

Visualization of Particle Interactions in Granular Media

Holger A. Meier, Michael Schlemmer, Christian Wagner, Andreas Kerren, *Member, IEEE*, Hans Hagen, *Fellow, IEEE*, Ellen Kuhl, and Paul Steinmann

Abstract—Interaction between particles in so-called granular media such as soil and sand plays an important role in the context of geomechanical phenomena and numerous industrial applications. A two-scale homogenization approach based on a micro and a macroscale level is briefly introduced in this paper. The behavior of the granular matter at the microscale level is captured by a discrete element method, allowing the simulation of breaking and forming of contacts between the single grains. The problem at the macroscale level is discretized by the finite element method. A computation of granular material in such a way gives a deeper insight into the context of discontinuous materials and at the same time reduces the computational costs. However, the description and the understanding of the phenomena in granular materials are not yet satisfactory. A sophisticated problem-specific visualization technique would significantly help to illustrate failure phenomena at the microscopic level. As main contribution, we present a novel 2D approach for the visualization of simulation data, based on the above outlined homogenization technique. Our visualization tool supports visualization at the microscale level, as well as at the macroscale level. The tool shows both aspects closely arranged in the form of multiple coordinated views to give users the possibility to analyze the particle behavior effectively. A novel type of interactive rose diagram was developed to represent the dynamic contact networks at the microscale level in a condensed and efficient way.

Index Terms—Visualization, coordinated views, time-dependent data, rose diagram, mechanics, granular media, particle interaction.

1 INTRODUCTION

THE simulation and analysis of the behavior of granular material is highly important for research in the field of geomechanical engineering. However, the commonly used numerical techniques are insufficient for a satisfying simulation of granular matter such as soil, powder, or sand. Our recently developed *two-scale simulation approach* for granular media based on a micro and a macroscale level improves this situation. At the macroscopic level, our approach utilizes a finite element method for efficient and robust large-scale engineering computations. The finite element method is essentially based on the continuum hypothesis, i.e., it implies that matter is distributed continuously in space and can be characterized rigorously through a set of field equations in terms of continuous quantities. When thinking of granular media, it is important to realize that these can be hardly

thought as continuous. In order to allow a more precise representation, we *zoom in to the microscopic simulation*, where matter is inherently discontinuous. At this second scale, simulations are performed with the discrete element method, which essentially evaluates particle-to-particle interactions. A discontinuous technique such as the discrete element method is able to represent discrete failure phenomena such as crack propagation more accurately than continuous strategies. In this sense, statements about particles can be regarded as a kind of constitutive law for the macroscopic level. The appendix of this article briefly outlines our two-scale simulation approach for granular media.

Although research in the area of *multiscale simulation* has been quite intense within the past decade, there is hardly any progress in the development of proper and efficient visualization toolkits. From an engineering point of view, a corresponding *multiscale visualization tool* for such simulations would significantly improve the understanding of complex failure phenomena: It allows *zooming in to the microscopic visualization* and may even support the development of new more sophisticated material models.

Individual visualization tools are available for the continuous macroscopic, as well as for the discontinuous microscopic level, but there is still no single visualization tool that provides a linkage between these two scales. Macroscopic visualization, which is typically based on relevant scalar, vector, and tensor fields, has been solved quite satisfactorily. However, microscopic visualization is still in its infancy and commonly restricted to provide an overview of the particle structure. The efficient combination of macroscopic and microscopic visualization tools is usually hampered by incompatible data structures.

The present research project aims at providing new visualization metaphors and introduces interaction patterns for the objective of effective and insightful multiscale visualization. An innovative combination and connection of

- H.A. Meier is with the University of Kaiserslautern, Department of Mechanical Engineering, PO Box 3049, D-67653, Kaiserslautern, Germany. E-mail: homei@rhrk.uni-kl.de.
- M. Schlemmer, C. Wagner, and H. Hagen are with the University of Kaiserslautern, Department of Computer Sciences, PO Box 3049, D-67653 Kaiserslautern, Germany. E-mail: [schlemmer, c_wagne, hagen}@informatik.uni-kl.de](mailto:{schlemmer, c_wagne, hagen}@informatik.uni-kl.de).
- A. Kerren is with Växjö University, School of Mathematics and Systems Engineering (MSI), Computer Science Department, Vejdes Plats 7, SE-351 95 Växjö, Sweden. E-mail: andreas.kerren@vxu.se.
- E. Kuhl is with Stanford University, Departments of Mechanical Engineering and Bioengineering, 496 Lomita Mall, Durand 217, Stanford, CA 94305. E-mail: ekuhl@stanford.edu.
- P. Steinmann is with the University of Erlangen-Nuremberg, Egerlandstr. 5, 91058 Erlangen, Germany. E-mail: paul.steinmann@ltn.uni-erlangen.de.

Manuscript received 28 Aug. 2007; revised 10 Feb. 2008; accepted 8 Apr. 2008; published online 22 Apr. 2008.

Recommended for acceptance by H. Hauser.

For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org, and reference IEEECS Log Number TVCG-2007-08-0115.

Digital Object Identifier no. 10.1109/TVCG.2008.65.

two views, namely, a Macroscale-Level View (finite element level) and a Microscale-Level View (discrete element level), help to improve research in the application area. As a major contribution, we present a novel type of the so-called *rose diagram*, being scalable and independent from the global coordinates of the simulation. Instances of this technique have been used in Information and Software Visualization, see, for example, the work of Chuah and Eick [6], [7]. We applied and extended the rose diagram to operate on data arising from the microscale-level analysis. Accordingly, our work also implies a linkage between Scientific Visualization (SciVis) and Information Visualization (InfoVis): our modified rose diagrams enable a selective visualization of force directions and revealing structures, as well as offer new viewpoints for engineers to efficiently analyze large contact networks of granular data sets. For the generation of the rose diagrams, a novel sweep line algorithm is introduced. This generation process includes an adapted cluster algorithm to produce numerically stable rose diagrams, even for large amounts of data, cf., Section 4.2. Interaction on the rose diagrams allows an enhanced visualization of a linked particle view. While the rose diagram gives an abstract visualization of the particle contacts, a *Granular View* shows them directly. Here, single particles are shown as individuals nested in the contact network. This kind of visualization is a typical representative of SciVis. Both representations, Granular View and Diagram View, are combined together in the Microscale-Level View. An interactive linkage between Micro and Macroscale-Level View is provided by displaying updated microscale rose diagrams as *rose glyphs* at the macro level. These techniques derived from multiple coordinated view research reveal local, as well as global interactions in the granular assemblies. Both simulation tools provide time-dependent data sets, manifesting themselves in individual *load steps*. Sequences of microscopic and macroscopic load steps are animated and controlled with a *VCR metaphor*, such as for playing, pausing, or rewinding.

1.1 Background Information

Provided in the following is a brief overview of the needed mechanical procedures. Keeping in mind that the focal point of this contribution is the visualization of data sets originating from multiscale computations (containing a macro and a microscale level) deeper insight of the related mechanics is found in the references and within Appendix A of this article. In particular, our visualization approach finds its main contribution at the microscale level.

The *two-scale homogenization*, detailed in Appendix A.1, incorporates a finite and discrete element approach. A standard *finite element method* (fem) is utilized at the macroscale level, assuming an overall homogeneous and continuous material. In general, fem is used for finding approximate solutions of partial differential equations, see, for example, [19], [27], [32]. Such solutions are typically displayed in the form of meshes, as shown in Fig. 1. A single finite element is one cell in the underlying mesh, cf., Fig. 3. Thus, fem is the method of choice for complex systems in structural mechanics that can be modeled by a set of appropriate finite elements interconnected at points (nodes). Each finite element at the macroscale level has a set of so-called *Gauss points*. Typically, four Gauss points for each finite element are used for the simulation and consequently in our visualization examples, too. In general, their number is not fixed. In the present case, we link each macroscopic Gauss point to a microscopic structure. The

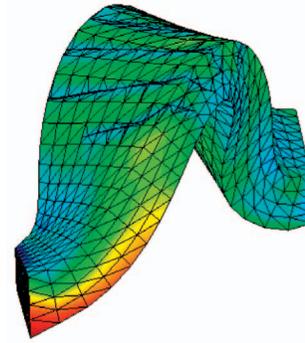


Fig. 1. Visualization of brittle fracture during folding rocks. Colors indicate stress intensity. Tables of stress-induced folding patterns help to identify geologically relevant failure patterns during geophysical field studies. The visualization is particularly important for industrial applications such as groundwater and hydrocarbon flow because fractures control the permeability of rock units. In contrast to the proposed multiscale visualization technique, current algorithms lack information about the microstructural failure mechanism. Taken from that in [21].

microstructure under consideration is assumed to be heterogeneous, as well as discontinuous. This microstructure is modeled by a *discrete element method* (dem, see [10]). Due to the focus of this manuscript, the homogenization approach and the microscale level computation are outlined in Appendices A.1 and A.2. Despite the importance of the finite element approach, we will not address the theme of finite element computation in detail. Rather, we refer the interested reader to standard finite element literature.

This article is organized as follows: Related work concerning the mechanical and visualization aspects are presented in Section 2. Our visualization system and techniques for the micro and macroscale level are illustrated in Section 3, while details of the implementation are given in Section 4. Representative results are discussed followed by a conclusion and ideas for future improvements. The appendix includes a brief summary of the homogenization process from an engineer's point of view. We decided to give this detailed information for the interested reader. These additional insights are provided for a better understanding and interpretation of the screen-shot examples given within this paper.

2 RELATED WORK

In the context of finite and discrete element computation, most visualization tools are intrinsically combined with their computational counterpart. Extensions or changes are limited when it comes to commercial software products. The most prominent commercial software systems used in fem computations, among others, are ABAQUS FEA [1] and ANSYS [2]. Fig. 1 shows a typical example of a fem mesh visualization, including a so-called stress plot. Such plots show forces that have an effect on finite elements. In the case of dem analysis, DEM Solutions [15] and software from Itasca [20] are well-known systems. These systems provide visualizations that are more or less "simple" particle simulations with few possibilities of interaction. To the best knowledge of the authors, there is no software that is capable of performing the visualization or the computation of multiscale problems, including different computational approaches at each scale. Therefore, these standard tools are not able to display the needed two levels at the same time, i.e., there exist no integrated solution so far. Currently, the simultaneously

visual analyses of the microscale level in the context of the macroscale level is very difficult if not impossible: two or more different tools have to be used, and the visual mapping between two or more visual representations is complicated, not to mention data and interface incompatibilities. The main motivation of our new approach was to improve the current situation. Our tool supports a seamless integration of both worlds and, moreover, new visualization and interaction possibilities. Traditional tools cannot provide this and cannot be directly compared with our approach in general. However, these tools have advantages in some aspects, for instance, the display of stress plots. We plan to implement this feature in the future.

