Gesellschaft für Informatik (GI)

publishes this series in order to make available to a broad public recent findings in informatics (i.e. computer science and information systems), to document conferences that are organized in cooperation with GI and to publish the annual GI Award dissertation.

Broken down into the fields of
- Seminar
- Proceedings
- Dissertations
- Thematics

current topics are dealt with from the fields of research and development, teaching and further training in theory and practice. The Editorial Committee uses an intensive review process in order to ensure the high level of the contributions.

The volumes are published in German or English

Information: http://www.gi-ev.de/service/publikationen/lni/

ISSN 1614-3213
ISBN 978-3-88579-441-7

This volume contains papers from the second annual workshop of the DFG´s International Research Training Group "Visualization of Large and Unstructures Data Sets – Applications in Geospatial Planning, Modeling, and Engineering" held at Kaiserslautern in September 2007.

The topics covered in the papers range from fundamental visualization and interaction techniques to visualizations of application data from various domains, such as Geospatial Planning, Modeling, and Engineering.

Hans Hagen, Martin Hering-Bertram, Christoph Garth (Eds.):
Visualization of Large and Unstructured Data Sets

7

GI-Edition

Lecture Notes in Informatics

Hans Hagen, Martin Hering-Bertram, Christoph Garth (Eds.)

Visualization of Large and Unstructured Data Sets

Second Workshop of the DFG´s International Research Training Group "Visualization of Large and Unstructured Data Sets – Applications in Geospatial Planning, Modeling, and Engineering"

September 9–11, 2007, Kaiserslautern, Germany

Seminars

Hans Hagen, Martin Hering-Bertram, Christoph Garth (Eds.)

Visualization of Large and Unstructured Data Sets

Second workshop of the DFG's International Research Training Group

Visualization of Large and Unstructured Data Sets – Applications in Geospatial Planning, Modeling, and Engineering

September 9-11, 2007 Kaiserslautern, Germany

Gesellschaft für Informatik e.V. (GI)

Lecture Notes in Informatics (LNI) – Seminars
Series of the Gesellschaft für Informatik (GI)

Volume S-7
ISBN 978-3-88579-441-7
ISSN 1614-3213

Volume Editors
Prof. Dr. Hans Hagen
Department of Computer Science, University of Kaiserslautern
P.O. Box 3049, D-67653 Kaiserslautern, Germany
email: hagen@informatik.uni-kl.de
Dr. Martin Hering-Bertram
Department of Computer Science, University of Kaiserslautern
P.O. Box 3049, D-67653 Kaiserslautern, Germany
email: bertram@informatik.uni-kl.de
Dr. Christoph Garth
Institute for Data Analysis and Visualization, University of California, Davis
One Shields Ave., Davis, CA 95618, USA
email: cgarth@ucdavis.edu

printed by Köllen Druck+Verlag GmbH, Bonn

Preface

The International Research Training Group (IRTG) "Visualization of Large and Unstructured Data Sets – Applications in Geospatial Planning, Modeling and Engineering" is a joint effort of the University of Kaiserslautern (Germany) and the U.S. partners University of California at its Davis and Irvine campuses, Arizona State University, and University of Utah. It is funded by the German Science Foundation (DFG) under grant DFG GK 1131.

The primary research goal of this graduate program is the enhancement of scientific and information visualization techniques applied to large and unstructured data sets. Every visualization task is based on application data. For providing these data, we integrate applications from the domain "Geospatial Planning, Modeling and Engineering", which produce huge amounts of unstructured data that are of interest for the visualization tasks at hand. This integration is necessary to allow a deeper understanding of the provided data due to the sharing of knowledge through the projects.

Up to now, visualization of large and structured or small and unstructured data sets is the state of the art. Large and unstructured data sets are still not very well understood, especially with respect to visualization. In order to address these questions, we have defined a set of projects aiming at solving these problems. In detail, we are handling visualization problems, with respect to modeling, feature detection, and comparison tasks. For doing this, both the extension of existing techniques and the development of new ones are investigated.

In the application areas there is an increasing need to handle huge amounts of unstructured data that are produced either by data from field measurements like environmental observation stations, from experiments, and from simulation. For example, nowadays environmental monitoring systems are capable if measuring data at a very high resolution and in a large number of frequency bands. On the other hand, in scaled-down earthquake laboratory experiments within a centrifuge improved sensor technology permits the measurement of an increased number of parameters at higher sampling rates. Finally, earthquake simulations produce more and more data because of more elaborate simulation techniques. All these improvements in measurement technology lead to large, high-dimensional data sets. Visualizing these data is very useful to get new insights into the problems involved. The visualizations themselves are based on improved or newly developed visualization techniques like volume modeling, feature detection and visualization, etc.

