



**HAL**  
open science

# A comparison between the finite element method and a kinematic model derived from robot swarms for first and second gradient continua

Ramiro Dell'erba, P. D'avanzo, A C Rapisarda

## ► To cite this version:

Ramiro Dell'erba, P. D'avanzo, A C Rapisarda. A comparison between the finite element method and a kinematic model derived from robot swarms for first and second gradient continua. *Continuum Mechanics and Thermodynamics*, 2023, 10.1007/s00161-023-01217-9 . hal-04065640

**HAL Id: hal-04065640**

**<https://hal.science/hal-04065640>**

Submitted on 27 Apr 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



Ramiro dell'Erba  · P. D'Avanzo · A. C. Rapisarda

# A comparison between the finite element method and a kinematic model derived from robot swarms for first and second gradient continua

Received: 2 February 2023 / Accepted: 22 March 2023  
© The Author(s) 2023

**Abstract** In this paper, we consider a deformable continuous medium and its discrete representation realized by a lattice of points. The former is solved using the classical variational formulation with the finite element method. The latter, a 2D discrete “kinematic” model, instead is conceived to determine the displacements of the lattice points depending on interaction rules among them and thus provides the final configuration of the system. The kinematic model assigns the displacements of some points, so-called leaders, by solving Newton’s law; the other points, namely followers, are left to rearrange themselves according to the lattice structure and the flocking rules. These rules are derived from the effort to describe the behaviour of a robot swarm as a single whole organism. The advantage of the kinematic model lies in reducing computational cost and the easiness of managing complicated structures and fracture phenomena. In addition, generalizing the discrete model to non-local interactions, such as for second gradient materials, is easier than solving partial differential equations. This paper aims to compare and discuss the deformed configurations obtained by these two approaches. The comparison between FEM and the kinematic model shows a reasonable agreement even in the case of large deformations for the standard case of the first gradient continuum.

**Keywords** Swarm robotics · Discrete mechanical systems · Position based dynamic

## 1 Introduction

Using materials with custom-designed properties has pushed continuum mechanics towards new goals to meet the challenges proposed by increasingly high-performance materials. Increased performance of computer hardware allows simulations of standard and non-standard continua to be performed with relative ease, and many models have been proposed to describe the deformation of new materials. Therefore, new numerical approaches have been developed to obtain better and faster simulations at low computational costs that are parallelizable and can exploit the Compute Unified Device Architecture (CUDA). Thus, alternative systems to Newton’s equation have gained interest to get a reasonable description of the phenomenon. Among these is the concept of Position-Based Dynamic (PBD) [1].

---

Communicated by Andreas Öchsner.

---

Ramiro dell'Erba (✉)  
ENEA Robotics and Artificial Intelligence Laboratory, Rome, Italy  
E-mail: ramiro.dellerba@enea.it

P. D'Avanzo · A.C. Rapisarda  
University of Naples Federico II, Naples, Italy  
E-mail: paolo.dvz@gmail.com

A. C. Rapisarda  
E-mail: alessioxe@gmail.com

The origin of this idea lies in the Computer Graphics industry [2]; in video games, it is necessary to calculate the deformation and breakage of objects very quickly. The body is discretized, and PBD tries to obtain a plausible description of its deformation, considering only the relative positions of the points to describe the action of the internal forces. It is imperative in the visual effects industry to provide movies with simulated versions of complex physical phenomena that are too expensive or impossible to solve precisely. In such a commercial environment, these simulations must be as efficient and fast as possible while remaining realistic to the viewer's eye.

Initially, approaches to simulating dynamic systems in computer graphics were often based on force [3] and Newton's equations. Internal forces (i.e. elasticity) and external forces (i.e. gravity) are computed to obtain the displacements; they are derived from Newton's law using the discrete time integration method to update the velocities, from the accelerations, and finally, the positions of the discretized object.

FEM analysis, but also some numerical alternative as Hencky discretization [4–7] or isogeometric [8–13] are also used.

This classical approach presents many problems which are typical for numerical integration systems. Moreover, collision constraints and penetrations are difficult to manage. Also, while external forces are relatively easy to apply, problems occur in calculating internal forces. Sometimes these forces are represented by a massless linear spring (spring model); this model constitutes an important class of models that can be characterized in different ways by changing the geometry of the springs and their type. Some difficulties may arise when one has to separate bending stiffness from torsional stiffness. It is possible to establish micro–macro relationships to link spring constants with the constitutive parameters of bulk materials (i.e. Young's modulus and Poisson's ratio). Some other models to describe internal forces may be unstable or require high computation time.

Another possibility involves the use of energy minimization methods and the actions principles [14]. These are guided by variational principles, which state that a constrained surface will take the form that minimizes the total strain energy. Unfortunately, differential equations are still needed to calculate the force by expressing the energy of a surface in terms of its local deformation and differentiating the energy with respect to position. The problem is even more difficult in the case of the second gradient, where the energy is written in a more complex way, and Lamé's two constants increase by five. This because in the case of strain gradient theory of a centro-symmetric and isotropic material, there are 5 additional constants (so they are 7 instead of 2 in the simplest case).

Another possibility emerged in the 1970s with smoothed particle hydrodynamics (SPH) [15]. But in this case one always starts with the Navier–Stokes equations to be discretized and thus the computational cost is high.

We emphasize how in computational sciences the main focus is on accuracy. In contrast, the main issues in the Computer Graphics industry are stability, robustness, and computational speed, keeping the results plausible. So far, a new method has been developed in which it is preferable to have direct control over the positions of objects, often represented by the vertices of a mesh, avoiding collision between them and managing their displacements instead of dialoguing with forces. The method is known as Position-Based Dynamic (PBD) and was introduced by Jakobsen in 2001 [16]. It works directly on positions, which makes manipulations and deformations easy to manage. Moreover, with the PBD approach, overshooting and energy problems, related to explicit integration can be avoided because the integration can be directly controlled. The advantage of this technique, even over a mass-spring model, is its stability, even when taking large time steps, which is not always possible in the mass-spring model. The new position can be expressed as an algebraic function of the old position and the time step. The algorithm is iterative and can be stopped at any time if a certain inaccuracy is accepted: If the time step is larger the algorithm can go faster and thus a time/accuracy trade-off must be chosen. Problems with internal forces are avoided by replacing them with a set of geometric constraints. In the simple case, the constraints are holonomic and bilateral. To simulate elastic behaviour, for example, the mass-spring model can be converted to a constraint system, which requires the distance between connected nodes to be equal to the initial distance. This system can be solved sequentially and iteratively, moving nodes to satisfy each constraint until sufficient accuracy is achieved. One of the main limitations may be the dependence of the results on the number of iterations of the solver. Still, one of the significant advantages of a position-based approach is its controllability. Constitutive parameters of the material are hidden in the rules of interaction between first neighbours, hence in the constraints.

In some previous papers [17–20], we have shown how a 2D kinetic model is able to handle different types of phenomena, such as elasticity, plasticity, and fracture, using the same framework. The ultimate goal is not to obtain an exact description of the deformation, but a plausible visible description. This work aims to obtain an algebraic equation that can calculate the position of a point using the placement of other points as input.

This kinematic approach has the advantage of working with simple algebraic equations and the complexity of the system increases only linearly, rather than quadratically, as the number of points increases. We want to emphasize now that, in this paper, we use an approach that is only partially similar to PBD because, so far, we only use position data.

In the context of underwater swarm robotics, the problem related to flocking rules to determine what the behaviour of a single robot should be so that the swarm reaches a certain assigned configuration is often faced. One option to accomplish this task is to consider what a given number of its neighbours are doing. Using this information, a pure kinematic model can be developed with which we try to reproduce some behaviours of two-dimensional deformable bodies according to the standard Cauchy model as well as to the second gradient theory. All the constitutive properties of the material are within the rules determining the kinematic displacements of the particles, each with respect to the others. This tool has an advantage in terms of computational cost and is very flexible to be adapted to samples of complex geometry and different behaviours. The results are encouraging and even fracture can be considered easily and in many different ways.

The other PBD methods we found in the literature require knowledge of the particle velocity, are related to Newton's equation, and use mass explicitly. We avoid, for now, the explicit use of these concepts by using a purely positional kinematic approach. Therefore, in this work, we refrain from considering basic aspects such as mass, velocity, and conservation of momentum but only evaluate kinematic characteristics and transformation rules. The most crucial problem, then, is how to reconnect the classical constitutive equations, which determine the characteristics of the model and the material, to the parameters of the kinematic model under study.

The proposed model is based on swarm intelligence theory [21]. The algorithm that exploits is parallelizable and runs with the CUDA programming language, so we can obtain numerical simulations in a few seconds, even if they are very complex, using a modest desktop PC. Remark that the model does not yet have any physical meaning. It concerns only a set of variables, which are points in a Cartesian plane, that change coordinates in successive steps by iterations of an algorithm. Therefore, physical concepts such as "motion", "time" or "velocity" are not considered at this stage of the model development. It will be the subject of future work to connect the model parameters with the constitutive equations of the material.

This tool could be helpful in complex microstructures not easily analysed by the Cauchy continuum [22]. Classical Cauchy continua cannot accurately predict highly inhomogeneous microstructures [23–27] and [28]. However, generalizations must be introduced by considering additional degrees of freedom to account for kinematics at the microstructure level or by including higher displacement gradients than the first one in the strain energy density [29–36]. The latter is particularly relevant when considering the technological interest in developing exotic mechanical metamaterials capable of performing targeted tasks; therefore, investigating new and efficient algorithms, such as our tool, is of great interest.

In the second gradient theory, the Lagrangian function also depends on the second derivatives of the displacement field and not only on the first ones. The origin of this model lies in the fact that higher-order derivatives have sometimes been included to obtain more general models; Lagrangian functions dependent on higher-order derivatives are often called non-local in the literature [37–50]. This terminology can be explained by considering that, while for a standard material, in a discrete framework with a lattice of points, a simple interaction between a given point and the immediately neighbouring points, in the continuum limit involves an energy function based on the first derivative of the placement field according to the Piola's ansatz. In the case of the  $n$ th gradient, the interaction among points involves all the  $n$ th frame neighbouring points in the lattice. Therefore, in the continuum limit, the energy depends on the  $n$ th derivative of the placement [51–55]. Indeed, in practice, one must measure finite differences of higher order in a discrete framework to estimate the values of the higher derivatives. Furthermore, by introducing new characteristic lengths, higher-order models also introduce new parameters that require appropriate methods for their determination [56–62].