The general design of our visualization tool is based on standard coordinated and multiple view visualization techniques. Both scale levels are represented by two main views, whereas the Microscale-Level View consists of several subviews. The scale levels are interconnected using brushing and glyphs. An excellent starting point for related work of this kind of visualization techniques is the annual conference series on Coordinated and Multiple Views in Exploratory Visualization (CMV) or the work of Roberts [29]. However, in our case, we solved an additional challenging problem: Based on the nature of the two-scale simulation approach, we had to find a solution to visualize a 1:n relationship between one simulation step in the Macroscale-Level View and possibly several simulation steps in the Microscale-Level View. We solved this problem by using two nested VCR controls, see Section 3 for detailed explanations.

In this paper, we also present a modified visualization of rose diagrams. These diagrams, introduced by Nightingale [5], [26] for the first time, are commonly employed for displaying data in the form of circular histograms. Rose diagrams are used in statistics but are also a favorite method of depicting orientations, which makes them interesting for our visualization approach. In our case, we use rose diagrams to display the contact distribution at the microscale level, also because experts of our application domain are familiar with static variants of this visual representation. In general, rose diagrams have two main challenges: the decision about the initial starting point and the widths of the circular sectors (rose petals at the end), as well as the scaling to the available space [37]. The most available and used drawing algorithms are simple implementations based on Java applets. Some interesting domain-specific solutions recommend to let the user choose petal width and starting point or to give simple default values [31]. This idea also reflects our vision. Additionally, our improved rose diagram supports the computation of optimized default values for initial petal width and number of petals, as well as extended possibilities in interaction. Its construction is based on a sweep line paradigm, known from computational geometry. An example for a similar application of this class of algorithms is the work of Fortune [16] on the computation of Voronoi diagrams. The book [13] gives a good overview of the various application areas of this paradigm. Furthermore, our method also incorporates a type of hierarchical clustering, i.e., agglomerative clustering, in order to compute the final petal widths. A lot of work has been done based upon agglomerative clustering: the most relevant ideas, incorporated into our approach, are those of Ward [36], where he used a similar variance criterion as in our clustering process. Day and Edelsbrunner [12] present a range of optimized algorithms regarding this topic, including the algorithm of Defays [14].

We discovered during the implementation and testing of our tool that minimized versions of rose diagrams, i.e., rose glyphs, are very well suited for a linkage between micro and macroscale level. They give an impression of the underlying rose diagrams that they abstract. In general, glyphs have been heavily used in InfoVis and SciVis. They are graphical entities that convey multiple data values via visual attributes such as shape, color, position, or size [35]. There is a considerable literature on glyphs and a lot of tools use them. We pick out one example work for a specific domain written by Chuah and Eick [6], [7]. They describe glyphs for a software project management data, where abstract data and time are mapped to specific visual attributes of glyphs. Thus, their 3D wheel glyphs (as two-dimensional (2D) projections) look a little bit similar to our rose glyphs. However, orientation is irrelevant in contrast to our approach. We must represent the correct orientation of contact forces at the microscale level. Glyphmaker [28] is an older exploratory tool that provides a general visualization approach based on the use of glyphs as graphical elements for data representation. The placement of glyphs plays an important role, and Glyphmaker is able to map specific attributes of data onto position in space. This is also an interesting difference to our approach because positioning of rose glyphs is fixed on corresponding Gauss points in the fem mesh. Another general system, called XmdvTool, was initially developed by Ward [34]. It supports several methods for displaying flat form data and hierarchically clustered data such as Star Glyphs or Dimensional Stacking. A later article on a taxonomy of glyph placement strategies [35] presents structure-driven positioning of glyphs. Such a structure can be the (linear) time that is also important in our case. We decided to show rose glyphs that change over time by animation, controlled via VCR controls. According to Ward, a completely different approach would be to show information about all time steps simultaneously in the macrolevel view, as proposed in the SpiralGlyphics system [22]. However, this would be not useful for our requirements because such spirals are space consuming, and we would need four spirals for each Gauss point. Moreover, the strong top-down interaction style (see Section 5), as well as the nested microscopic and macroscopic load steps are reasons speaking against it.

Shaw et al. [30] analyzed glyphs generated by superquadrics [3] in the context of 3D scalar and vector fields to indicate flow. They hope that preattentive shape finding can be supported using glyphs. This work is interesting for us as shape alone is a preattentive feature. Therefore, we are hopeful that perception of our b/w rose glyphs is done preattentively. However, it is possible that the other features of the Macroscale-Level View destroy this effect.

3 VISUALIZATION

While current visualization systems deal solely with either fem or dem visualizations, our approach joins both scale levels. To achieve this, we combine standard SciVis methods with enhanced InfoVis methods. This leads to novel visualization possibilities for data resulting from multiscale computations and to an improved understanding of the complex interrelationship between the different scale levels. To parallel the simulation scheme (see Appendix A), the visualization scheme is divided in two main views: a macroscopic and a microscopic one. The Macroscale-Level View serves as a representation of the fem mesh and offers standard visualization techniques for the common fem approach. Moreover, special previews for the

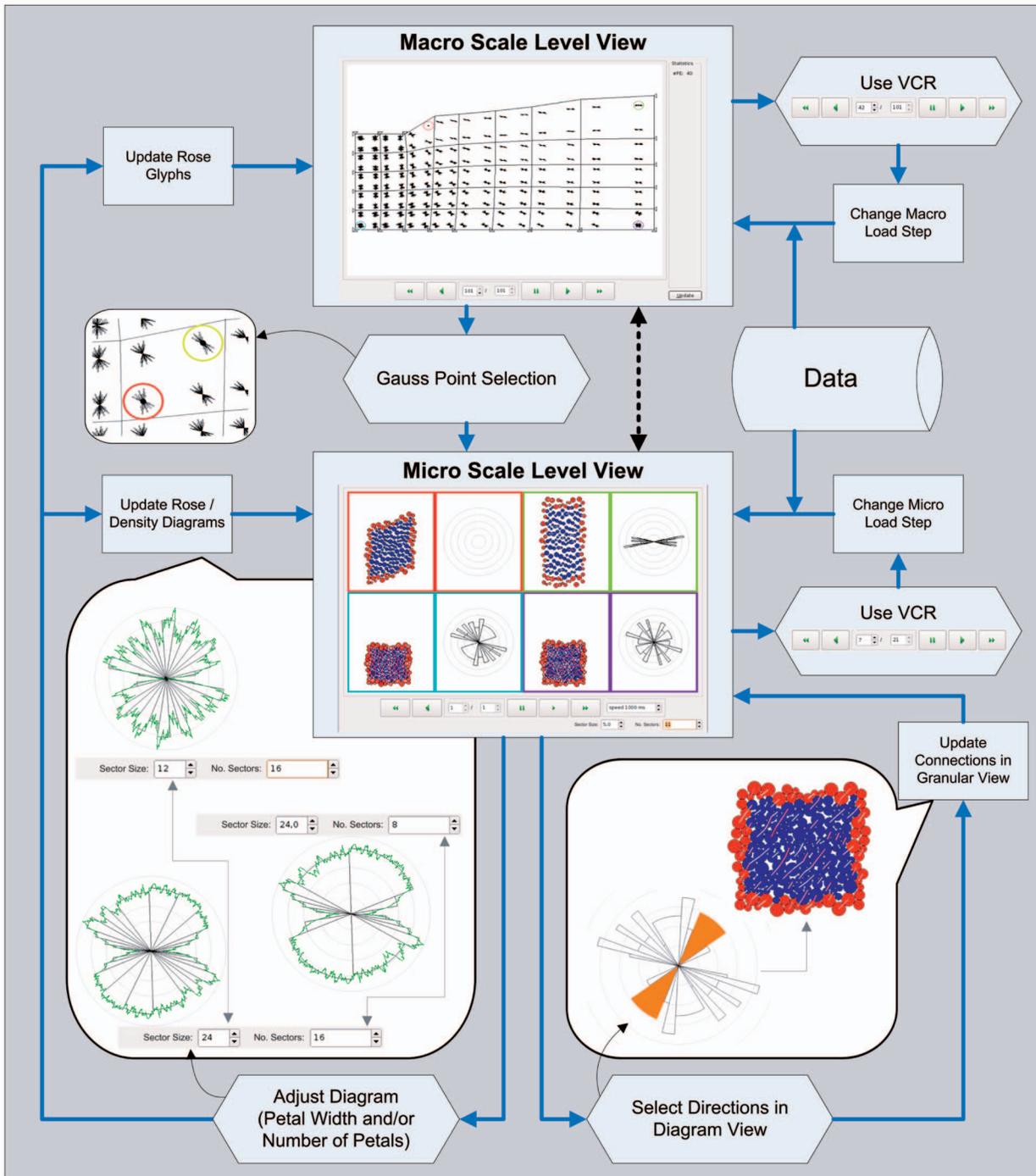


Fig. 2. Interaction with both Macro and Microscale-Level View. This diagram also shows the information flow between the different levels together with possible adjustments for the visual representation. Note that the term “initial petal width” used in this article is labeled as “[Sweep] Sector Size” within our tool. The same holds for the number of petals (“No. Sectors”). We decided to use this terminology in our tool because it reflects better the adjustments of the algorithm discussed in Section 4.

underlying microscale level, so-called *rose glyphs*, have been introduced. For each Gauss point included in the *fem* mesh, a link to the corresponding Microscale-Level View is available. Both levels contain time-variant data, partitioned into so-called *load steps*. Hereby, one load step at the macroscale level corresponds to multiple load steps at the microscale level (1:n relationship). The control of this time variation is accomplished by a *VCR metaphor* introduced in both view levels. Each view level is described in the

following. Furthermore, possible interactions between the visualization levels are specified.