The current issue of GI's Lecture Notes in Informatics presents the results of the second annual workshop of this IRTG held in Kaiserslautern, September 9-11, 2007. The aim of this meeting was to bring together all project partners, advisors, and of course PhD students as well as to report on the different research projects. After three days of presentations and lively discussions, slightly more than three months were spent on writing papers that cover the outcome of the second year of the graduate program and give surveys on related topics. These papers were cross-reviewer internally as well as by the project advisors. Note that the covered topics do not include all ongoing projects because not all IRTG members were able to attend.

We would like to thank all attendees for their contribution to this fruitful workshop. We are also grateful to the Gesellschaft for Informatik e.V. (GI) for publishing the workshop papers in its LNI series.

August 2008

Hans Hagen
Martin Hering-Bertram
Christoph Garth

Workshop Organizers

Inga Scheler
University of Kaiserslautern
Daniel Steffen
University of Kaiserslautern
Torsten Bierz
University of Kaiserslautern

Principal Investigators of the IRTG[1]

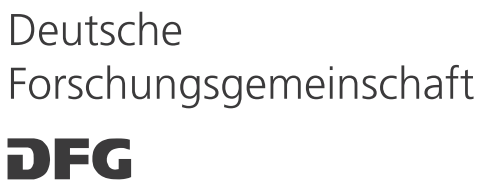
Prof. Dr. Hans Hagen (Speaker and Overall Principal Investigator)
Department of Computer Science
University of Kaiserslautern
P.O. Box 3049
D-67653 Kaiserslautern
email: hagen@informatik.uni-kl.de

Prof. Dr. Bernd Hamann (Speaker and Co-Principal Investigator)
Institute for Data Analysis and Visualization
Department of Computer Science
University of California, Davis
One Shields Avenue
Davis, CA 95616-8562, USA
email: bhamann@ucdavis.edu

Prof. Dr. Jan Aurich (Co-Principal Investigator)
Department of Mechanical and Process Engineering
University of Kaiserslautern
P.O. Box 3049
D-67653 Kaiserslautern
email: aurich@cck.uni-kl.de

[1] http://www.irtg.uni-kl.de

Sponsor



Partner Universities

Contents

On the modification of phonon tracing

Eduard Deines and Frank Michel
International Research Training Group
University of Kaiserslautern
{deines,michel}@informatik.uni-kl.de

Abstract: Phonon tracing is a geometric approach estimating acoustics in closed rooms. This work contains discussions of possible refinements and extensions of the algorithm.
In computer graphics exist numerous level-of-detail approaches decreasing the representation details of objects in order to speed up computations and rendering of virtual scenes. Different ideas for the realization of a level-of-detail approach in acoustics are presented. For this purpose the phonon tracing algorithm has to be modified. With these modifications the room impulse response can be calculated with respect to the user requirements on calculation time and accuracy.

1 Introduction

Most methods applied for acoustic simulation of enclosures are based on geometric acoustics [AB79, Bor84, Kro68, Kul84, Vor89, FCE+98, KJM04, Sve02]. Prevalent, the sound energy or intensity transported from the sound source to the listener is determined. In our previous work [BDM+05] we presented the phonon tracing algorithm, an energy based approach for room acoustic simulation. The present paper covers several modifications and extensions of the phonon tracing algorithm.

In contrast to sound pressure the sound energy quantities can hold only positive values. Thus modeling of interference phenomena is not possible. Therefore we modify the phonon tracing algorithm such that it traces pressure instead of energy facilitating the representation of interference.

If the structure of a surface is of the same order as the wave length, diffuse reflection of the sound wave front occurs. We extend the tracing step of the phonon tracing algorithm considering diffuse reflection depending on given scattering coefficients.

Furthermore we describe a modification of the collection step of the phonon tracing algorithm in order to allow an adaptive collection of the sound particles. The idea is to adjust the level of detail of the calculated room impulse response to the user requirements on the calculation accuracy and time. Therefore, we subdivide the particles stored in the phonon map into clusters of similar trajectories. Depending on the user's input parameters a subset of the phonons is collected determining the impulse response.

The remainder of our paper is structured as follows. In the next section we will describe

the modified phonon tracing algorithm which calculates the pressure response instead of energy. In section 3 we describe the integration of diffuse reflection into the tracing step. Afterwards, in section 4 we present a novel level of detail approach to phonon tracing. Finally we will conclude our paper with a discussion of future work.