Sometimes remote actions are excluded, which has led to the introduction of local Lagrangians; however, in some situations, non-local Lagrangians must be considered. For example, in composite materials where microfibers are present, these can introduce non-local interaction between points that are not close in space due to the connection between distant points provided by the fibres.

We are aware that many mathematical results are challenging to satisfy, but we believe that the proposed discrete algorithm can be helpful in a wide variety of mechanical applications.

The paper is organized as follows. The first section contains a description of the swarm robot model. The second section shows numerical results comparing a 2D continuum with a first and second gradient energy and the proposed kinematic model. Future developments on this topic and open questions are presented in the conclusions.

## 2 The tool

The use of robot swarms in underwater work is becoming increasingly important; flocking rules have been applied to determine the displacement of the individual robot in order to reach the assigned final configuration of the entire swarm [63]. The new position of the robot is determined by the position of its neighbours; we applied the same algorithm to describe the deformation of a lattice of material particles. This is a purely kinematic approach to the problem, without the use of the concepts of mass or force.

In practice, we proposed to consider a transformation operator, with constraints, between matrices representing the particle configuration,  $C_t$ , for a discrete set of time steps  $t_1, t_2, \dots, t_n$ . This operator can be represented by a fourth-order tensor that transforms a two-dimensional matrix into another two-dimensional matrix. The continuous two-dimensional body is discretized into a finite number of particles lying, in the undeformed configuration, in the lattice node. A complete description of the algorithm can be found in [17, 18, 20] where we obtained a plausible description of success even in the case of material fracture. In the previous articles, we have defined four types of particles that make up the lattice but, if necessary, new types can be introduced, to describe other properties, due to the modular structure of the algorithm. They were the leaders whose movement is assigned and determines the movement of the other particles. The followers whose movement is determined by the flocking rule with other particles; we also had the frame, introduced to avoid edge effects and the fourth type concerns fracture. These last two will not be used in this paper, where a model without the frame avoiding edge effect with different flocking rules is considered.

Some hints from graph theory help us to note that time and velocity are concepts that do not belong to this model; therefore, we prefer to talk about pseudo-time as an update step of the algorithm.

We then consider a graph as a mathematical structure used to model pairwise relationships between objects. We denote a graph  $G = (V, L)$  as an ordered pair of a set  $P$  of points, a set  $L$  of links. We will consider only undirected graphs, so there is no direction associated with the links. Considering  $n$  points  $V_i$ , with  $i \in \{1, 2, \dots, n\}$ . We define the adjacency matrix as:

$$A_{i,j} = 1 \text{ if } V_i \text{ is linked to } V_j; 0 \text{ if not} \quad (1)$$

the matrix  $A$  has dimension  $n \times n$ .

This is not the only possible definition of adjacency matrix; to satisfy some properties of our swarm we can change it, i.e. change the definition of neighbour.

A path on the graph is defined as a finite or infinite sequence of links joining a sequence of points; it will be useful to define the concept of neighbourhood order, related to gradient and non-local order.

To describe the time evolution of our swarm we consider a set of  $p$ -time (pseudo-time) steps, i.e. a set of ordered discrete values of the time variable  $t$  that we will denote with  $T = \{t_1, t_2, \dots, t_n\}$ . We talk about pseudo-time because  $t$  is only an index parameter of the loop. A swarm, denoted as  $S$ , is composed by  $n$  material points  $P_i$  ( $i$  from 1 to  $n$ ). At pseudo-time  $t_m$  the  $i$  element of the swarm has position  $P_i(t_m) = (x_i, y_i)(t_m)$  where  $x, y$ , are the coordinates in a reference system. So far  $P_i$  is the material point and  $P_i(t_m)$  is its position represented in Cartesian coordinate. At time  $t_0$ , we denote by  $C_0$  as the reference configuration of  $S$ . Therefore,  $S$  is the swarm as collection of points  $P_i$  while the configurations  $C_m$  are the set of coordinates of them at pseudo-time  $t_m$ . We assume that  $C_0$  is the representation of a bidimensional crystalline lattice to be specified. For each element  $i$  in  $S$  we define a fixed set of neighbour's points, calculated in the reference configuration  $C_0$ . Their number, typically, is  $nc$ , the coordination number of the lattice. We call this set of point the first neighbours. The process is iterative so we can define the neighbours of each neighbours and call this set the second neighbours.

The first neighbours of a point  $P_i$  so are all the points of  $S$  at time  $t_0$  adjacent to the point  $P_i$  in the meaning of the graphs (so it is related to the lattice coordination number). The second neighbours of a point  $P_i$  are all the points of  $S$  whose minimum path without repetitions consists of only two links. Therefore, defining  $l(i, j)$  as the minimum path without repetitions between points  $i$  and  $j$ , the  $k$ -th neighbours of a point  $i$  is the set

$$Ne_k(i) = \{j \in C_0 : l(i, j) = k\} \quad (2)$$

So,  $Ne_k(i)$  is the set of point neighbours of  $i$  order  $k$ . We call it the shell of order  $k$ .

We can also consider pairs of opposite neighbours of chosen point; this will be useful in defining the rule to compute an alignment term. The set of pairs  $k$ -order opposite neighbours of a point  $P_i$  is the set

$$S_k(i) = \{(h, g) \in Ne_k(i) \times Ne_k(i) : (P_h - P_i) = -(P_g - P_i)\} \quad (3)$$

Therefore, we have defined, assigned a point  $P_i$  and its  $k$  neighbours a couple of point composed by the  $k$  neighbours and its opposite neighbours  $h$ . So, the  $j$ -th pair of the point  $P_i$  will be indicated with  $\rho_{ij}$  and its average point of the pair as  $\langle \rho_{ij} \rangle$  for each  $t_m$ .

Note that  $Ne_k(i)$  is the  $i$ -th row of the adjacency matrix: this means that our adjacency matrix  $A$  is not  $n \times n$  but  $n \times m$  where  $m$  is the number of the total neighbours considered until the  $k$ -th order.

We select a set of leaders  $L$ ; in analogy to biological swarms they are a subset,  $L$ , of  $C_0$  whose trajectory is determined. These elements drive the evolution and the deformation of the swarm and they have a ‘‘priori’’ imposed motion and are not influenced by other elements in the swarm.

All the remaining elements of  $C_0$  form the set of the Followers,  $F$ . The movement of the follower is guided by the flocking rules. A possible choice of the rules is:

$$\mathbf{r}_i(t_{m+1}) = \mathbf{r}_i(t_m) + \mathbf{Int}_i(t_m) \quad (4)$$

Where  $\mathbf{Int}_i(t_m)$  is an interaction term between the point  $i$  and its neighbours; one of the possibilities is to choose the interaction term so

$$\mathbf{Int}_i(t_{m+1}) = \sum_{h=1}^k \sum_{j \in Ne_h(i)} \mathbf{W}_{i,j}(t_m) + \sum_{h=1}^k \sum_{l \in S_h(i)} \mathbf{Y}_{i,l}(t_m) \quad (5)$$

Where  $k$  is the number of neighbours of the point  $i$ .

$$\mathbf{W}_{i,j}(t_m) = \frac{\alpha_j}{N_i} \left( \|\mathbf{r}_j(i, t_m) - \mathbf{r}_j(i, t_0)\| \right) \frac{\mathbf{r}_i(t_m) - \mathbf{r}_j(i, t_m)}{\|\mathbf{r}_i(t_m) - \mathbf{r}_j(i, t_m)\|} \quad (6)$$

$$\mathbf{Y}_{i,l}(t_m) = \frac{\beta_l}{N_i} (\mathbf{r}_i(t_m) - \langle \rho_{i,l}(t_m) \rangle) \quad (7)$$

Where the quantities are explained as follow.

$\mathbf{r}_i(t_{m+1})$  is the new position of the point  $P_i$ .  $\mathbf{Int}_i(t_m)$  is the Interaction term among the point  $P_i$  and all its neighbours in the  $h$ -th order shell until the  $k$ -th order.  $N_i$  is the number of the neighbours of the point  $P_i$ .  $\mathbf{r}_{j,i}(t_m)$  is the new position of the  $j$ -th neighbours of the point  $P_i$  at the  $p$ -time step  $t_m$ .  $\alpha_j$  is the coefficient that modulates the intensity of the ‘‘elastic’’ interaction ( $\mathbf{W}_{i,j}$ ) and the convergence velocity. It depends by the  $j$ -th neighbour’s shell order.  $\beta_l$  is the coefficient that modulates the intensity of the ‘‘alignment’’ interaction ( $\mathbf{Y}_{i,l}$ ). It depends by the  $j$ -th neighbour’s shell order; the  $l$  is the label of the couple  $\rho_{i,l}$ .  $\langle \cdot \rangle$  has the meaning of simple average value of the distances,

The term  $\mathbf{W}_{i,j}$  can simulate an elastic behaviour of the material and the rule contains a dependence on the distances between the point and its neighbours trying to emulate a similarity with Hooke’s law. We remark that the proposed algorithm is still different from a particle system with spring interactions and some constrained, because we do not give a constitutive equation for elastic potential. We implement a recursive set of rules in order to define the evolution. It should be noted how in Eqs. 5 and 6 although the interaction is between two particles it occurs for all neighbours, so this would be a ‘multi-body’ Hooke’s law.

The  $\mathbf{Y}_{i,l}$  term was introduced to handle two problems. The first is to obtain a way to modulate the bending response of the swarm. Indeed, the  $\mathbf{Y}_{i,l}$  term pushes the point  $i$  to be located in the centroid of the considered pair of opposite neighbours. It is an alignment term but we will not always consider the  $\mathbf{Y}_{i,l}$  term in this paper using other methodologies. The second problem is to avoid the overlap of points during displacement.

In previous articles we have supposed the existence of a frame, but not here. This is because, unlike [17, 18], to balance the interaction on the edges of the swarm we do not need all shells to be complete (i.e. each shell has the maximum number of neighbours expected for the considered order). When used, Frame points have their own rule of motion and are moved in a different pseudo-time than the other points in the swarm. In their absence therefore, here all non-Leader points are moved simultaneously and this makes the algorithm leaner, smoother and faster. However, we noticed that the edges of the swarm suffer from local overlaps and oscillations; they cause exotic behaviours that propagate within the whole lattice, especially increasing the term  $\alpha_j$ . Thus, our idea about the incompleteness of neighbouring shells was wrong, and to correct it, we correct the term  $1/N_i$  with  $1/(2Z_k - N_k(i))$ , in Eq. 6, where  $Z_k$  is the number of points in the shell of order  $k$  when it is complete.