3.1 Interaction and Coordination Aspects

In this section, we give a brief overview about the general usage of our visualization tool and about the relationship between the different components. Fig. 2 guides the reader through possible interaction scenarios. Note that a detailed description of the views is given below.

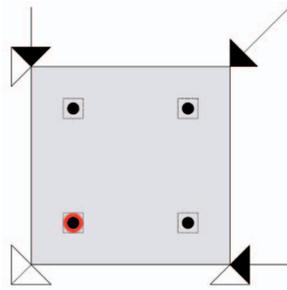


Fig. 3. Single finite element, containing a typical number of Gauss points. The Gauss points are displayed as big black dots. Selected Gauss points are highlighted using a simple color scheme. Bearings are represented by unfilled triangles, while forces are symbolized by arrows.

Usually, the engineer starts with the Macroscale-Level View. The current *fem* mesh based on the simulation data is displayed. Using the VCR controls, the user can play continuously or stepwise all load steps at the macro level. Of course, the tool allows the user to modify the animation speed or to jump to specific load steps directly. Gauss points are displayed as black dots or rose glyphs. It is possible to select an arbitrary number of Gauss points, which are interesting for the user and to switch to the Microscale-Level View. This view shows all granular microstructures (*representative volume elements*, abbreviated as *rves*) together with related rose diagrams of the selected Gauss point set. Thus, each selected Gauss point corresponds to exactly one time-dependent sequence of a *rve*/diagram pair. Analogously to above, our tool supports the exploration of all time steps, i.e., the change of the micro load step by using a second VCR control (nested in the global time scheme). As it normally does not suffice to watch predefined images only, we offer several kinds of interaction: On one hand, the user can interact with the visual representations themselves, for example, by selecting specific sectors of a rose diagram. As a result, only those contact forces within the corresponding *rve* are displayed that have the same direction as the selected sector (defined by its two radii). On the other hand, the tool allows the user to adapt the visual representations, such as by adjusting initial petal widths or the number of petals. Section 3.3 gives detailed explanations about all interaction possibilities within the Microscale-Level View.

To bridge the gap between the two main views and to close the interaction circle, we can provide each Gauss point of the *fem* mesh with a rose glyph in the Macroscale-Level View. It always displays the actual state of the Diagram View at the micro level. Thus, these rose glyphs give the user an excellent overview about the entire simulation process at the macro level. In summary, there is a perfect one-to-one mapping between the following four concepts: a Gauss point, the entire set of particles in a *rve* of a microscopic load step, a full rose diagram (including its density function if displayed, see below) that gives a more abstract impression of the particle contacts of the same *rve*, and a compact rose glyph that is based on that rose diagram.

3.2 Macroscale-Level View

The *fem* mesh together with related features are displayed in the Macroscale-Level View, see also Fig. 12 for a more complex screen-shot example. We explain the most important features in Fig. 3 that shows a single finite element:

- **Bearings.** A bearing is a state or device to prevent/permit constrained relative motion between several parts. Bearings can be classified according to the motions they allow (e.g., rotation or linear movement) and according to their principle of operation. Here, bearings usually prevent the motion of specific *fem* nodes. They are represented in our visualization tool by unfilled triangles, which is the standard representation in structural mechanics. In Fig. 3, we have four bearings that prevent moving the finite element down and to the left.
- **Forces.** External forces can act on the *fem* nodes, for example, if a footing is being pressed into the soil ground. We used (filled) arrows to visualize forces. The direction of the forces is collinear to the arrow directions, while the magnitudes of the forces are represented by the length of the arrows. In our small example, we have three different forces that push against the four bearings.
- **fem mesh.** As already described before, the current *fem* mesh based on the simulation data is displayed in the Macroscale-Level View.
- **Gauss points.** The *fem* mesh contains a specific number of Gauss points for each finite element. Their positions, i.e., the locations where the microscopic constitutive equations are evaluated, are a standardized input to the finite element calculation. In the case of nonlinear discrete, as well as finite element computation, loads and deformations are applied in multiple load steps, ensuring the convergence of the algorithm. These load steps can be thought of as pseudo-time steps, controlled by the VCR control. Each animation step in the Macroscale-Level View is related to a corresponding synchronized animation of the Microscale-Level View. Connections between both scale levels take place at the Gauss points. They are depicted by big black dots or rose glyphs. Both views support linked highlighting of Gauss points and corresponding microstructures/diagrams with a shared color coding.
- **Rose glyphs.** Our tool uses rose glyphs in order to present features of the Microscale-Level View, i.e., rose diagrams with/without density functions. They are superimposed on the associated Gauss points and are created from the same underlying data set with the same method as the large rose diagrams at the microscale level. In short, rose glyphs are minimized versions of rose diagrams. If the user wishes to see them, then they are permanently displayed and modified over time (also if the Microscale-Level View is closed). In this way, the user is able to perceive rough differences in the behavior of the considered material under the current force distribution. These kinds of features are discussed in Section 3.3.

A more concrete and more complex application example with 40 finite elements is described in Section 5.

3.3 Microscale-Level View

The visual analysis of data sets, generated by dem computations at the microscale level, is of great interest to

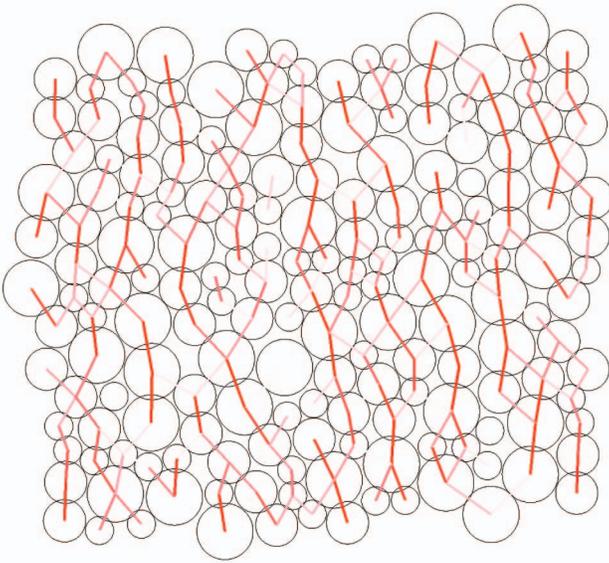


Fig. 4. Granular View of a single rve in overlap mode. The particles are depicted as black bounded circles to facilitate the perception of the overlaps. The magnitude of the overlaps is visualized by a simple color-coding scheme of the branch vectors between the connecting particles. Dark red represents a deep penetration, while light red is associated with a narrow penetration.

researchers in the engineering community. Each Gauss point contributes

- a set of visualization data, i.e., data that needs to be visualized at the macroscale level, see Section 3.2, and
- data gained from the computations at the microscale level at a particular Gauss point. These are necessary to visualize at the microscale level.

Thus, in order to conquer the task of providing an adequate tool to visualize all relevant data, our tool provides a second view level. This second view level solely supplies views of microscopic data for selected Gauss points.

Similar to the visualization procedure at the macroscale level, different load steps have to be visualized at the microscale level. In detail, each load step at the macroscale level contains multiple load steps at the microscale level. For the sake of familiarity, the VCR metaphor, as presented in Section 3.2, is reutilized for this task.

Color coding is used to easily and efficiently relate Gauss points of the finite element mesh at the macroscale level to the rve at the microscale level; the frame color of arbitrarily selected Gauss points corresponds to the color of the box framing the rve. In each of these boxes, two different views are available: Granular View and Diagram View.

3.3.1 Granular View

The first view, called *Granular View*, shows the particles pictured as circles, see Fig. 4. The user can choose between a filled or an unfilled representation. While the filled representation (*standard mode*) gives a good overall impression of the granular assembly, the unfilled representation (*overlap mode*) emphasizes the overlaps between the grains. Additionally, the filled view shows the periodic boundary particles in a different color. The GUI allows a fast change

between both modes. The identification of overlaps is of great practical importance. Their magnitude is directly related to the contact normal force acting between the particles in contact and affect the particle displacement, see Appendix A.2. Note that there is no real “overlap” of two particles in reality. Particles are assumed to be compact, and one particle cannot penetrate another one in real life. However, this name is commonly used in mechanics and serves as a convincing metaphor to describe the processes more pictorially.

Another feature of the Granular View consists of the possibility to show the *particle contact network* directly. The user can opt for showing or hiding connection lines; their main feature is based on depicting the contact connectivity between the grains. Moreover, the user can observe the magnitude of the normal contact forces by enabling a color mapping of this feature onto the drawn connection lines. The color mapping is based on using a saturation gradient, where more saturated red lines (“darker” lines) indicate greater magnitudes of the normal contacts forces and less saturated red lines (“lighter” lines) indicate smaller magnitudes. Mostly, the best perception of the contact network is achieved if the user switches to the overlap mode, i.e., if the particle representations are unfilled. In standard mode, the red particles at the border of the rve make it sometimes difficult to distinguish dark red connection lines from them. Very strong overlaps between periodic boundary particles can thus be overlooked in worst case. During the overall computation, single particles, as well as particle groups move—often stepwise—inside the bounded rve. These processes yield to new particle arrangements and contact networks. Visual inspection includes the analysis of connection lines, running through the particle assembly, or the appearance and disappearance of voids inside the granular structure.

As a welcome side effect, these lines visually enclose voids and nonconnected particles that helps to identify such structures in the rve. For example, the interested reader can find a nonconnected inner particle in the shown Granular View in Fig. 10 near the bottom left corner of the particle contact network. Two nonconnected periodic boundary particles can also be discovered at the periphery of the whole rve.