2 Tracing the sound pressure

This section describes the modification of the phonon tracing approach, where sound pressure is used for calculations, instead of energy. The main idea is, analogous to the algorithm described in [BDM+05], to trace sound particles outgoing from the sound source through the given scene constructing the phonon map. Afterwards the phonons are collected in order to calculate the room impulse response at a given listener position. In contrast to the energy approach where quadratic attenuation is used because of the spatial particle density, linear pressure attenuation is facilitated. This is obtained by modeling linear pressure attenuation by Gaussian basis functions dilated proportional to the traversed distance (see figure 1). Here, single particles can be considered as individual micro-sources in analogy to the image-source method.

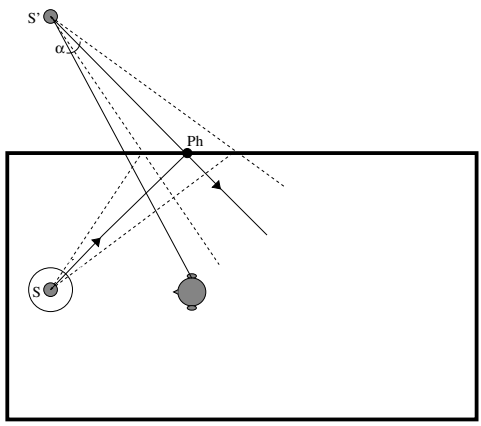


Figure 1: Sketch of the phonon tracing algorithm to trace sound pressure inside closed rooms.

2.1 Algorithm specification

We define:

- S a spherical sound source in point x_0 in $\mathbb{R}^3$
- S_s a unit sphere in $\mathbb{R}^3$, i.e. $S_s = \{x \in \mathbb{R}^3; \|x\| = 1\}$

- $\{v_i\}_{i=1,\dots,N}$ a set of directions distributed on S_s with regard to the emission distribution of the sound source S
- $\psi_i : S_s \mapsto \mathbb{R}, x \mapsto g(\arccos(\langle x, v_i \rangle))$, $i = 1, ..., N$ the partition of unity over S_s

For a unit impulse (Dirac) δ at time delay $\Delta t = \frac{1}{f_s}$ the pressure $P(t, x)$ at time $t \in \mathbb{R}$ in x reads:

$$P(x,t) = P_1 \frac{1}{\|x\|} \delta\left(t - \frac{\|x\|}{c}\right) \sum_{i=1}^{N} \psi_i\left(\frac{x}{\|x\|}\right) \tag{1}$$

where P_1 is the pressure in 1 m distance from the sound source S, c is the sound velocity, f_s the sampling rate, and $\delta(t)$:

$$\delta(t) = \begin{cases} 1 & \text{if } t \in [0, \Delta t] \\ 0 & \text{else} \end{cases} \tag{2}$$

As mentioned before we choose g as Gaussian basis function of an angle $\vartheta = \arccos(\langle x, v_i \rangle)$, $i = 1, ..., N$:

$$g(\vartheta) = \alpha \cdot \exp\left(-\frac{\vartheta^2}{2\sigma^2}\right) \tag{3}$$

For evaluation of the room acoustics a uniform distribution of sound rays is used. In order to achieve the partition of unity of the basis functions for the uniform distribution (the needed parameter can also be derived for non uniform distributed direction vectors), it must hold:

$$\int_{S_s} g \, \mathrm{d}S = \frac{4\pi}{N} \;\Leftrightarrow\; \alpha = \frac{2}{N} \left(\int_{\vartheta=0}^{\pi} \exp\left(-\frac{\vartheta^2}{2\sigma^2}\right) \cdot \sin\vartheta \, \mathrm{d}\vartheta \right)^{-1} \tag{4}$$

where 4π is the surface area of the unit sphere S and N the number of direction vectors. The integral in the above equation has to be solved numerically, but if we assume $\sigma \ll 1$, then it holds:

$$\int_{\vartheta=0}^{\pi} \exp\left(-\frac{\vartheta^2}{2\sigma^2}\right) \cdot \sin(\vartheta) \, \mathrm{d}\vartheta \approx \int_{\vartheta=0}^{\pi} \exp\left(-\frac{\vartheta^2}{2\sigma^2}\right) \, \mathrm{d}\vartheta \tag{5}$$

By substituting $u = -\frac{\vartheta^2}{2\sigma^2}$ and $\mathrm{d}u = -\frac{1}{\sigma^2}\vartheta \, \mathrm{d}\vartheta$ we get:

$$\int_{\vartheta=0}^{\pi} \exp\left(-\frac{\vartheta^2}{2\sigma^2}\right) \, \mathrm{d}\vartheta \approx \sigma^2 \tag{6}$$