The boundary elements may have a larger displacement than the inner followers, due to the lack of points at the edge of the swarm not balancing the interaction. This because sometimes a border point can have a neighbour’s number less than  $k$ . This reduces the weight of the edge points and coregulates the imbalance. This modified rule will be used thereafter.

Several methods have been used to avoid particle overlap and to overcome the alignment parameter  $Y_{i,1}$ ; one of them is the use of intermediate propagation stages where the leaders are frozen and the lattice resettles.

In previous work [19] it has been observed how the deformation information in the swarm propagates through its elements as the pseudo-temporal steps move forward. The strain information will propagate through the swarm proceeding in shells that are proportional to the  $k$ -order of the neighbours considered in the interaction.

To avoid point's overlap, we introduce the parameter  $\gamma$ ; It consists in a certain number of settling steps while the leaders are stopped. In the previous model it was necessary to pay attention to the pseudo-velocity of the leaders, assign them an extremely low value (resulting in longer computation times) to avoid the overlap problem (especially in compression simulations) and give enough time to the followers to adapt according to the deformations. We introduced algorithm cycles  $\gamma$  between two pseudo-time steps; in these intermediate cycles the leaders are stopped while the followers continue to resettle. This has the significance of changing the information about the rate of propagation of deformation within the swarm. What changes in the visible motion is the stiffness of the deformation but the final configuration obtained is the same.

With the parameter  $\gamma$  we can modulate how many shells are involved in a pseudo-temporal step; it is then related to the rate of propagation of the movie deformation in the swarm. This concept is analogous to the concept of correlation length in lattice models (such as the Ising model) and the concept of persistence length in polymer models.

It needs to find an optimum between the pseudo-speed (which can be higher) and the increase in machine time due to the increase in computational steps.

Practically, we start with a swarm occupying the nodes of a crystal lattice, move the leaders to an assigned position, and calculate the positions of the followers by means of the above equations.

The code was initially developed in Mathematica by Wolfram research. It was later converted into the Python language to take advantage of the CUDA libraries to parallelize the calculation. The simulation involves particles (or agents) moving in two-dimensional space and interacting with each other according to certain rules or behaviour. The code imports several libraries, including matplotlib, NumPy, math, itertools, SciPy, timeit, numba and tqdm. Another section of the code includes several functions for creating the swarm. These functions include ADJMatrix, which creates an unordered adjacency matrix and is called by the lattice function, and NBORD, which reorders the elements of the adjacency matrix and is also called by lattice. The purpose of these functions is to create a network of connections between the agents in the swarm, allowing them to interact with each other. This is done by calculating the distance between each agent and its neighbours and creating a matrix representing these connections. The remaining code includes functions to simulate the behaviour of the agents in the swarm, updating their positions based on interactions with each other and the environment. The code can also include functions to visualize the swarm as it evolves over time using CUDA. It takes several input parameters, such as step size, start and stop times, number of particles, their initial positions and velocities, and various interaction parameters. The function then prepares the data to invoke a CUDA kernel, which is responsible for the calculation of the swarm's evolution. The kernel uses the input parameters to calculate the new positions and velocities of the particles, based on their interactions with neighbouring particles and the swarm leader. The function then returns the updated positions and velocities of the particles, as well as the time step at which the function finished its calculation. The kernel is structured in such a way that it performs the particle interactions in parallel on several GPU threads.

### 3 Results

In this section, we present some comparisons between our tool and the solution obtained by COMSOL software using finite elements analysis on a square lattice.

Let us consider a swarm  $S$  constituted by a finite number of elements, and indicate by  $C_0$  the reference configuration of  $S$  a time  $t = t_0$ . In  $C_0$  the elements of  $S$  occupy the nodes of the lattice. We consider a set of time steps, i.e. a set of ordered discrete values for the time variable  $t$  that we will denote with  $T_m = \{t_1, t_2, t_m\}$ . Consider an orthonormal reference system with axes that are parallel to the lattice, whose unit length is equal to the side of the lattice and whose origin coincides with the left bottom vertex of the swarm. To identify the elements of the swarm, we use the coordinate  $(x_{ij}, y_{ij})(t_m)$  of the element occupying the node  $(i, j)$  at the time  $t_m$ . For each element  $(i, j)$ , we associate a set of neighbours  $k$ th order as the  $-$ th coordination number of the lattice cell. By this, we define, in case of  $k = 1$  and  $k = 2$ , the definition of first and second neighbours, including the neighbours of the first neighbours. Actually, we use a definition of  $k$ -th neighbours

that is Lagrangian, i.e. the definition has memory of the identity of the neighbours of a given element of  $S$ . The reason is that we are interested in describing matter in solid state, where the behaviour of constitutive particles depends on local molecular interactions which are preserved during the evolution of the system. In the next papers, we shall relax this hypothesis.

So, we select a leader (or more than one) that is an element  $L$  placed at a node  $\{i, j\}$  of the lattice with an assigned displacement  $M$ , i.e.

$$M : t_m \in T_m \rightarrow (x_{ij} y_{ij})(t_m)$$

That means a selection of points of  $S$  whose initial coordinates are a subset of  $C_0$ .

In particular, the deformations we have compared are related to a rectangular specimen of size  $1.2\text{ m} \times 0.3\text{ m}$  that undergoes simple tensile stress with and without a slot (dimensions of the slot are  $0.9\text{ m} \times 0.075\text{ m}$ ), posed in the centre of the sample. Points on the left side are clamped while the points on the right side of the sample are moved up to the final position which is assigned. Each time unit is composed by two steps. The movement of the leaders and the movement of the followers. No fracture is considered in this case. The simple rule, governing follower's motion is expressed by Eq. 4. This rule can be defined as "barycentre of its neighbours" in case of complete shell that means when all neighbours are considered. The neighbours are determined by the coordination number of the lattice; therefore, the leader's motion implies a displacement of the first layer that propagates in successive "time steps" to the other particles, according to the  $\gamma$  frames used. This means that the displacements, at each time step, involve a larger shell of points until it regards all the lattice points. The quotation marks are to remind that in our model the time is only fictitious because it is related to the iterative process. The model solved with COMSOL treats stationary case; therefore, the time is not important at all.

This led us to two problems: how to link our model's parameters ( $\alpha_i, \beta_i, \gamma$ ) with the constitutive parameters (e.g.  $\nu, E$ ) and how to simulate higher-order gradients continua.

In this section, we will show how we managed an empirical link between the Poisson coefficient  $\nu$  with the only parameter  $\alpha_i$  in order to reproduce with tiny errors a first-order gradient continuum; meanwhile, for second-order gradient continua also  $\beta_i$  plays an important role.

After that, we will show a quantitative comparison between our simulations and FEM solutions, an important step forward with respect to the previous papers [64] and [18].

We want to underline that in this paper we are not considering the fracture. Moreover, the dynamic evolution of the swarm is only a mean to obtain the final equilibrium configuration; in the future we shall discuss as to link this evolution with a real motion. Remember as the value of  $\gamma$  does not influence the final configuration as shown in [18], but only the intermediate steps.

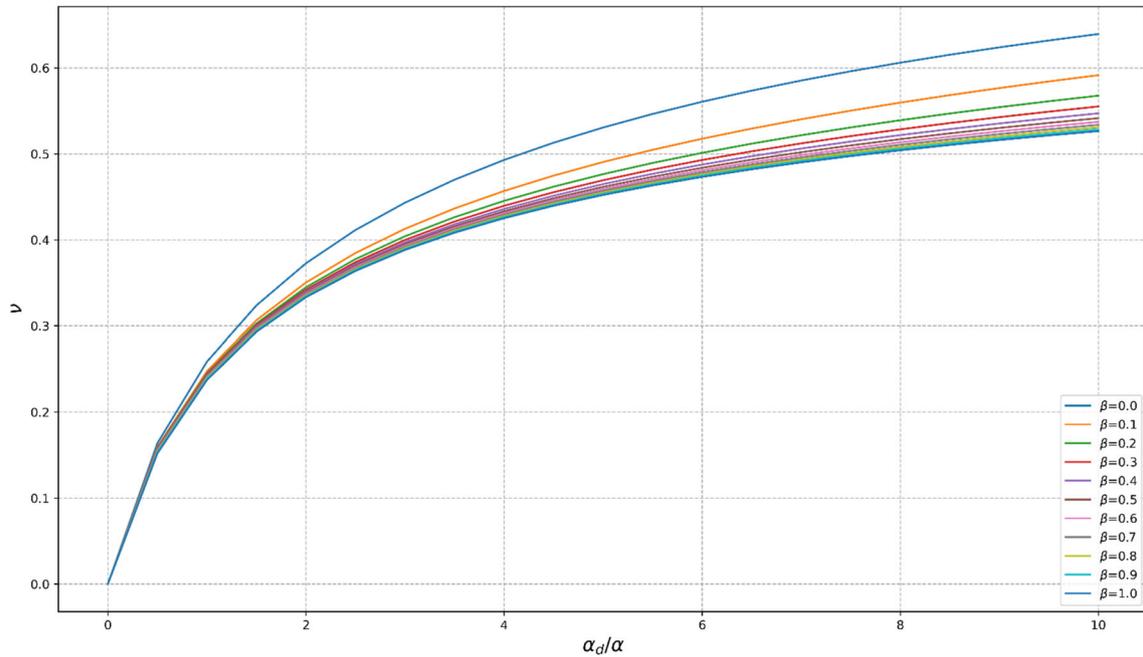
### 3.1 First order gradient

In the previous papers [64] and [18], we stated that the interaction term  $\mathbf{W}_{i,j}$  is sufficient to describe qualitatively a first-order gradient deformation. In this subsection, we want to validate this statement showing that our model can describe first-order gradient deformation also quantitatively, comparing our results with those obtained by FEM analysis.

In [64], we noted that using all the  $\alpha_i$  with the same value  $\alpha$  (Eq. 1) the Poisson effect does not exist in order to obtain Poisson's effect we can assign different values of  $\alpha$  for each neighbour in all the shells of the swarm; by this way, we can vary the Poisson effect. In particular, in the case of a square lattice, we assign a value  $\alpha$  for neighbours positioned in vertical and horizontal directions with respect to the centre point of the shell, and a different value,  $\alpha_d$ , for neighbours positioned in the diagonal direction.