3.3.2 Diagram View

The second view, titled *Diagram View*, shows a *rose diagram*. The rose diagram, originally introduced by Nightingale [26] in 1858, offers a 2D graphical representation of discrete and discretized circular data. In its definition, the classical rose diagram relates to a modified version of a histogram. Thereby, histogram bars correspond to the petals (i.e., the circular sectors) of the rose diagram. The main difference between rose diagrams and histograms is the intrinsically periodic data representation of the rose diagram. We use an enhanced type of rose diagram, laid out in Section 4, to represent the directions of contact forces between all touching particles within one rve. Depending on the deformed rve (including the changed particle setup inside), a distinguishing load transfer direction inside the granular structure might be found. This load transfer direction is well analyzable with the aid of the rose diagram representation, while in the Granular View, a clear analysis might be

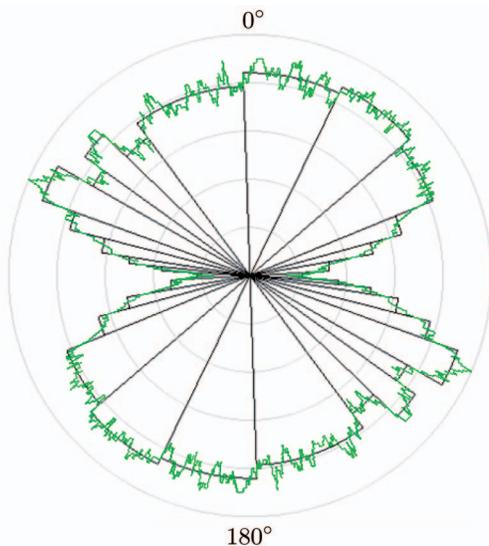


Fig. 5. A rose diagram cut out from a screen shot of the Diagram View. As an initial petal width, the user selected 24, and we have 16 “different” petals because of the reflection symmetry (32 petals in sum) of our rose diagrams. Both values can be adapted by the user in order to allow target-oriented visual analyzes. Additionally, a green colored density function was faded in. This is an optional feature of our tool and can be hidden.

unmanageable. A screen shot given in Fig. 5¹ shows a typical rose diagram provided by our visualization tool. The data source of this rose diagram is the *rve* displayed in Fig. 4. The different visual elements of our example are explained in the following.

Each rose petal represents the *direction* and *weight* of a specific set of individual contact forces. Let us take a closer look at the bigger petals. This diagram shows us that the most contact forces are vertically directed; we only have few horizontal forces. The particle contact network in Fig. 4 substantiates this fact. An exception is the narrow but long petal that nearly reaches the peripheral circle at about 120 degrees. It means that we have a large amount of forces in this specific direction but only few forces in slightly differing directions. In this sense, the individual petal width gives us an overview about similar directed forces (variance) and the petal length about their weight comparing to all force directions. The overall appearance depends on the user defined initial petal width (denoted in angular degrees) and the number of petals that are used in the drawing algorithm, see Section 4. Here, we merely present the high-level idea of the algorithm in the following: At first, compute a directional density function for the Gauss point given the contact forces between all the particles in the *rve*. Merge regions of the density function that have constant values using bottom-up agglomerative clustering to combine the regions of lowest variance, then decrease the number of circular sectors until the target number of petals is reached. Note that we use a different terminology here. From an algorithmic and geometric point of view, we merge *circular sectors* with individual changeable radii and sizes until a break condition is reached. The final set of circular sectors forms the *petals* of our rose diagram. According to Newton’s third law, the resulting rose diagram is reflection

1. Probably, only 14 different petals can be counted in the printed version of this article. Using the tool, the user will easily discover two tiny additional petals located at about 90 degrees.

symmetric because of bidirectional contact forces. Therefore, we only indicate the half number of occurring petals in all examples of this paper, for example, 16 in Fig. 5.

The degree of refinement, i.e., the number of desired petals, and the initial petal width can be arbitrarily determined by the user. The initial petal width can vary in the interval $[0.1, 90]$ and the number of petals in $[1, \infty]$. In fact, the visualization tool offers default values for these two parameters that are computed in consideration of the current overall force distribution. However, in the most cases, the user wishes to change these parameters individually. Thus, the user’s expertise is included in the visual representation to improve the process of visual analysis. In our example, the user could be interested to learn more about the vertical forces. These are displayed relatively rough by 3-4 petals. Now, there are two possibilities: first, the user can include the density function, which was the basis for the petal computation, to see more details. Second, he/she can change one or both parameters in order to compute a better suited rose diagram.

Fig. 2 shows in the bottom left-hand part possibilities how to adjust the rose diagram: Our running example is located at the bottom of the figure (initial petal width: 24, number of petals: 16). Based on these parameters, a decreased initial petal width leads to a more fine-granular appearance of the rose diagram (on the upper left). Whereas a decreased number of petals leads to melted petals if their differences in length are not too high (on the lower right).

As indicated, a novel sweep line algorithm has been developed to construct the diagram itself. Both parameters, initial petal width and number of petals, are input parameters of this algorithm. Basically, a delta region around an angle is observed that forms a *sweep sector*. The appearances of angles in this delta region are counted and represented as petals. Note that the entered number of petals is an absolute value, i.e., the resulting rose diagram has exactly the double number of petals, see below; whereas the entered petal width is only an initial value for the computation. The final result depends of the distribution of the force directions. Thin equispaced circles around the diagram center have been added for a better visual comparison of the petals lengths in the diagram itself and among different load steps/data sets.

In addition to the rose diagram, the Diagram View offers the possibility of showing a density representation, see Fig. 5 too. One can regard this *density diagram* as a rose diagram with an infinite number of petals. The density diagram (or density function) depends only on the delta region, used to count the contact angles. A wider delta angle will lead to a smoother representation, while a very small delta angle will lead into a very rough high-frequency signal. The choice of this angle is not obvious for different data sets. Therefore, the density diagram can be used to visually determine an appropriate initial petal width for rose diagrams. If we would use fixed angles or fixed frequency spectra, one could extract the border of the density diagram to perform shape comparisons.

The most relevant feature of the Microscale-Level View is that it supports straightforward interaction between the rose diagram representation and the *rve* within in Granular View, linking them together. Besides the ability of transforming the shape of the rose diagram by adjusting the scanning parameters (see above), it is possible to select parts of the diagram; a selected rose petal is highlighted in orange together with its mirror in the same rose diagram. Doing

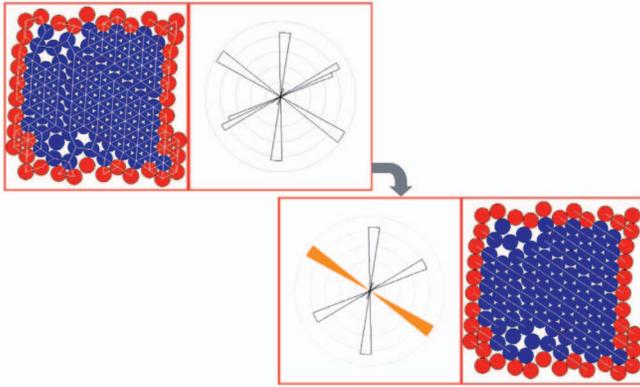


Fig. 6. The selected (orange) petal narrows the display of contact forces in the Granular View down. Contact forces are only displayed if their direction lie within both petal's legs. Standard mode was used because the magnitude of overlaps is very small in overall and that results in contrast problems in the overlap mode.

this, a user can specify an interesting force direction within the rose diagram, and all different directed contact forces are hidden in the Granular View. Fig. 6 gives an example of this process. This is a kind of filtering or brushing technique very similar to the *angular brushing* approach presented by Hauser et al. [17]. Their method allows the user to interactively specify a subset of slopes between two axes of parallel coordinates, which then yields all those data points to be marked as part of the current focus. The difference to our work is that we select angles given by the range that a specific petal provides. Thus, the selection of a rose petal lets the tool only display contacts inside the Granular View, which are collinear to the selected petal of the rose diagram. This filtering feature enables the user to directly observe the internal contact directions in the Granular View, cf., Fig. 2, bottom right-hand part. It can be utilized in any load step to reveal directional shifts.

3.4 Scalability

Our tool scales fine within the user requirements. This is woven into the normal usage regarding the interaction: it makes no sense to display 2,000 rose diagrams in the Microscale-Level View. However, 300-400 rose glyphs at the macro level (corresponds to 80-100 finite elements) are perceptible at once and can be used for further analysis, for focusing, and/or stepwise refinement of visualization parameters. For instance, our case study in Fig. 12 shows 160 rose glyphs. Domain experts discovered the four striking Gauss points without any problems. Of course, standard interaction techniques, such as zoom and pan, allow the user to handle larger finite element meshes. On the basis of the selected Gauss points, it is possible to analyze the underlying microstructures in more detail, as described in Section 5. Further research is needed to give a final answer concerning the scalability question. An interesting question is whether high-density tiled display technologies could improve this situation.

4 IMPLEMENTATION ASPECTS

This section discusses the algorithms needed for the new rose diagram construction. In Section 4.1, the general background and some definitions are given to support further explanations. A more detailed view on the algorithm itself can be found in Section 4.2.

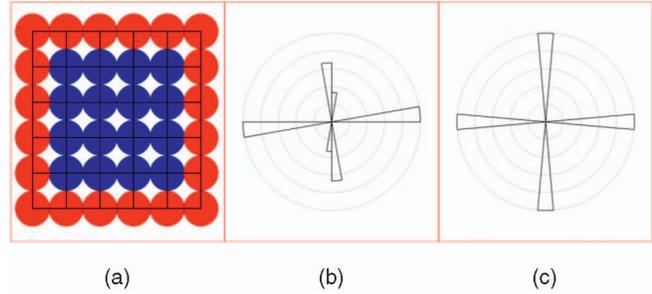


Fig. 7. (a) Particle assembly containing numerically small perturbations in vertical connections. (b) Rose diagram with visualization errors that is based on classical drawing methods. (c) Our new rose diagram drawing algorithm is able to handle such situations.

4.1 General Ideas

Key aspects of the presented visualization tool can be found in the capability to handle and visualize data sets resulting from multiscale computations. In general, these kinds of computations produce a large amount of structured data separated in multiple files. In our case, data are ordered by index triples that consist of macro load steps (mls), elements (elem), and Gauss points (gp). This leads to a specific number of files $\text{nof} = \text{mls} * \text{elem} * \text{gp}$. Thus, the file management results in an interesting challenge. Each file contains macroscopic data, e.g., stress tensors associated with the aforementioned index triple. Furthermore, each file contains data of the microscale level computation, e.g., particle positions partitioned by microscopic load steps. The number of microscopic load steps differs in each file depending on the applied deformation.

The amount of current data, as well as the intention of a later three-dimensional (3D) extension require an efficient programming language and an extendable visualization framework. In our case, we decided to use C++ [33], Qt [4], and OpenGL [38] for the implementation.

