preferable direction. The

model is implemented in a such way that generally any number of ellipsoids oriented arbitrary can be included. Several influence functions oriented in a desirable direction allows this way to model material behavior with preferable direction. Finally a simulation with two concurrent ellipsoids pointing in different directions was conducted. This is illustrated in Figure 7. In the first simulations the model is equipped with two ellipsoidal influence function, one with the preferable direction of  $45^\circ$  angle from the x-axis in the x–y and the second orthogonal to the x-axes. second simulations the model is equipped with one with the preferable direction of  $45^\circ$  angle from the x-axis in the x–y and the second parallel to the x-axes. The orientation of the two ellipsoidal influence function can be seen on Figure 7. The deformation pattern is similar but it difference of the values of the axial stress is clearly seen on Figure 7. The algorithm simulates this by storing several ellipsoids, running the scaling routine on each one and then returning the largest value. This leads to higher axial stress for the case when the second ellipsoid is orthogonal to the loading direction. When putting two ellipsoids in the influence function, a larger ellipsoid dominates a smaller one. This is working as expected—the associated forces modulated by an ellipsoid that envelops another one will always be the largest, and as such will always be selected. However, having only a part of the smaller ellipsoid extend out of the larger one will give rise to measurable differences. With a weighted approach, rather than selecting the largest force always, this could probably be made to be much more convincing.

## 5. Conclusions

An anisotropic peridynamic model has been formulated and implemented in PDLAMMPS. The anisotropy have been introduced by a non-spherical influence function replacing the original spherical influence function. To explore the effects of anisotropic influence function within the peridynamic theory, simulation of a beam loaded on compression have been performed. The suggested non-spherical influence function is not restricted to a specific peridynamic model and can in general be applied to extend any isotropic peridynamic model.

The model shows good consistency when simulations with equal major axes and the semi-axes are performed, i.e., the model return the same result as the original isotropic influence function implemented in PDLAMMPS. The model also shows good internal consistency by reproducing results accurately when given different ellipsoids with the same proportions. Doubling the major axis but proportioning the semi-axes in the same way is accurate to six digits of precision, which is as high as the visualizing software will display.

The beam acts qualitatively and quantitatively differently when there is an ellipsoid influence function present compared to the spherical case. With all the ellipsoid influence functions the beam bends and cracks in the x–y-plane rather than in the x–z-plane, due to the fact that all the ellipsoids are oriented in, and thus give rise to anisotropy in, said x–y-plane. The different orientations of the ellipsoids also give rise to different orientation of the slip bands of the buckling beams. The axial stress is affected in the preferable direction of the ellipsoidal influence function.

Finally we have demonstrated that two or more preferable direction can be introduced in the model and the influence functions can be used to control that. In that way the model can easily be extended to include several preferable direction to model elastic-plastic deformation of anisotropic heterogeneous crystalline matter.

## Conflict of interest

The authors declare no conflict of interest.

## References

1. Silling SA (2000) Reformulation of elasticity theory for discontinuities and long-range. *J Mech Phys Solids* 48: 175–209.
2. Askari E, Bobaru F, Lehoucq RB, et al. (2008) Peridynamics for multiscale materials modeling. *J Phys Conf Ser* 125: 012078.
3. Silling SA, Askari E (2005) Meshfree method based on the peridynamic model of solid mechanics. *Comput Struct* 83: 1526–1535.
4. Silling SA, Epton M, Weckner O, et al. (2007) Peridynamics States and Constitutive Modeling. *J Elasticity* 88: 151–184.
5. Parks ML, Lehoucq RB, Plimpton SJ, et al. (2008) Implementing peridynamics within a molecular dynamics code. *Comput Phys Commun* 179: 777–783.
6. Seleson P, Parks ML, Gunzburger M, et al. (2009) Peridynamics as an upscaling of molecular dynamics. *Multiscale Model Sim* 8: 204–227.
7. LAMMPS, Available from: <http://lammps.sandia.gov>.
8. Parks ML, Seleson P, Plimpton SJ, et al. (2011) Peridynamics with LAMMPS: A User Guide v0.3 Beta. Sandia National Laboratories.
9. Mitchell J (2011) A nonlocal ordinary state-based plasticity model for peridynamics. Sandia National Lab Report.
10. Mitchell J (2011) A non-local, ordinary-state-based viscoelasticity model for peridynamics. Sandia National Lab Report.
11. Rahman R, Foster JT (2014) Implementation of elastic–plastic model in LAMMPS. University of Texas at San Antonio, Sandia National Laboratory.
12. Rahman R, Foster JT (2014) Implementation of linear viscoelasticity model in pdlammps. University of Texas at San Antonio, Sandia National Laboratory.
13. Rahman R, Foster JT, Plimpton SJ (2014) PDLAMMPS—made easy. University of Texas at San Antonio, Sandia National Laboratory.
14. Lehoucq RB, Silling SA (2008) Force flux and the peridynamic stress tensor. *J Mech Phys Solids* 56: 1566–1577.
15. Seleson P, Parks ML (2012) On the role of the influence function in the peridynamic theory. University of Texas at San Antonio, Sandia National Laboratory.
16. Trefethen LN, Bau III D (1997) *Numerical Linear Algebra*.



©2018, the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)