Then we analysed the Poisson effect varying  $\alpha_d$  and keeping  $\alpha$  fixed. We observed that for  $\alpha_d = 0$  there is no Poisson effect; indeed, the displacement of each particle depends only on the interaction with vertical and horizontal neighbours. Therefore, the vertical and horizontal displacement are uncoupled. Increasing  $\alpha_d$ , keeping  $\alpha$  fixed, the Poisson effect become more and more significant. Furthermore, we noted that varying both  $\alpha_d$  and  $\alpha$  keeping their ratio fixed does not alter the Poisson effect (for values of the parameters in standard range as specified in [64]).

Thus, we hypothesize that only the ratio ( $\alpha_d/\alpha$ ) affects the intensity of the Poisson effect. Particularly, we expect that there is an increasing monotone function  $\nu = \nu(\alpha_d/\alpha)$  because we are increasing the horizontal and vertical links.



**Fig. 1** Poisson coefficient versus ratio  $\alpha/\alpha_d$  (Different  $\beta$ )

To test this hypothesis, we perform several simulations varying the ratio ( $\alpha_d/\alpha$ ) and evaluating “geometrically” the Poisson coefficient  $\nu$  for each simulation. This means that we compute as Poisson coefficient the ratio of the transverse strain to the longitudinal strain of the specimen; the transverse strain is considered in the middle of the beam. We obtained the relation  $\nu(\alpha_d/\alpha)$  interpolating the collected data, see Fig. 1 (Consider the value of  $\beta = 0$ . Inverting this relation, we can evaluate the ratio ( $\alpha_d/\alpha$ ) that reproduces the desired Poisson coefficient. Remember as this relationship is purely empirical and there is no theoretical link between our parameter and Poisson coefficient. Forcing the coupling between the ratio ( $\alpha_d/\alpha$ ) to higher values, we are introducing a sort of anisotropy in our model showing Poisson’s ratio larger than 0.5.

We underline that this analysis was performed only for swarms without the slot. This is because we try to simulate the same material with different shapes.

### 3.2 Second-order gradient

In previous papers [18,64], we supposed that second-order gradient phenomena could be obtained extending the  $\mathbf{W}_{i,j}$  interaction to second neighbours. We have observed that the  $\mathbf{Y}_{i,1}$  interaction is just enough to give good results, and we believe that it is a better candidate to describe second-order gradient phenomena with respect to the use of the second neighbours. Therefore, in the following we will consider only the  $\mathbf{W}_{i,j}$  and  $\mathbf{Y}_{i,1}$  interactions extended to the first neighbours in order to simulate the second gradient continua. No second neighbours will be considered in this paper. Since  $\beta$  is linked to three alignment points it is able to introduce non-local interaction albeit in a simple way. For these reasons, we explore the effect of  $\beta$  in reproducing second gradient phenomena.

As in the first gradient case, we performed several simulations varying the parameters ( $\alpha, \alpha_d, \beta$ ) considering the same  $\beta$  for all the couples of first neighbours. For each simulation we valued geometrically the Poisson coefficient. In Fig. 1 is shown how  $\beta$  influences  $\nu$ : fixed the ratio ( $\alpha_d/\alpha$ ), the Poisson coefficient decreases if  $\beta$  increases. A possible explanation could be that increasing  $\beta$  the system becomes stiffer, so Poisson effect is decreasing.

In this case, the relation  $\nu = \nu(\alpha_d/\alpha, \beta)$  is not trivial: different triplets ( $\alpha, \alpha_d, \beta$ ) lead to the same Poisson coefficient but with different global deformations of the swarm. So, being unable to select the right triplets for now, we proceeded in heuristic way to find the triplets that better fit COMSOL simulations.

Also, in this case the magnitude of the Poisson effect does not depend directly on  $\alpha$ , but the ratio ( $\alpha_d/\alpha$ ) affects the intensity of the Poisson effect, as shown in Fig. 1 for different  $\beta$ .

**Table 1** Test parameters

Point's number	2425
Point's number (with slot)	2070
Point's number subset	297
Point's number subset (with slot)	274
Beam dimensions (FEM)	1.2 m $\times$ 0.3 m
Slot dimensions (FEM)	0.9 m $\times$ 0.075 m
Gamma	100
Final strain	25%

**Table 2** Poisson coefficient; Nominal, as measured by FEM and our tool

Sample	$\nu_N$	$\nu_S$	$\nu_M$	$\alpha$	$\alpha_d/\alpha$	Max error	Mean error
First gradient	0.2	0.1470	0.1473	0.5	0.433	0.00060	0.00017
	0.3	0.2630	0.2627	0.5	1.028	0.00106	0.00036
	0.4	0.4428	0.4428	0.5	2.885	0.00343	0.00120
First gradient with slot	0.2	NO	NO	0.5	0.433	0.00139	0.00100
	0.3			0.5	1.028	0.00148	0.00041
	0.4			0.5	2.885	0.00314	0.00230
Second gradient	0.2	0.1245	0.1225	0.005	1	0.00109	0.00054
	0.3	0.2317	0.2352	0.005	2.5	0.00253	0.00144
	0.4	0.4188	0.4104	0.005	8	0.00648	0.00419
Second gradient with slot	0.2	NO	NO	0.005	1	0.00504	0.00229
	0.3			0.005	2.5	0.00734	0.00342
	0.4			0.005	8	0.00839	0.00581

### 3.3 Comparison

To obtain a quantitative measure of the agreement between the results we consider the coordinates of a subset of points,  $N_g$ , in the deformed configuration and we measure the distances between the corresponding points using the two methods, FEM and our tool. A colour plot with the differences shown as a very good agreement can be reached. Three cases are considered with three different nominal Poisson coefficients (0.2, 0.3 and 0.4). The swarm is composed of about 2500 elements; the pictures show a sub sampling of about 300 elements. The value of  $\beta$  is 0 in the case of the first gradient, while 1.3 in the second gradient model; we found that this value has the better match with the FEM solutions. The value of  $\alpha/\alpha_d$  is given in Table 2.

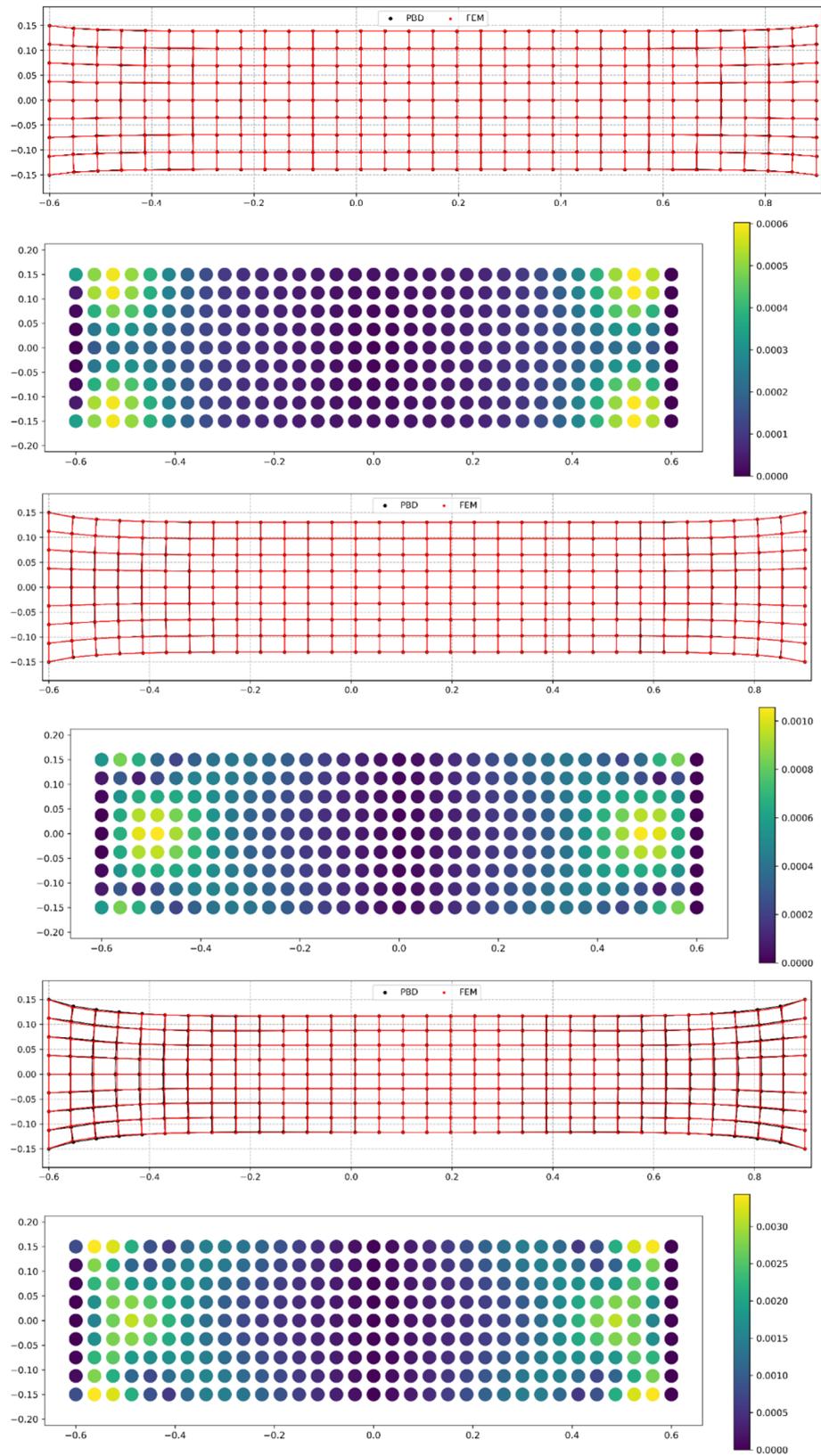
The motion is characterized by the features presented in Table 1.

We then modulated the parameters of our model so that the results coincided as closely as possible with those obtained by the FEM. We have considered two kinds of sample (with and without slot), three Poisson coefficients and first and second gradient cases. Finding the match was not easy, especially in the case of the second gradient, where the parameters involved are more than two owing to the presence of  $\beta$ . Having obtained very similar results, we can make the following considerations.

The differences between our tool solution and FEM are very small. In Table 2, there are max and mean errors, computed between the coordinates of the lattice, for all the cases studied.