The key aspect of the visualization at the microscale level relates to a novel procedure for rose diagram construction. In the present case, data for a rose diagram arise from the unit contact normal vectors \mathbf{n}_{ij} . A unit contact normal vector \mathbf{n}_{ij} results from the normalized branch vector \mathbf{l}_{ij} , which connects the centers of the particles i and j (cf., Appendix A.2). Due to the ambiguousness of the unit contact normal vectors \mathbf{n}_{ij} and \mathbf{n}_{ji} , respectively, rose diagrams of granular media are intrinsically reflection symmetric. Please note that each unit contact normal vector can be interpreted as an angle inside a polar coordinate system.

Traditionally, the rose diagram construction follows a simple but error-prone procedure: First, a circular domain is partitioned into equal sized sectors. Then, the unit contact normal vectors are related to the appropriate circular sector by comparing the contact normal angle with the circular sectors. The sector radius is computed by counting the number of contact normal angles related to each sector. A normalization of sector radii closes the classical rose diagram construction, and the set of computed sectors yields the petals of the rose diagram. In particular cases, classical rose diagram construction leads to unexpected outcomes relating to numerical errors in the underlying data.

An example of such a misbehavior in the classical rose diagram construction is demonstrated in Fig. 7: This case shows a particle constellation in which the connections are orthogonally aligned (Fig. 7a). After the application of some load steps, the vertical connections become slightly perturbed

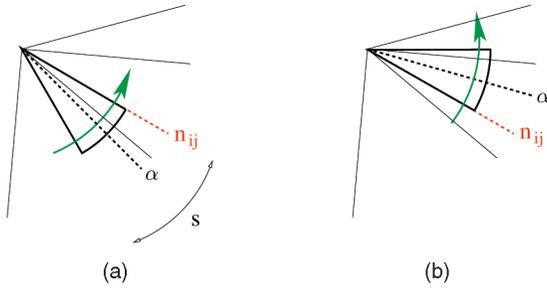


Fig. 8. Illustration of the inEvent and the outEvent. (a) shows the inEvent for a contact normal vector n_{ij} , marked by the red dashed line, with its corresponding angle α , marked by the black dashed line. (b) shows the outEvent to the same normal vector n_{ij} . The associated size of the sweep sector is defined by s .

about the vertical axis. As a result, these connections are perturbed over two neighboring sectors, which has a strong impact on the sector value distribution. Another problem in using classical rose diagrams is that the orientation of sectors and connections can visually differ (Fig. 7b). Our rose diagram construction prevents such errors by keeping the sector width variable (Fig. 7c): we convert the set of appearing contact normal angles into a function defined on the angular continuum. The specification of an user-defined region of influence s (defined by the initial petal width) leads to a definition of the directional density ρ :

$$\rho(\alpha) = \frac{1}{s} \sum_i \chi_{[a,b]}(\text{angle}(\mathbf{e}_1, \mathbf{n}_{ij})). \quad (1)$$

For an arbitrary angle α , the region of influence boundaries are defined as $a = \alpha - s/2$ and $b = \alpha + s/2$, respectively. They delimit the right open interval $[a, b)$. The angle between a direction n_{ij} and the unit vector e_1 influences the density function ρ if and only if α lies in the region between $k = \text{angle}(\mathbf{e}_1, \mathbf{n}_{ij}) - s/2$ and $l = \text{angle}(\mathbf{e}_1, \mathbf{n}_{ij}) + s/2$, respectively. Hence, the density function ρ is a piecewise constant function constructed by a sum of rectangle impulse functions. Each rectangle impulse function corresponding to a direction n_{ij} can be identified as set containment function χ with angle domain boundaries k and l .

4.2 Algorithm

The ideas discussed in the last section lead to a relatively simple algorithm containing different steps for each Gauss point. An overview of the developed algorithm is given in Algorithm 1.

Algorithm 1. Determination of rose diagrams.

```

for all FEM cells f do
  for all Gauss points g in f do
    particles = all particles in rve for g
    contacts = computeContacts(particles)
    densityfunc = computeDensity(particles, contacts)
    sectors = computeSectors(densityFunc)
    drawRose(sectors, GLYPH)
  if selectedForMicroView(g) then
    drawRose(sectors, DIAGRAM)
  end if
end for
end for

```

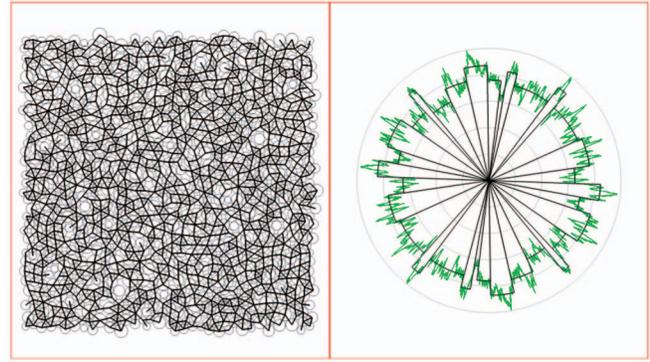


Fig. 9. Nearly *isotropic* particle distribution with resulting density function for an initial petal width of 5 degrees and clustering to 16 petals (32 petals displayed due to reflection symmetry).

The paradigm of sweep line algorithms is commonly used in computational geometry. Our *sweep sector algorithm* for the computation of the density function (see Algorithm 2) is based on collecting data related to the contact normal angles. Therefore, the sweep sector of size s is positioned twice for each normal direction, once in the case of an inEvent and once in the case of an outEvent. An inEvent, see Fig. 8a, is defined by positioning the left side of the rotating sweep sector onto the angle. In this case, the associated angle α_{in} is defined by $\alpha_{in} = k$. Whereas, outEvent is set by placing the right side of the sector onto the angle, see Fig. 8b. Thereby, the associated angle α_{out} yields $\alpha_{out} = l$. The number of angles, which correspond to n_{ij} and which are inside the sweep sector, is counted and stored. In the case of an inEvent, the number of directions in the sector gets increased, and in the case of an outEvent, it gets decreased alike. Processing all occurrences by sorting them in increasing order with respect to the associated angle results in the density function ρ . This density function is depicted green in Figs. 9 and 10 (see below for further explanations of these images).

Algorithm 2. computeDensity(particles, contacts).

```

E = empty event list
for all contacts c do
  inAngle = c.angle - s/2
  outAngle = c.angle + s/2
  insertEvent(E, inAngle, INEVENT)
  insertEvent(E, outAngle, OUTEVENT)
end for
sort E by increasing angles
while E not empty do
  e = top(E)
  begin new density function value d at e.angle
  if e.event == INEVENT then
    density++
  else if e.event == OUTEVENT
    density--
  end if
end while

```

Now, this definition of the density function ρ gives us the ability to generate a more sophisticated rose diagram. Algorithm 3 contains a pseudocode implementation of ρ . The main disadvantages in using the classical rose diagrams are the needs to choose a definite start angle and a constant sector size or sector number, respectively, cf., the discussion

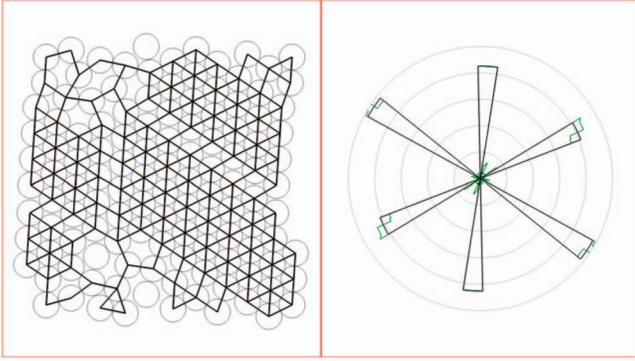


Fig. 10. Example of an *anisotropic* particle distribution with resulting density function for initial petal width of 10 degrees and clustering to three petals (six petals displayed due to reflection symmetry).

in Fig. 7. The corresponding rose diagram results now by using a clustering algorithm. In detail, all piecewise constant parts of the density function initialize a sector for the rose diagram. Then, the neighboring constant density function sectors, which contain the smallest variance, are combined until the interactively adaptable quantity of petals is reached. This procedure solves the discussed problems.

Algorithm 3. computeSectors(density function).

```
wanted = number of chosen sectors
S = empty sector queue
for all density function values d
  insertSector(S, d.angle, next(d).angle, d.density)
end for
while S.size > wanted do
  (sl, sr) = pair of sectors with minimal variance in S
  remove(S, sl)
  remove(S, sr)
  newdensity = avg(sl.density, sr.density)
  insertSector(S, sl.leftAngle, sr.rightAngle, newdensity)
end while
```

For a number of finite combinations, the proposed density function reaches a constant value and is represented by a single sector. In our case, a full distance matrix, resulting from the distance metrics of neighboring cluster sectors, is not necessary. This conclusion results from the fact that each sector has only two neighbors; formally, the distance to all other sectors is ∞ , and the neighboring information can be found and updated in $\mathcal{O}(1)$. Thus, the calculation of clustered rose diagrams is not limited by the number of particle contacts. As there are always two sectors that form one new clustered sector, the runtime and memory usage are both of order $\mathcal{O}(n)$.

Fig. 9 illustrates an example for a nearly *isotropic* case of particle distributions. The resulting density function for a chosen petal width of 5 degrees is shown in green color. This is clustered as far as there are 16 remaining petals. Due to the intrinsic reflection symmetry of the data, 32 petals are shown. Fig. 10 shows an *anisotropic* example. The user can clearly identify three different force directions. If the user wants to see, for example, a more fine-granular visualization, then he/she can change the initial petal width and/or the number of petals, as described in Section 3.3.2.

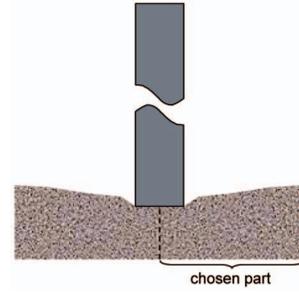


Fig. 11. Footing being pressed into a soil block. We assume that the footing is regular without any variances and that the soil is homogeneous. Under these prerequisites, we only regard one half of the soil block because of the reflection symmetry along the vertical axis through the middle of the footing.

We achieved a nearly perfect scalability for the calculation of rose diagrams. Linear computation time and storage requirements provide a solid foundation for handling large data sets. However, our data itself can get very extensive: Even for the relatively small data set of our case study in Section 5 with 16,000 rose diagrams over all load steps about 1 Gbyte of memory is needed. This results in a spatial limitation of finite element meshes and load steps to be visualized. However, this can be easily overwhelmed by implementing a strategy to only swap necessary data into the main memory. This will be done in a future version.

5 RESULTS

There are many possible case studies that can be presented in order to clarify usage and benefits of our approaches. A complex and practical application is, for example, a soil slippage on mountain slopes. This is a current problem in these times of global warming. At the moment, we cannot handle such a huge problem because of the aforementioned scalability reasons. A smaller standard example in geomechanical engineering is the examination of soil samples. A block of soil is clamped and standardized influences are simulated.

In this section, we discuss the application of our visualization tool to a well-known problem from the area of geomechanics. This problem is similar to the aforementioned example and consists of a footing being pressed into the soil ground, see Fig. 11. The *final deformed* structure is displayed in Fig. 12 in the form of a finite element mesh. Note that it shows the last load step of the deformation process, i.e., the deformation is finished, and no forces affects the medium because of Newton's second law. The overall soil at the macroscale level is discretized by $5 \times 8 = 40$ finite elements. Note that due to the reflection symmetry of the considered problem, only one half of the complete problem is modeled. Each finite element contains four Gauss points. Assuming a virtually rigid footing, we prescribe the deformation of three nodes, i.e., only vertical displacements are applied to these nodes. They are located near the upper left corner of the finite element mesh. These vertical displacements induce reaction forces at all bearings, but they are not depicted. The total deformation is applied in 100 load steps.



Fig. 12. Macroscale-Level View of the *final* deformed state. 40 finite elements are used to discretize one half of the footing problem. Red dots were manually added to this screen shot to point at the three nodes displaced by the footing. Boundary conditions are applied by displacements of the restricting bearings. Glyphs are visible and give a good overview of the underlying microstructure. All rves of the user selected and marked Gauss points are depicted in Fig. 13.

Each Gauss point is related to one rve at the microscale level. That leads to a total number of 160 rves. On the other hand, each rve contains 182 grains and is generated with the algorithm proposed in [24].

Glyphs in the Macroscale-Level View give a good overview of the related contact network inside the associated rve at the microscale level. Thus, identification and analysis of certain quantities of interest is done in a minimum of time. Let us consider our example in Fig. 12 in more detail: The first two left “columns” of finite elements appear relatively undistorted because of the bearings around this part, as well as of the vertical displacement in this area. Additionally, the user can watch that the most glyphs show more or less isotropies that corresponds to this observation. Four Gauss points of special interest are selected in the Macroscale-Level View according to their rose glyph representation. Surprisingly, the red marked Gauss point in our screen shot is not provided with a rose glyph because there are no overlapping particles. This is in contrast to the violet one that catches in the user’s eye because of its nearly isotropic shape. All of these selected Gauss points can be analyzed in parallel at the microscale level in further detail, see Fig. 13. The visual linkage between macro and microscale level is supported by using of corresponding color frames. Additionally, the capability of one tool to handle all the data resulting from multiscale computations leads to an enormous work reduction.

At the microscale level, our novel approach of visualizing contact network data yields to heavily improved insights into the force distribution between the particles compared to the standard software discussed in Section 2. These advantages are summarized in the following: Interaction features, relating the petals of the rose diagram to the corresponding contacts inside the rve, turn out to be a great help in analyzing the load carrying behavior of granular

materials, cf., Fig. 6. Furthermore, the novel formulation of the rose diagram excludes influences based on the numerical errors of the visualization data. We discuss the four striking Gauss points from the engineer’s perspective in clockwise order based on the frames in Fig. 13:

- In fact, the red framed diagram pair shows no connection forces between the particles. It follows that the soil has no stiffness in this area. We can carefully presume that the soil has failed (actual research question in the domain; probably a length information is needed to decide this), and this could result in an unstable footing. Without the help of our tool, the user would use several standard visualization tools, as described in Section 2. Most likely, he/she would not discover such interesting details because there are no visual and semantical mapping between them.
- The green pair shows a mainly horizontal force distribution. In contrast to the turquoise and the violet pair that present a more isotropic scenario, one can say that the soil can hold additional horizontal forces in this point. A closer look into the Granular View reveals interesting contact networks that are partly independent. Therefore, we have “chain-building” horizontal particle overlaps (clusters) that confirm our aforementioned assumption.
- A quite isotropic case is indicated by the violet-framed diagram pair. Nearly all directions occur in the Granular View and point to a pretty good stiffness of the soil in this area.
- By selection of a rose petal in the turquoise one, our visualization tool marks the selected petal in orange color and only displays the corresponding force direction in the particle assembly. In more detail, all other contact forces that are directed in other directions, e.g., from left to right, are hidden in Granular View. In such a way, the user can explore

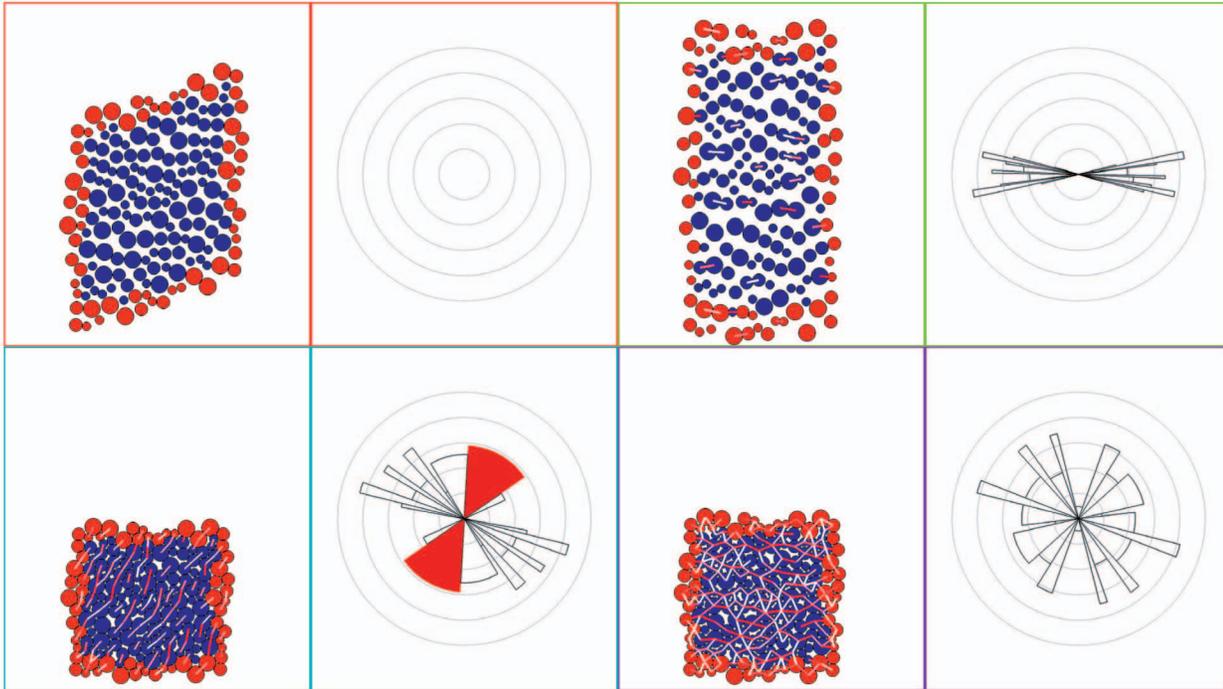


Fig. 13. Microscale-Level View of four selected rves. Each pair, i.e., a particle assembly and the corresponding rose diagram, is related to one Gauss point at the macroscale level by colored frames.

the assembly step-by-step, filter out uninteresting parts and discover possibly unexpected coherences.

That kind of visual analysis gives information about the granular material in itself. This is current (or even future) research in Geomechanics and of great importance for a better understanding of the behavior of powder, sand, etc. under external influences. The presented results are not only of interest when considering granular soils, but also applies to other kinds of granular materials. In this manuscript, we have documented one specific application of soil stability, which is of particular interest in civil engineering and geophysics. However, the approach is general enough to be appealing to other fields, dealing with granular media. In the process of structural design, it is of fundamental importance to know the forces distribution in granular assemblies. The unique visualization technique introduced in this manuscript provides further insight for civil, mechanical, or even chemical engineers. The present work ultimately supports the design of granular material related structures, e.g., storage containers for granular media such as hoppers. The visualization tools derived in this work allow a detailed understanding of complex force networks and local concentrations of forces that might eventually lead to structural failure.

6 CONCLUSION

We have presented a new system for interactive visualization of a two-scale homogenization process. The novel combination and connection of both views, namely, the Macroscale-Level View (*fem* level) and the Microscale-Level View (*dem* level), helped to improve research in the area of geomechanical engineering. Besides the implementation of standard visualizations, we have introduced a new class of rose diagrams for the purpose of visualizing force directions on

a microscale level. Interaction with these novel diagrams enables selective visualization of force direction patterns in granular media. This shows relevant contacts inside the load-carrying network, making our system an important new tool for further research in this area. The clustering algorithm used for the rose diagram generation takes advantage of the petal neighborhood structure, so that it becomes suitable for large data sets.

In future work, we plan to improve performance especially for the computation of the connectivity network for faster loading of data sets. Since parameter adjustments for density computations still have to be determined manually by the user, we also plan to provide an adaptive parameter adjustment. The display of stress plots that can be integrated in our *fem* mesh visualizations is an easy to implement but very useful feature; an example was shown in Fig. 1 of this article. A further issue is a better color coding of our contact networks within the Granular View if the user decides to choose the standard mode. The tool should also support an EPS export of selected visualizations. High-quality printouts are wished by domain specialists for presentations and journal publications. The quality of screen shots does not suffice for many occasions.