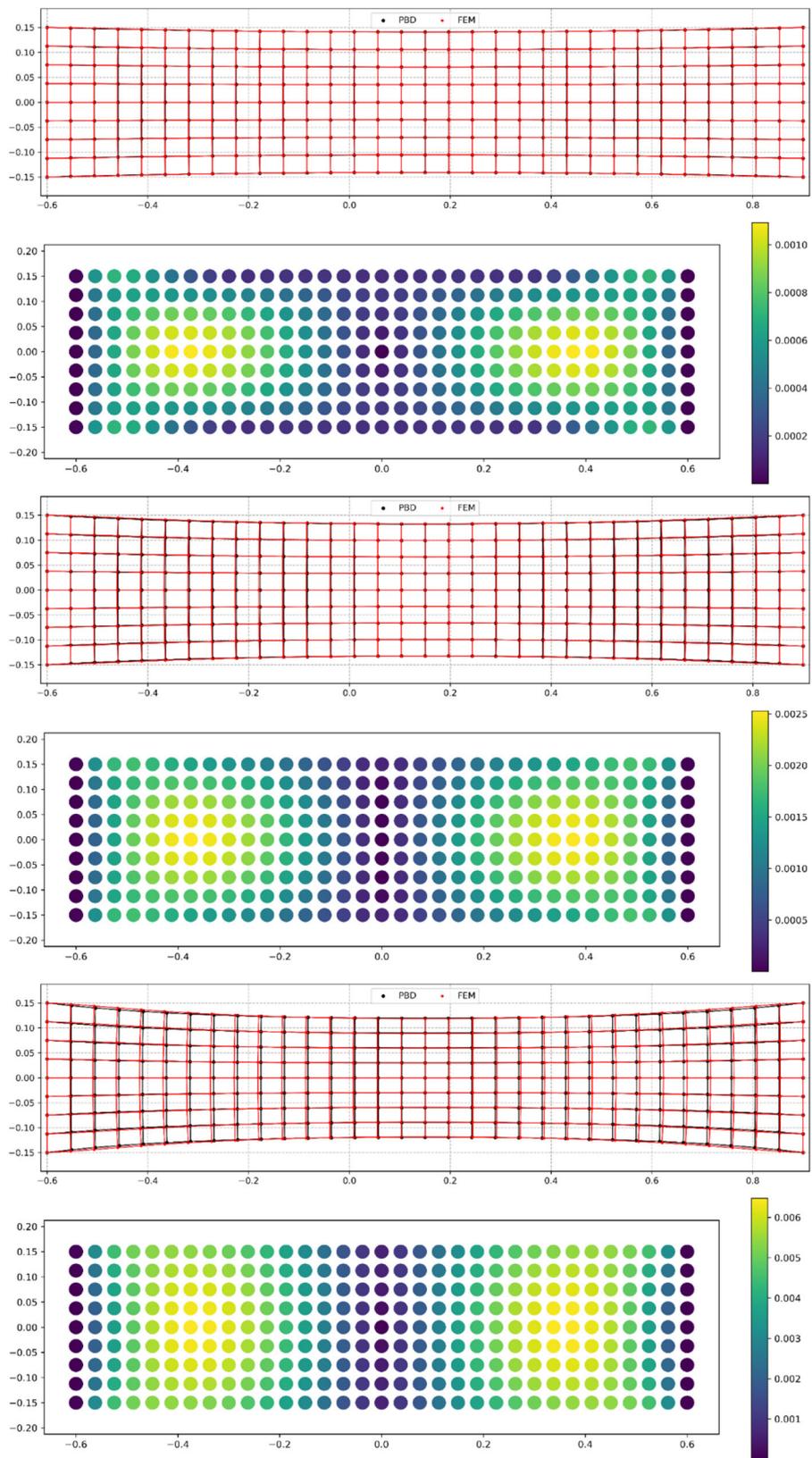
Now we will see where the differences are located within the sample. In the case of the first gradient, without slot, these are located in the lateral area (coloured yellow, see Fig. 2). In the centre of the sample, the differences are almost zero, while near the leaders, where the deformations are greater, the area is more critical. However, the location of the differences is not quite the same; increasing Poisson coefficient we have different areas where the mismatch is greater, also if they always are distributed close to the corners as can be seen from Fig. 2. Far from the leaders, where are concentrated deformation owing to edge effect and at the centre of the specimen, we note a uniform deformation, as it is expected in this kind of tests. In contrast, in the case of the second gradient, the main differences are always located in the same position, i.e. about one-quarter and three-quarters of the sample, and in the centre with respect to the vertical axis. See Fig. 3. This area corresponds to the location of maximum second gradient deformation. Also, in the case of the sample with the slot there are differences. It would seem that for the second gradient the difference between our method and the FEM solution is more pronounced close to the slot towards its centre; the slot is a critical area.

Generally, the agreement between models is greater in the first gradient case. But there is also a qualitative fact. Take for example the case of the sample with the slot in the second gradient condition. For high values of

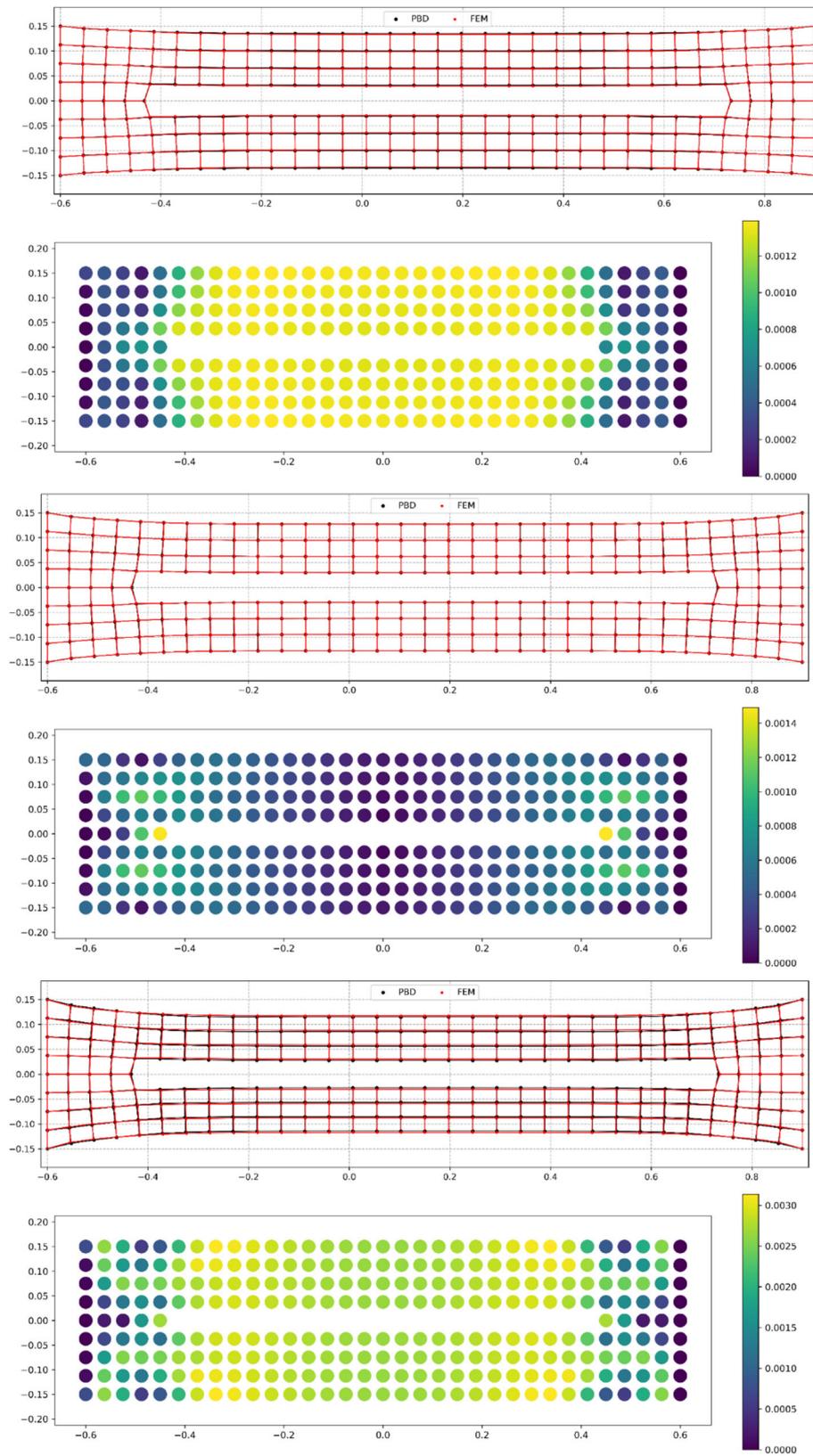


**Fig. 2** Final configuration of the sample for the three Poisson coefficients (0.2, 0.3 and 0.4). Red lattice is the FEM solution, while the black is by our tool. Differences between them are shown in coloured plots. First gradient case (colour figure online)

### A comparison between the finite element

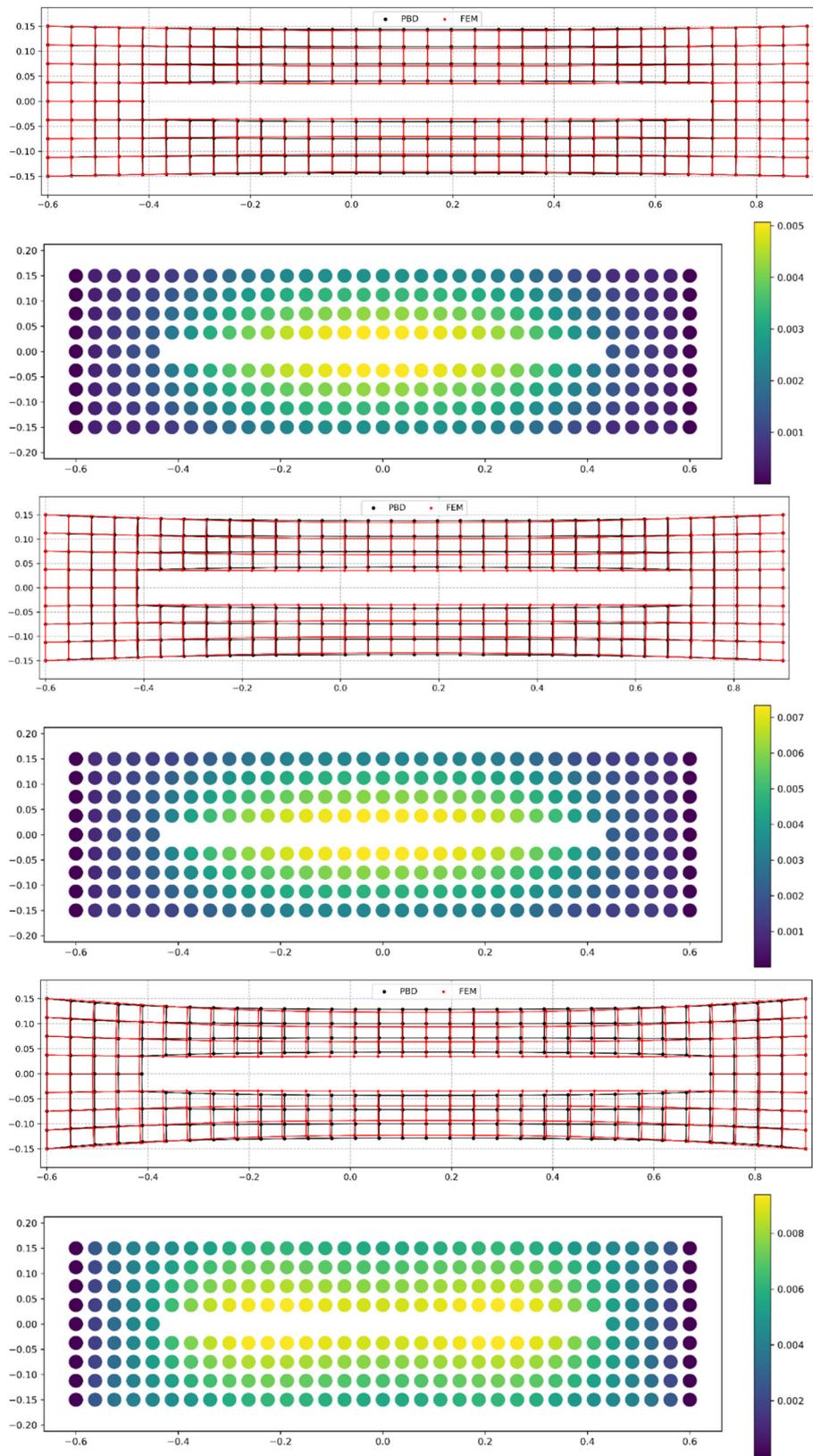


**Fig. 3** Final configuration of the sample for the three Poisson coefficient (0.2, 0.3 and 0.4). Red lattice is the FEM solution while the black is by our tool. Differences between them are shown in coloured plots. Second gradient case. Units on the axes are metres (colour figure online)

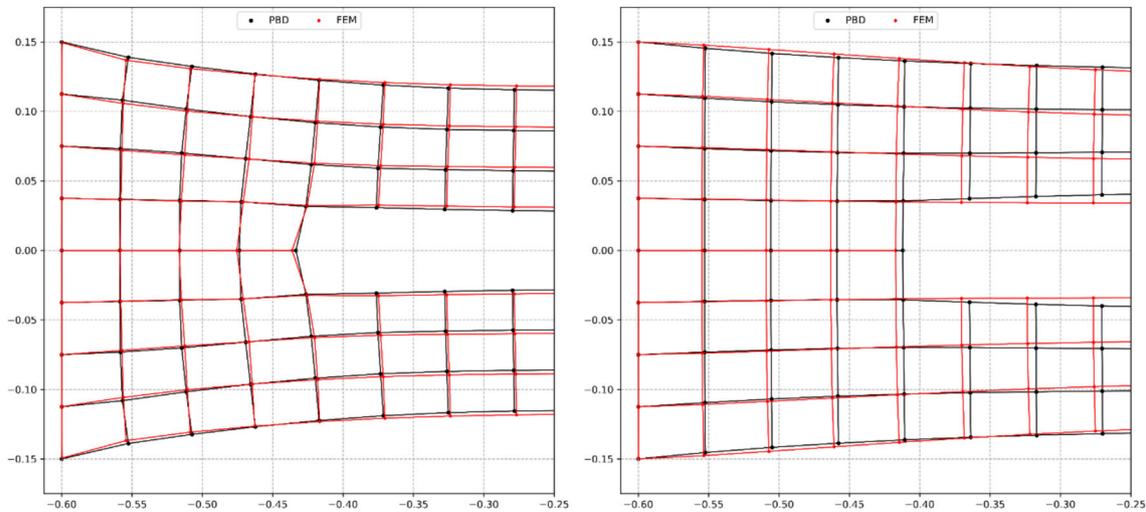


**Fig. 4** Final configuration of the sample for the three Poisson coefficient (0.2, 0.3 and 0.4). Red lattice is the FEM solution while the black is by our tool. Differences between them are shown in coloured plots. First gradient case with slot. Units on the axes are metres (colour figure online)

# A comparison between the finite element



**Fig. 5** Final configuration of the sample for the three Poisson coefficient (0.2, 0.3 and 0.4). Red lattice is the FEM solution while the black is by our tool. Differences between them are shown in coloured plots. Second gradient case with slot. Units on the axes are metres (colour figure online)



**Fig. 6** Differences between FEM solution (red) and our solution (black) for the case of higher Poisson coefficient (0.4) in the first gradient case (left) and second gradient case (right). Units on the axes are metres (colour figure online)

the Poisson’s modulus, we can see an important qualitative difference regarding the deformation of the slot. In our kinematic model, it assumes a convex configuration while in the FEM model it remains a rectangular slot. Look at Figs. 4 and 5. We do not have an explanation for this.

Focusing our attention on a particular area of our sample, in the worst case with the highest Poisson’s modulus, we can see the quantitative and qualitative differences described above, see Fig. 6. Moreover, look now at this interesting phenomenon, in Table 2. In the FEM analysis, we assigned the Poisson coefficient and performed the simulation; if we measure “a posteriori” the Poisson coefficient from its geometrical meaning, as a variation of the sample width, we get a rather different value from the nominal one used in the calculation. This is because the deformation is not purely uniaxial. Again, the agreement with our model (without a nominally assigned Poisson coefficient) is very good as can be seen in Table 2. In the case of the second gradient, the differences between the nominal and measured values are even greater due to the higher stiffness of the sample.

In Table 2,  $\nu_N$  is the nominal Poisson coefficient that was used in the constitutive continuum equation for FEM analysis,  $\nu_S$  is measured by FEM simulation and  $\nu_M$  is measured by our tool. From Table 2, we note that the trend of these values closes to 0.4 is different going towards its limit corresponding to the incompressible behaviour. Note as  $\beta = 1.3$  in the second gradient and 0 in the first gradient to approximate FEM solutions.

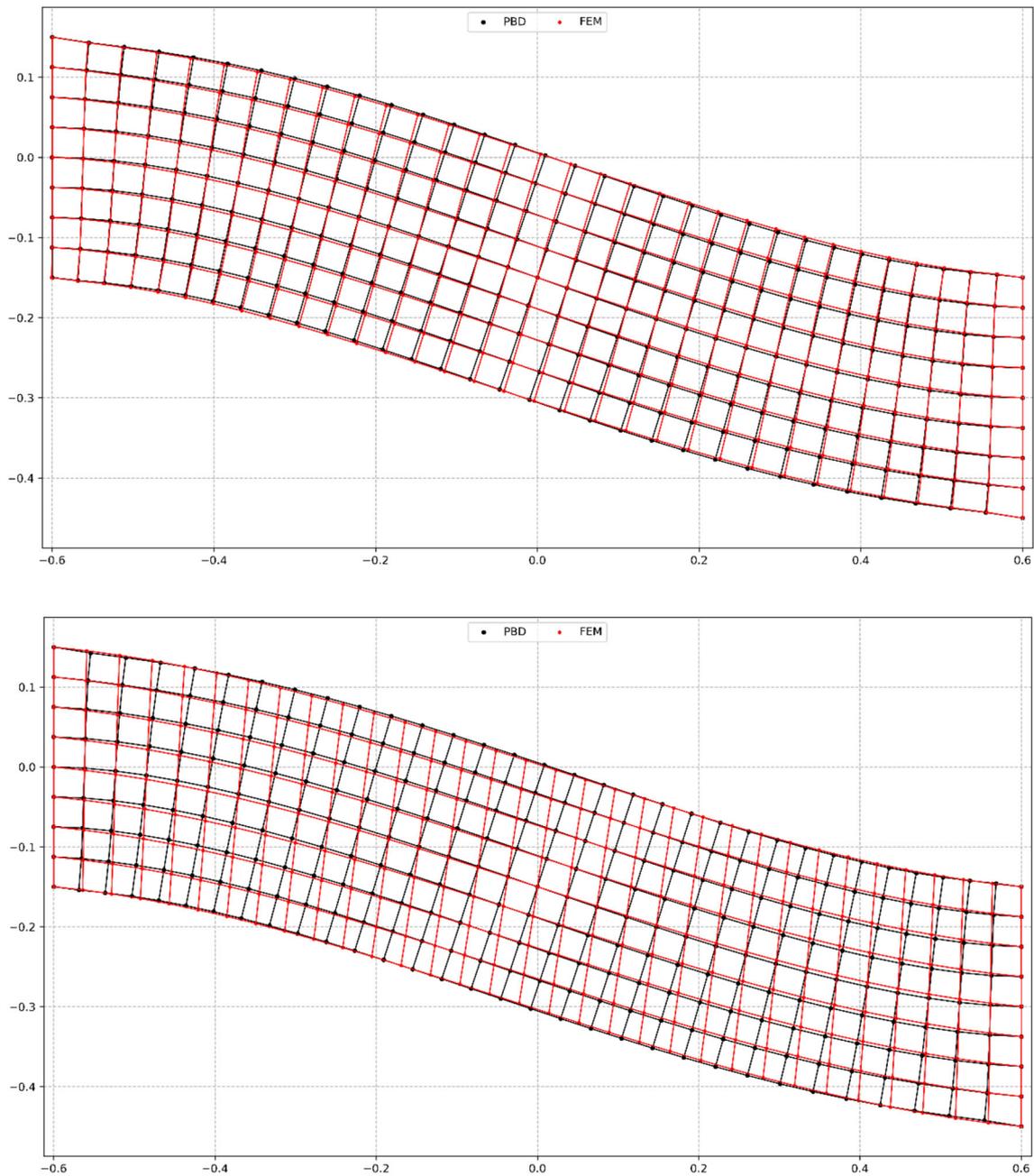
### 3.4 Limit of the model: an example

In this section, we intend to give some insights about the limitations of our model, particularly on the second gradient effects. For this purpose, we consider the same specimen as in the previous cases subjected to a shear test, with Poisson coefficient 0.4. We can see that the deformed external shape of the specimen, solved by means of the COMSOL system, is in very good agreement with our model in both the first and second gradient cases (Fig. 7). However, the internal strain distribution is quite different, especially in the case of the second gradient. In particular, our model shows excessive rotation of the vertical sections, compared with the solution found with the COMSOL system. Probably, the linearity of the rule (Eq. 4) fails to capture some aspects of the strain. In this paper, we stop here but these results deserve more attention in a future paper.