Another challenge will be the development of further methods to transfer our technique to 3D data. Currently, the underlying mechanics is based on several simplifications, for example, the limitation to 2D data sets (particles are circles). The next step would be to migrate to 3D. This will increase the simulation quality because real particles can have a complex spatial structure (different 3D shapes, not only spheres). From a visualization point of view, this will lead to a completely new definition of the Granular View and to the transformation of our rose diagrams into diagrams that are able to display 3D force directions. A

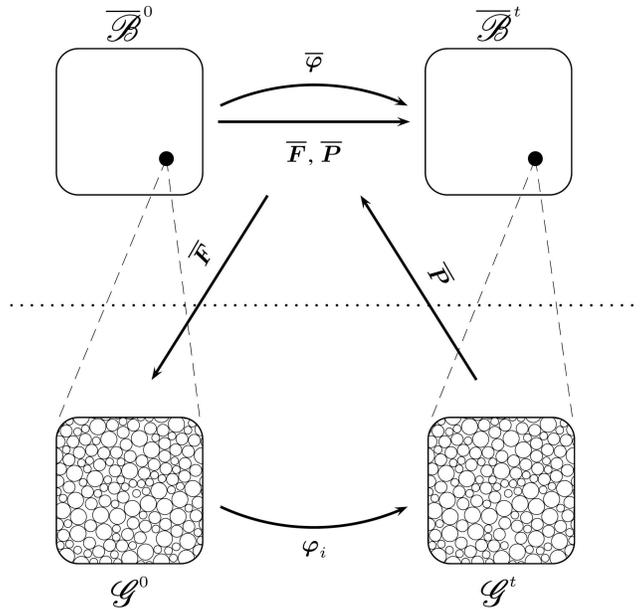


Fig. 14. Schematic illustration of the two-scale homogenization to clarify the most important functional connections. In the upper half, the macroscopic level is depicted, i.e., there are no discretizations. Each black dot represents any point, not necessarily a Gauss point. The lower half shows an abstraction of the microscopic level in which we have discrete media.

predictable problem will be the massive overlapping of particles and contact networks.

APPENDIX A

DOMAIN BACKGROUND

In the following appendix, we give a more detailed overview regarding the mechanical procedures of our two-scale homogenization approach. To provide the interested reader with a clearly arranged review, the complete homogenization cycle is depicted in Fig. 14. It shows a schematic illustration of the two-scale homogenization: The upper left quarter shows the macroscopic body in the reference configuration $\bar{\mathcal{B}}^0$ at time zero. The macroscopic body in the current configuration $\bar{\mathcal{B}}^t$ is depicted in the upper right quarter. Mapping between the reference and current configuration is accomplished by the macroscopic nonlinear deformation map $\bar{\varphi}$. The granular microstructure \mathcal{G}^0 at time zero is found in the lower left quarter, whereas the current micro structure \mathcal{G}^t at time t is found in the lower right corner.² The mapping between the configurations of the granular microstructure is based on the microscopic nonlinear deformation map φ_i . Applying the macroscopic deformation gradient tensor $\bar{\mathbf{F}}$ to the granular micro structure results in a stress response $\bar{\mathbf{P}}$, denoting the Piola stress.

A.1 Two-Scale Homogenization Approach

Modeling and simulation of confined granular medium includes the description of the behavior of single grains and grain groups. The advantages of homogenization schemes

2. Note that both microstructures can be equal after the mapping with φ_i . The “particles” in the lower half in Fig. 14 are symbolic, meant in order to give a hint to a discrete medium.

are found in the capability to project microscopic quantities onto macroscopic behaviors: In our case, the microscopic level shows the “true structure” of the granular media \mathcal{G} , consisting of an arbitrary arrangement of grains, see Fig. 14 (lower left). Such approaches allow to convey the distinction of dry granular matter, i.e., the ability to form and break contacts between the single grains. This feature is of great importance since forces inside granular media are solely transmitted at contact points. In virtue of arbitrary arrangements of the grains inside, quantities inside the granular assembly vary strongly over the domain of interest. Nevertheless, the mechanical power at the macroscale level has to be equivalent to the volume averaged mechanical power at the microscale level [18]:

$$\bar{\mathbf{P}} : \dot{\bar{\mathbf{F}}} = \langle \mathbf{f}_{ij} \cdot \dot{\mathbf{l}}_{ij} \rangle. \quad (2)$$

Therein, quantities holding an over bar are associated with the macroscopic scale, whereas quantities without are related to the microscale level. Both, the Piola stress tensor $\bar{\mathbf{P}}$, see [23], as well as the material velocity gradient tensor $\bar{\mathbf{F}}$ are considered to be two field tensors, i.e., standing between the initial and the current configuration, see Fig. 14. The branch vector \mathbf{l}_{ij} between two particles i and j is defined as the vector connecting the particle centers. \mathbf{f}_{ij} represents the contact force between the particles i and j . Superimposed dots on quantities point out the derivative with respect to time. For the sake of simplicity, we will omit the definition of the volume average procedure, denoted by $\langle \bullet \rangle$, and refer the interested reader to the publications [18], [25], [39]. Using (2), fluctuations at the microscale level are smeared over the microscopic domain. The Hill theorem builds the fundament of most homogenization procedures and is used to link averaged microscopic to macroscopic quantities. Additionally, this theorem leads to a choice of an appropriate *representative volume element (rve)*.

A geometric periodic rve is a periodic sample of the material of interest. If it is large enough to capture all effects of the true material and small enough to be representative for the entire microstructure of the domain of interest at the microscale level, then this rve is laid out. Note that we assume a geometric periodicity with the period equal to the length of the rve itself, see Fig. 15. Furthermore, we demand a length at the microscale level to be significantly smaller than a length at the macroscale level, i.e., $l \ll L$.

An effective algorithm to produce ab initio geometric periodic rves has been introduced in [24]. This algorithm allows the generation of rves by a given grain size distribution.

At the macroscale level, only the homogenized, i.e., averaged effective material response is considered. Therefore, we can assume the material at the macroscale level to be homogeneous, allowing the application of a continuum approach. In what follows, boundary conditions at the microscale level are solely applied according to the Taylor model, i.e., we assume all particle fluctuations to be equal to zero.

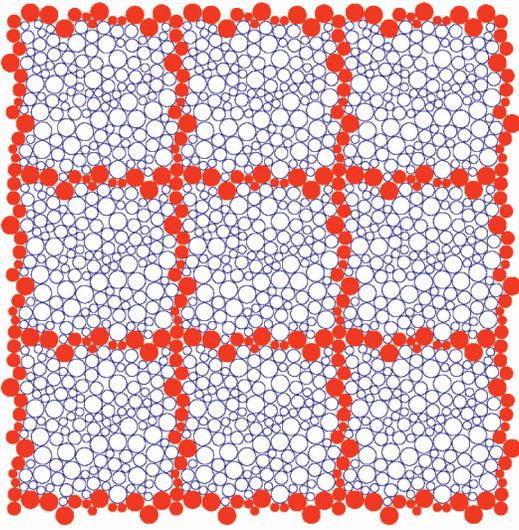


Fig. 15. Granular assembly showing nine geometric periodic cells. The period is measured between the centers of the red particles. These generate the boundary of the geometric periodic rve and belong to the boundary particle set, where as the particles in the inner are associated with the inner particle set. Note that the choice of the geometric periodic rve is not unique.

A.2 Microscale-Level Computation

The mechanics at the microscale level of our homogenization approach are based on the *dem*, introduced in [10]. The authors presented an explicit discrete numerical method to analyze the behavior of granular media. The *dem* allows the simulation of the behavior of granular matter in a natural way, yielding equivalent results if compared to experimental data [9], [11]. Complicated behaviors such as continuously nonlinear stress/strain response, dilation related to mean stress, transition from brittle to ductile behavior, hysteresis and memory, and breaking and forming of contacts between the single grains automatically appear from the *dem* [8].

Note that without a loss of generality with regards to the visualization, we restrict this contribution to smooth particles. Thus, a tangential part of the contact forces is disregarded. Additionally, we assume a dry particle assembly and exclude any kind of attraction forces.

In the case of dry granular media, forces inside the granular assembly solely transmit at contact points. Contact forces between particles depend on a penalty force approach. The magnitude of the normal contact penalty forces relates to the overlap between the particles, multiplied by a stiffness coefficient. The overlap, $\varepsilon_{ij} = \|\mathbf{l}_{ij}\| - [r_i + r_j]$, between two particles computes by subtracting the particle radii from the length of the branch vector. For the distance between the grains being greater than the sum of the particle radii, i.e., $\varepsilon_{ij} > 0$, the magnitude of the normal penalty force is set to zero. In the case of contact, i.e., the distance between the particles is equal to or less than the sum of their radii, the magnitude of the contact force is calculated by a potential energy function ψ_n , which depends on the particle overlap. The normal contact force that is unequal to zero for $\varepsilon_{ij} < 0$ is thermodynamically conjugated to the particle overlap ε_{ij} :

$$\mathbf{f}_{n,ij} = \psi'_n(\varepsilon_{ij})\mathbf{n}_{ij}. \quad (3)$$

Here, \mathbf{n}_{ij} represents the contact normal related to the branch vector, $\mathbf{n}_{ij} = \mathbf{l}_{ij}/\|\mathbf{l}_{ij}\|$. The prime on the energy function ψ_n denotes the derivative with respect to the overlap of the particles i and j .

Each single particle is mapped by the macroscopic deformation gradient tensor $\overline{\mathbf{F}}$. Thus, the position of particle i at time $n + 1$ is prescribed by

$$\mathbf{x}_i^{n+1} = \overline{\mathbf{F}} \cdot \mathbf{x}_i^0. \quad (4)$$

The macroscopic deformation gradient tensor $\overline{\mathbf{F}}$ is understood as a linear map between the initial particle position \mathbf{x}_i^0 and the current particle position \mathbf{x}_i^{n+1} . Additionally, (4) identifies the macroscopic deformation gradient tensor as the driving quantity of the whole homogenization process. To complete the homogenization cycle, an averaged stress quantity is returned to the macroscopic level, as shown in Fig. 14. With regards to the mechanical power at the macroscale level, the macroscopic Piola stress tensor $\overline{\mathbf{P}}$ is selected. The macroscopic Piola stress computes to

$$\overline{\mathbf{P}} = \left\langle \mathbf{f}_{ij}^{n+1} \otimes \mathbf{l}_{ij}^0 \right\rangle. \quad (5)$$

Its determination completes the computation at the microscale level.