## 4 Conclusion

This paper deals with the possibility of describing a deformable body employing the rules governing the behaviour of a swarm of robots.

We have presented a simple algorithm easily adaptable to different problems to describe complex physical effects plausibly. This discrete model is purely kinematic, so only the positions of the particles describing the body appear. Deformation is applied via a displacement assigned to the leading particles, while the follower



**Fig. 7** Final configuration of the sample undergone a shear test. First gradient case (up) and second gradient case (down). Units on the axes are metres

particles move according to “position rules” until a final equilibrium configuration is reached. These displacements are determined by the relative position of the particle’s neighbours, as in a swarm of birds. Therefore, the deformed configuration is not computed from Newton’s law, but only from the relative positions between the particles in the system, saving machine time for computation since this approach requires only the solution of an algebraic equation. The model is also easily extensible in the case of second gradient continua.

In this work, we compared the results obtained by solving a simple tensile problem by FEM and using our kinematic tool. Different values of Poisson’s modulus were used and applied to a compact sample and another with a crack in the centre to compare the solutions at critical points. The results obtained are in good agreement, allowing us to hope that an improvement in the model, together with an understanding of the physics related

to its parameters, may lead us to satisfactory quantitative predictions. Having demonstrated that we are able to obtain the same results, it is now necessary to find the physical meaning of the parameters of our model to make predictions of behaviour starting from defined initial conditions. Therefore, the results presented in this paper could be a promising theoretical tool for discrete approximation of continuous deformable bodies with generalized strain energy, encompassing non-Cauchy continua. In our opinion, what is attractive in the proposed model is its remarkable simplicity and flexibility in the choice of the type of interaction considered and the contact actions imposed. It allows us to describe non-local interaction and the use of second neighbours' particles can be interpreted as a second gradient theory, even though, in this case, the comparison results with the FEM approach are not as good as in the case of the first gradient model.

The proposed model can be enriched and generalized in many ways; more complex interactions can be considered, for example, cases where the interaction can be weighted by local density, distance, directionality, or other characteristics. A micromorphic approach is the natural generalization of the tool, where a material body is described as a continuous collection of deformable particles of finite size endowed with internal structure.

In our paper, we showed how flocking rules, used to determine the position in a robot swarm, can describe the deformation of a bidimensional continuum.

Our tool has many similarities to PBD, but unlike the PBD methods used in computer graphics, we do not require knowledge of velocity and we do not introduce any force to account for external actions. We remain in the framework of a fully kinematic system.

In this article only, static final configurations are shown. Here, we are not interested in evolution, but the model is able to show evolution in pseudo-time and specimen failure (see [18]).

We want to emphasize again that our model does not have a physical meaning yet, but we are working on a physical analogy. This is because we aim to create a complex swarm-based algorithm that can describe material deformations and not obtain another algorithm based on solving physical equations. Clearly, to assay the goodness of the model, we have to compare it with already available proven models and keep in mind that it is still developing.

The results presented are interesting but are still at a preliminary stage; however, they show good agreement with the deformations calculated by traditional methods. It is clear that in order to have a predictive theory of behaviour, we should know the displacement of the leaders instead of imposing it a priori, as done in this paper. The most important future development is to understand how the constitutive parameters of the materials are related to the parameter choices we make in our tool so that we can connect with the usual methods of continuum mechanics. However, the tool has shown enough flexibility to suggest that many other behaviours can be described once the association with the constituent parameters is established. Generalization to 3D should present no difficulty, but there is a need for some optimization in the code to keep computation time in the order of seconds using a standard desktop PC.

The model, based on the behaviour of robotic systems introduced here, will need further investigation and generalization in both theoretical and numerical aspects. Still, it promises to be of great interest.

**Funding** Open access funding provided by Ente per le Nuove Tecnologie, l'Energia e l'Ambiente within the CRUI-CARE Agreement.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Bender, J., Müller, M., Otaduy, M.A., Teschner, M., Macklin, M.: A survey on position-based simulation methods in computer graphics: a survey on position-based simulation methods in computer graphics. *Comput. Graph. Forum* **33**(6), 228–251 (2014). <https://doi.org/10.1111/cgf.12346>
2. Bender, J., Müller, M., Macklin, M.: Position-based simulation methods in computer graphics. In: *Eurographics (Tutorials)*, (2015) [Online]. Available: [https://www.researchgate.net/profile/Jan\\_Bender/publication/274940214\\_Position-Based\\_Simulation\\_Methods\\_in\\_Computer\\_Graphics/links/552cc4a40cf29b22c9c466df/Position-Based-Simulation-Methods-in-Computer-Graphics.pdf](https://www.researchgate.net/profile/Jan_Bender/publication/274940214_Position-Based_Simulation_Methods_in_Computer_Graphics/links/552cc4a40cf29b22c9c466df/Position-Based-Simulation-Methods-in-Computer-Graphics.pdf). Accessed Sep 6, 2017

3. Rivers, A. R., James, D.: FastLSM: fast lattice shape matching for robust real-time deformation, vol. 26. (2007)
4. Turco, E., Barchiesi, E.: Equilibrium paths of Hencky pantographic beams in a three-point bending problem. *Math. Mech. Complex Syst.* **7**(4), 287–310 (2019)
5. Turco, E., Misra, A., Pawlikowski, M., Dell’Isola, F., Hild, F.: Enhanced Piola–Hencky discrete models for pantographic sheets with pivots without deformation energy: numerics and experiments. *Int. J. Solids Struct.* **147**, 94–109 (2018)
6. Turco, E.: ‘Modelling of two-dimensional Timoshenko beams in Hencky fashion’, in developments and novel approaches in nonlinear solid body mechanics, pp. 159–177. Springer, Cham (2020)
7. Turco, E., Dell’Isola, F., Cazzani, A., Rizzi, N.L.: Hencky-type discrete model for pantographic structures: numerical comparison with second gradient continuum models. *Z. Angew. Math. Phys.* **67**(4), 85 (2016)
8. Turco, E.: Discrete is it enough? The revival of Piola–Hencky keynotes to analyze three-dimensional *Elastica*. *Contin. Mech. Thermodyn.* **30**(5), 1039–1057 (2018)
9. Greco, L., Cuomo, M.: An implicit  $G^1$ -conforming bi-cubic interpolation for the analysis of smooth and folded Kirchhoff–Love shell assemblies. *Comput. Methods Appl. Mech. Eng.* **373**, 113476 (2021)
10. Cuomo, M., Greco, L.: An implicit strong  $G^1$ -conforming formulation for the analysis of the Kirchhoff plate model. *Contin. Mech. Thermodyn.* **32**(3), 621–645 (2020)
11. Greco, L., Cuomo, M., Contrafatto, L.: A reconstructed local B formulation for isogeometric Kirchhoff–Love shells. *Comput. Methods Appl. Mech. Eng.* **332**, 462–487 (2018)
12. Yildizdag, M.E., Demirtas, M., Ergin, A.: Multipatch discontinuous Galerkin isogeometric analysis of composite laminates. *Contin. Mech. Thermodyn.* **32**(3), 607–620 (2020)
13. Obohat, M.A., Tahvilian, E., Yildizdag, M.E., Ergin, A.: Three-dimensional multi-patch isogeometric analysis of composite laminates with a discontinuous Galerkin approach. *Proc. Inst. Mech. Eng. Part M J. Eng. Maritime Environ.* **235**(4), 820–833 (2021)
14. Abali, B.E., Müller, W.H., Dell’Isola, F.: Theory and computation of higher gradient elasticity theories based on action principles. *Arch. Appl. Mech.* **87**(9), 1495–1510 (2017)
15. Park, C.Y., Zohdi, T.I.: Semi-implicit operator splitting for the simulation of Herschel–Bulkley flows with smoothed particle hydrodynamics. *Comput. Part. Mech.* **7**(4), 699–704 (2020). <https://doi.org/10.1007/s40571-019-00301-9>
16. Jakobsen, T.: Advanced character physics. In: Game Developers Conference 2001 (2001)
17. dell’Erba, R.: Swarm robotics and complex behaviour of continuum material. *Contin. Mech. Thermodyn.* **31**, 989–1014 (2019). <https://doi.org/10.1007/s00161-018-0675-1>
18. dell’Erba, R.: Position-based dynamic of a particle system: a configurable algorithm to describe complex behaviour of continuum material starting from swarm robotics. *Contin. Mech. Thermodyn.* **30**, 1069–1090 (2018). <https://doi.org/10.1007/s00161-018-0663-5>
19. Battista, A., Rosa, L., dell’Erba, R., Greco, L.: Numerical investigation of a particle system compared with first and second gradient continua: Deformation and fracture phenomena\*. *Math. Mech. Solids* **22**, 2120–2134 (2017). <https://doi.org/10.1177/1081286516657889>
20. D’Avanzo, P., Rapisarda, A.C., Sirletti, S.S.: Fracture phenomena in Swarms. In: Altenbach, H., Eremeyev, V.A., Galybin, A., Vasiliev, A. (eds.) *Advanced Materials Modelling for Mechanical, Medical and Biological Applications*, pp. 99–167. Springer International Publishing, Cham (2022). [https://doi.org/10.1007/978-3-030-81705-3\\_8](https://doi.org/10.1007/978-3-030-81705-3_8)
21. Wiech, J., Eremeyev, V.A., Giorgio, I.: Virtual spring damper method for nonholonomic robotic swarm self-organization and leader following. *Contin. Mech. Thermodyn.* **30**(5), 1091–1102 (2018). <https://doi.org/10.1007/s00161-018-0664-4>
22. Abali, B.E., Vazic, B., Newell, P.: Influence of microstructure on size effect for metamaterials applied in composite structures. *Mech. Res. Commun.* **122**, 103877 (2022). <https://doi.org/10.1016/j.mechrescom.2022.103877>
23. Spagnuolo, M., Cazzani, A.M.: Contact interactions in complex fibrous metamaterials. *Continuum Mech. Thermodyn.* **33**(4), 1873–1889 (2021)
24. Spagnuolo, M., Barcz, K., Pfaff, A., Dell’Isola, F., Franciosi, P.: Qualitative pivot damage analysis in aluminum printed pantographic sheets: numerics and experiments. *Mech. Res. Commun.* **83**, 47–52 (2017)
25. Placidi, L., Barchiesi, E., Misra, A.: A strain gradient variational approach to damage: a comparison with damage gradient models and numerical results. *Math. Mech. Complex Syst.* **6**(2), 77–100 (2018)
26. Placidi, L., Misra, A., Barchiesi, E.: Two-dimensional strain gradient damage modeling: a variational approach. *Z. Angew. Math. Phys.* **69**(3), 1–19 (2018)
27. Timofeev, D., Barchiesi, E., Misra, A., Placidi, L.: Hemivariational continuum approach for granular solids with damage-induced anisotropy evolution. *Math. Mech. Solids* **26**(5), 738–770 (2021)
28. Vazic, B., Abali, B.E., Newell, P.: Generalized thermo-mechanical framework for heterogeneous materials through asymptotic homogenization. *Contin. Mech. Thermodyn.* **35**(1), 159–181 (2023). <https://doi.org/10.1007/s00161-022-01171-y>
29. Reda, H., Karathanasopoulos, N., Rahali, Y., Ganghoffer, J.F., Lakiss, H.: Influence of first to second gradient coupling energy terms on the wave propagation of three-dimensional non-centrosymmetric architected materials. *Int. J. Eng. Sci.* **128**, 151–164 (2018)
30. Reda, H., Rahali, Y., Ganghoffer, J.F., Lakiss, H.: Wave propagation in 3D viscoelastic auxetic and textile materials by homogenized continuum micropolar models. *Compos. Struct.* **141**, 328–345 (2016)
31. Laudato, M., Manzari, L., Barchiesi, E., Di Cosmo, F., Göransson, P.: First experimental observation of the dynamical behavior of a pantographic metamaterial. *Mech. Res. Commun.* **94**, 125–127 (2018)
32. Barchiesi, E., Laudato, M., Cosmo, F.: Wave dispersion in non-linear pantographic beams. *Mech. Res. Commun.* (2018). <https://doi.org/10.1016/j.mechrescom.2018.11.002>
33. Laudato, M., Barchiesi, E.: Non-linear dynamics of pantographic fabrics: Modelling and numerical study’, in *Wave dynamics, mechanics and physics of microstructured metamaterials*, pp. 241–254. Springer, Cham (2019)
34. Eugster, S.R.: Numerical analysis of nonlinear wave propagation in a pantographic sheet. *Math. Mech. Complex Syst.* **9**(3), 293–310 (2022)
35. Arefi, M., Bidgoli, E.M.R., Dimitri, R., Tornabene, F.: Free vibrations of functionally graded polymer composite nanoplates reinforced with graphene nanoplatelets. *Aerosp. Sci. Technol.* **81**, 108–117 (2018)