ACKNOWLEDGMENTS

The authors thank the German Research Foundation (DFG) for the financial support within the DFG International Research Training Group 1131, "Visualization of Large and Unstructured Data Sets—Applications in Geospatial Planning, Modeling, and Engineering" at the University of Kaiserslautern, Germany. They also like to acknowledge Burkhard Lehner, Computer Graphics and Visualization Group, University of Kaiserslautern, for their support and Daniel Spikol, CeLeKT Group, Växjö University, for carefully proof reading this paper.

REFERENCES

- [1] Abaqus FEA, <http://www.simulia.com/>, 2008.
- [2] Ansys, <http://www.ansys.com/>, 2008.
- [3] A. Barr, "Superquadrics and Angle-Preserving Transformations," *IEEE Computer Graphics and Applications*, vol. 1, no. 1, pp. 11-23, 1981.
- [4] J. Blanchette and M. Summerfield, *C++ GUI Programming with Qt 4*. Prentice Hall, 2006.
- [5] L. Brasseur, "Florence Nightingale's Visual Rhetoric in the Rose Diagrams," *Technical Comm. Quarterly*, vol. 14, no. 2, pp. 161-182, 2005.
- [6] M.C. Chuah and S.G. Eick, "Managing Software with New Visual Representations," *Proc. IEEE Symp. Information Visualization (InfoVis '97)*, p. 30, 1997.
- [7] M.C. Chuah and S.G. Eick, "Information Rich Glyphs for Software Management Data," *IEEE Computer Graphics and Applications*, vol. 18, no. 4, pp. 24-29, July/Aug. 1998.
- [8] P.A. Cundall, "A Discontinuous Future for Numerical Modeling in Soil and Rock," *Proc. Third Int'l Conf. Discrete Element Methods (DEM '02) Discrete Element Methods, Numerical Modeling of Discontinua*, B.K. Cook and R.P. Jensen, eds., pp. 3-4, 2002.
- [9] P.A. Cundall and O.D.L. Strack, *The Distinct Element Method as a Tool for Research in Granular Media*, Report to the Nat'l Science Foundation Concerning NSF Grant ENG76-20711, Part 1, 1978.
- [10] P.A. Cundall and O.D.L. Strack, "A Discrete Numerical Model for Granular Assemblies," *Géotechnique*, vol. 29, no. 1, pp. 47-65, 1979.

- [11] P.A. Cundall and O.D.L. Strack, *The Distinct Element Method as a Tool for Research in Granular Media*, Report to the Nat'l Science Foundation Concerning NSF Grant ENG76-20711, 7, 1979.
- [12] W. Day and H. Edelsbrunner, "Efficient Algorithms for Agglomerative Hierarchical Clustering Methods," *J. Classification*, vol. 1, no. 1, pp. 7-24, 1984.
- [13] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf, *Computational Geometry: Algorithms and Applications*. Springer, Jan. 2000.
- [14] D. Defays, "An Efficient Algorithm for a Complete Link Method," *The Computer J.*, vol. 20, no. 4, pp. 364-366, 1977.
- [15] *Dem Solutions*, <http://www.dem-solutions.com/demsolutions.html>, 2008.
- [16] S. Fortune, "A Sweepline Algorithm for Voronoi Diagrams," *Algorithmica*, vol. 2, pp. 153-174, 1987.
- [17] H. Hauser, F. Ledermann, and H. Doleisch, "Angular Brushing of Extended Parallel Coordinates," *Proc. IEEE Symp. Information Visualization (InfoVis '02)*, p. 127, 2002.
- [18] R. Hill, "On Constitutive Macro-Variables for Heterogeneous Solids at Finite Strain," *Proc. Royal Soc. London A*, vol. 326, pp. 131-147, 1972.
- [19] K.H.H. Huebner, D.L. Dewhurst, D.E. Smith, and T.G. Byrom, *Finite Element Method*. John Wiley & Sons, 2001.
- [20] *Itasca Consultants GMBH*, <http://www.itasca.de/>, 2008.
- [21] P. Jager, S.M. Schmalholz, D.W. Schmid, and E. Kuhl, *Brittle Fracture During Folding Rocks—A Finite Element Study*, submitted for publication, 2008.
- [22] B. Lipchak and M. Ward, "Visualization of Cyclic Multivariate Data," *Visualization 97 Late Breaking Hot Topics*. pp. 61-64, IEEE CS Press, 1997.
- [23] L.E. Malvern, *Introduction to the Mechanics of a Continuous Medium*. Prentice Hall, 1969.
- [24] H.A. Meier, E. Kuhl, and P. Steinmann, "A Note on the Generation of Periodic Granular Microstructures Based on Grain Size Distributions," *Int'l J. Numerical and Analytical Methods in Geomechanics*, vol. 32, no. 5, pp. 509-522, 2008.
- [25] S. Nemat-Nasser, "Averaging Theorems in Finite Deformation Plasticity," *Mechanics of Materials*, vol. 31, pp. 493-523, 1999.
- [26] F. Nightingale, *Notes on Matters Affecting the Health, Efficiency, and Hospital Administration of the British Army*. Harrison & Sons, 1858.
- [27] D.H. Norrie and G. De Vries, *An Introduction to Finite Element Analysis*. Academic Press, 1978.
- [28] W. Ribarsky, E. Ayers, J. Eble, and S. Mukherjee, "Glyphmaker: Creating Customized Visualizations of Complex Data," *Computer*, vol. 27, no. 7, pp. 57-64, July 1994.
- [29] J.C. Roberts, "Exploratory Visualization with Multiple Linked Views," *Exploring Geovisualization*, A. MacEachren, M.-J. Kraak, and J. Dykes, eds., Elsevier, Dec. 2004.
- [30] C.D. Shaw, J.A. Hall, C. Blahut, D.S. Ebert, and D.A. Roberts, "Using Shape to Visualize Multivariate Data," *Proc. Workshop New Paradigms in Information Visualization and Manipulation (NPIVM '99)*, pp. 17-20, 1999.
- [31] L.K. Stewart, K.J. Woolfe, and D.P. Zwartzb, "A New Tool for the Integration, Graphical Presentation and Comparison of Files Containing Palaeocurrent Data," *Computers & Geosciences*, vol. 27, no. 3, pp. 351-355, 2001.
- [32] G. Strang and G.J. Fix, *An Analysis of the Finite Element Method*. Prentice Hall, 1973.
- [33] B. Stroustrup, *The C++ Programming Language*, special third ed. Addison-Wesley Longman, 2000.
- [34] M.O. Ward, "Xmdvtool: Integrating Multiple Methods for Visualizing Multivariate Data," *Proc. Conf. Visualization (VIS '94)*, pp. 326-333, 1994.
- [35] M.O. Ward, "A Taxonomy of Glyph Placement Strategies for Multidimensional Data Visualization," *Information Visualization*, vol. 1, pp. 194-210, 2002.
- [36] J.H. Ward, Jr., "Hierarchical Grouping to Optimize an Objective Function," *J. Am. Statistical Assoc.*, vol. 58, pp. 236-244, 1963.
- [37] N.A. Wells, "Are There Better Alternatives to Standard Rose Diagrams," *J. Sedimentary Research*, vol. 70, no. 1, pp. 37-46, 2000.
- [38] R.S. Wright, B. Lipchak, and N. Haemel, *OpenGL SuperBible: Comprehensive Tutorial and Reference*, fourth ed. Addison-Wesley Professional, 2007.
- [39] T.I. Zohdi and P. Wriggers, *Introduction to Computational Micromechanics*. Springer, 2005.



Holger A. Meier received the MSc (Dipl.-Ing.) degree from the Ruhr-University Bochum, Germany, in December 2004. In January 2005, he joined the International Research Training Group, "Visualization of Large and Unstructured Data Sets Applications in Geospatial Planning, Modeling, and Engineering," Kaiserslautern, Germany. His current research interests consist of the computational homogenization of particulate media and its visualization.



Michael Schlemmer received the MS degree in computer science from the University of Kaiserslautern, Germany, in 2004. He was a DFG IRTG 1131 fellow from 2005 until 2008 and is currently completing his PhD thesis. His main research interests include scientific visualization, computer vision, and pattern recognition.



Christian Wagner received the MS degree from the Technical University of Kaiserslautern, Germany. He is currently working toward the PhD degree in the International Research Training Group IRTG 1131, University of Kaiserslautern, funded by the German Research Association DFG. His research interests include visualization and visual analytics.



Andreas Kerren received the BS, MS, and PhD degrees from Saarland University, Saarbrücken, Germany. He is currently an associate professor in computer science at the School of Mathematics and Systems Engineering, Växjö University, Sweden, where he is the head of the research group for Information and Software Visualization (ISOVIS). He was and is a member of several program and organizing committees of international conferences and workshops and has served as a reviewer for several international journals. His main research interests include software visualization, information visualization, visual analytics, software engineering, computer science education, and human-computer interaction. He is a member of the IEEE.



Hans Hagen received the BS and MS degrees from the University of Freiburg and the PhD degree from the University of Dortmund. He is currently a professor in the Department of Computer Sciences, University of Kaiserslautern, Germany, and is heading the research group for the Computer Graphics and Computer Geometry. He is both a national and international pioneer in his research domains of geometric modeling and scientific visualization. Additionally, his scientific interests cover the domains of computer graphics and computer aided geometric design. He received several offers, among them are dean and chairman positions at prominent American universities. His scientific work covers more than 250 refereed publications and 15 books. He is a fellow of the IEEE and the IEEE Computer Society.



Ellen Kuhl received the diploma degree from the University of Hannover, the PhD degree from the University of Stuttgart, and the habilitation from the University of Kaiserslautern, Germany, where she then held an assistant professor position for four years. She is currently working in the Departments of Mechanical Engineering and Bioengineering, Stanford University. Her research interests include the continuum and computational modeling of biomechanical growth and remodeling phenomena and the computationally guided design of novel stem-cell-based therapies for myocardial infarction.



Paul Steinmann received the diploma degree and the habilitation at the University of Hannover, Germany, and the PhD degree from the University of Karlsruhe. Currently, he is a full professor and the chair of applied mechanics at the Department of Mechanical Engineering, University Erlangen/Nuremberg, since 2007. From 1997 to 2007, he was appointed the same position at the University of Kaiserslautern. His main research interests include continuum mechanics and computational mechanics.

▷ **For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/publications/dlib.**