36. Fantuzzi, N., Tornabene, F., Baccocchi, M., Dimitri, R.: Free vibration analysis of arbitrarily shaped functionally graded carbon nanotube-reinforced plates. *Compos. B Eng.* **115**, 384–408 (2017)
37. Germain, P.: The method of virtual power in continuum mechanics. Part 2: microstructure. *SIAM J. Appl. Math.* **25**(3), 556–575 (1973)
38. Epstein, M., Smelser, R.: An appreciation and discussion of Paul Germain's The method of virtual power in the mechanics of continuous media, I: second-gradient theory. *Math. Mech. Complex Syst.* **8**(2), 191–199 (2020)
39. Barchiesi, E., Eugster, S.R., Placidi, L., dell'Isola, F.: Pantographic beam: a complete second gradient 1D-continuum in plane. *Z. Angew. Math. Phys.* **70**(5), 1–24 (2019). <https://doi.org/10.1007/s00033-019-1181-4>
40. dell'Isola, F., Seppecher, P., Della Corte, A.: The postulations á la D'Alembert and á la Cauchy for higher gradient continuum theories are equivalent: a review of existing results. *Proc. R. Soc. A Math. Phys. Eng. Sci.* **471**(2183), 20150415 (2015)
41. Auffray, N., dell'Isola, F., Eremeyev, V.A., Madeo, A., Rosi, G.: Analytical continuum mechanics à la Hamilton–Piola least action principle for second gradient continua and capillary fluids. *Math. Mech. Solids* **20**(4), 375–417 (2015)
42. Eugster, S., Steigmann, D.: Continuum theory for mechanical metamaterials with a cubic lattice substructure. *Math. Mech. Complex Syst.* **7**(1), 75–98 (2019)
43. Eremeyev, V.A., dell'Isola, F.: On weak solutions of the boundary value problem within linear dilatational strain gradient elasticity for polyhedral Lipschitz domains. *Math. Mech. Solids* **27**, 433–445 (2021)
44. Abdoul-Anziz, H., Seppecher, P.: Strain gradient and generalized continua obtained by homogenizing frame lattices. *Math. Mech. Complex Syst.* **6**(3), 213–250 (2018)
45. Eugster, S.R., dell'Isola, F., Fedele, R., Seppecher, P.: Piola transformations in second-gradient continua. *Mech. Res. Commun.* **120**, 103836 (2022). <https://doi.org/10.1016/j.mechrescom.2022.103836>
46. dell'Isola, F., Eugster, S.R., Fedele, R., Seppecher, P.: Second-gradient continua: From Lagrangian to Eulerian and back. *Math. Mech. Solids* **27**(12), 2715–2750 (2022)
47. Laudato, M., Ciallella, A.: 'Perspectives in generalized continua', in *Developments and Novel approaches in biomechanics and metamaterials*, pp. 1–13. Springer, Cham (2020)
48. Ciallella, A.: Research perspective on multiphysics and multiscale materials: a paradigmatic case. *Contin. Mech. Thermodyn.* **32**(3), 527–539 (2020)
49. Massoumi, S., La Valle, G.: Static analysis of 2D micropolar model for describing granular media by considering relative rotations. *Mech. Res. Commun.* **119**, 103812 (2022)
50. La Valle, G., Massoumi, S.: A new deformation measure for micropolar plates subjected to in-plane loads. *Contin. Mech. Thermodyn.* **34**(1), 243–257 (2022)
51. dell'Isola, F., Andreaus, U., Placidi, L.: At the origins and in the vanguard of peridynamics, non-local and higher-gradient continuum mechanics: an underestimated and still topical contribution of Gabrio Piola. *Math. Mech. Solids* **20**(8), 887–928 (2015)
52. Pisano, A.A., Fuschi, P.: Structural symmetry within nonlocal integral elasticity: theoretical issues and computational strategies. *Curved Layer. Struct.* **4**(1), 1–7 (2017). <https://doi.org/10.1515/cls-2017-0001>
53. Pisano, A., Sofi, A., Fuschi, P.: Finite element solutions for nonhomogeneous nonlocal elastic problems. *Mech. Res. Commun.* **36**, 755–761 (2009). <https://doi.org/10.1016/j.mechrescom.2009.06.003>
54. Oterkus, S., Madenci, E., Oterkus, E.: Fully coupled poroelastic peridynamic formulation for fluid-filled fractures. *Eng. Geol.* **225**, 19–28 (2017). <https://doi.org/10.1016/j.enggeo.2017.02.001>
55. Diyaroglu, C., Oterkus, S., Oterkus, E., Madenci, E.: Peridynamic modeling of diffusion by using finite-element analysis. *IEEE Trans. Compon. Packag. Manuf. Technol.* **7**(11), 1823–1831 (2017)
56. Yang, H., Abali, B.E., Timofeev, D., Müller, W.H.: Determination of metamaterial parameters by means of a homogenization approach based on asymptotic analysis. *Contin. Mech. Thermodyn.* **32**(5), 1251–1270 (2020)
57. Abali, B.E., Yang, H., Papadopoulos, P.: 'A computational approach for determination of parameters in generalized mechanics', in *Higher gradient materials and related generalized continua*, pp. 1–18. Springer, Cham (2019)
58. Abali, B.E.: Revealing the physical insight of a length-scale parameter in metamaterials by exploiting the variational formulation. *Contin. Mech. Thermodyn.* **31**(4), 885–894 (2019)
59. Shekarchizadeh, N., Abali, B.E., Barchiesi, E., Bersani, A.M.: Inverse analysis of metamaterials and parameter determination by means of an automatized optimization problem. *ZAMM J. Appl. Math. Mech. Z. für Angew. Math. und Mech.* **101**(8), 202000277 (2021)
60. De Angelo, M., Placidi, L., NejadSadeghi, N., Misra, A.: Non-standard Timoshenko beam model for chiral metamaterial: identification of stiffness parameters. *Mech. Res. Commun.* **103**, 103462 (2020)
61. Barchiesi, E., Misra, A., Placidi, L., Turco, E.: Granular micromechanics-based identification of isotropic strain gradient parameters for elastic geometrically nonlinear deformations. *ZAMM J. Appl. Math. Mech. Z. für Angew. Math. und Mech.* **101**(11), 202100059 (2021)
62. Placidi, L., Andreaus, U., Della Corte, A., Lekszycki, T.: Gedanken experiments for the determination of two-dimensional linear second gradient elasticity coefficients. *Z. für Angew. Math. und Phys.* **66**(6), 3699–3725 (2015)
63. dell'Erba, R.: Determination of spatial configuration of an underwater swarm with minimum data. *Int. J. Adv. Rob. Syst.* **12**(7), 97 (2015)
64. D'Avanzo, P., Rapisarda, A.C., Sirletti, S.S.: From the Swarm robotics to material deformations. In: Marmo, F., Sessa, S., Barchiesi, E., Spagnuolo, M. (eds.) *Mathematical Applications in Continuum and Structural Mechanics*, pp. 87–125. Springer International Publishing, Cham (2021). [https://doi.org/10.1007/978-3-030-42707-8\\_6](https://doi.org/10.1007/978-3-030-42707-8_6)