This leads to:

$$\alpha = \frac{2}{N\sigma^2} \tag{7}$$

Choosing σ there is a trade-off between smoothness of the partition of unity and the resolution of geometric details of the scene. For better approximation of the partition of unity the support of the Gaussian function g should contain n different direction vectors. The surface area for a direction reads:

$$a = n\frac{4\pi}{N} \tag{8}$$

Furthermore to fulfill the partition of unity the following must be true for an angle ϑ_0:

$$\exp\left(-\frac{\vartheta_0^2}{2\sigma^2}\right) \geqslant \frac{1}{2} \;\Rightarrow\; \sigma = \frac{\arccos\left(1-\frac{2n}{N}\right)}{\sqrt{2\ln 2}} \tag{9}$$

Thus, the Gaussian basis function with an approximated partition of unity over the unit sphere S reads:

$$g(\vartheta) = \frac{2}{N\sigma^2} \cdot \exp\left(-\frac{\vartheta^2}{2\sigma^2}\right) \tag{10}$$

Halton sequence. For the accuracy of the Monte-Carlo integration, the uniformity is more important than the randomness. The Halton quasi-random sequence (also applied in computer graphics [Kel96a, Kel96b]) is used for generating direction vectors. The Halton sequence is obtained by calculating the radical inverse function, which takes a number $i \in \mathbb{N}$ represented in prime base p and reflects it through the radical point:

$$\Phi_p(i) := \sum_{j=0}^{\infty} a_j(i) p^{-j-1} \in [0,1) \Leftrightarrow i = \sum_{j=0}^{\infty} a_j(i) p^j \tag{11}$$

The m dimensional Halton sequence reads:

$$z_i = (\Phi_{p_1}(i), ..., \Phi_{p_m}(i)) \tag{12}$$

The radical inverse function can be intutively understood in the following way. To get the radical inverse of i in base p, we write down the digits of i in the base p, then we invert their order and place the floating point in the beginning. The resulting number is a desired floating point number, expressed in base p. The Halton sequence is incremental, i.e. it is possible to increase the number of samples without discarding the already calculated samples. The algorithm described in [HS64] can be used for the calculation of the Halton sequence, which is not more expensive as determination of usual pseudo-random numbers.

2.2 Implementation

Now, as we have defined the Gaussian basis functions we can formulate the phonon tracing algorithm to trace pressure. The algorithm requires the following input parameters:

- position of the sound source S
- reference pressure P_1 in 1 m distance from S
- emission distribution E of S
- one ore more listener positions l_i
- a triangulated scene with tagged materials m_j
- a reflection function $\rho_j : \Omega \mapsto [-1, 1]$ for each material
- an acoustic BRDF for each material (if applicable)
- a number of phonons n_{ph} to be traced from the source
- a lower pressure threshold ϵ and a maximum number of reflections n_{refl} for terminating the phonon paths

The output of our approach is a filter f_i for each listener position l_i corresponding to the impulse response with respect to the sound source and the phonon-map. For each phonon ph the map contains the pressure spectrum p_{ph}, the phonon's position pt_{ph} at the reflection point, the image source q_{ph} from which we can calculate the phonons outgoing direction v_{ph} and the traversed distance d_{ph}, number of reflections r_{ph}, and the material m_{ph} at the current reflection.

2.2.1 Phonon emission step

The phonons sent out from the source are associated with the following quantities:

- a pressure spectrum $p_{ph} : \Omega \mapsto \mathbb{R}^+$
- the virtual source q_{ph}
- the phonon's current position pt_{ph}

As described in [BDM+05] $n_e = 10$ frequency bands are used for the simulation. Furthermore, the wavelets introduced in [BDM+05] are the basis functions for the pressure spectrum.

Phonons are emitted from the source S according to the emission probability distribution E and have a unit pressure spectrum $p_{ph,i} = 1$ $(i = 1, ..., n_e)$ at their starting point. At

the intersection of the phonon ray with the scene, the virtual source q_{ph} is calculated as follows:

$$q_{ph} \quad \leftarrow \quad q_{ph} + 2 \cdot \langle (pt_{ph} - q_{ph}), n \rangle \cdot n \tag{13}$$

where n is the surface normal at the intersection point pt_{ph}, $\langle \cdot, \cdot \rangle$ denotes the scalar product. The pressure is reduced according to the reflection function ρ_j of the local material m_j. The phonon is fixed at the intersection point, and stored in the global phonon map. If the termination conditions are not satisfied, the tracing process is continued. Otherwise a new phonon is started from the source. The flow chart in figure 2 outlines the emission step.

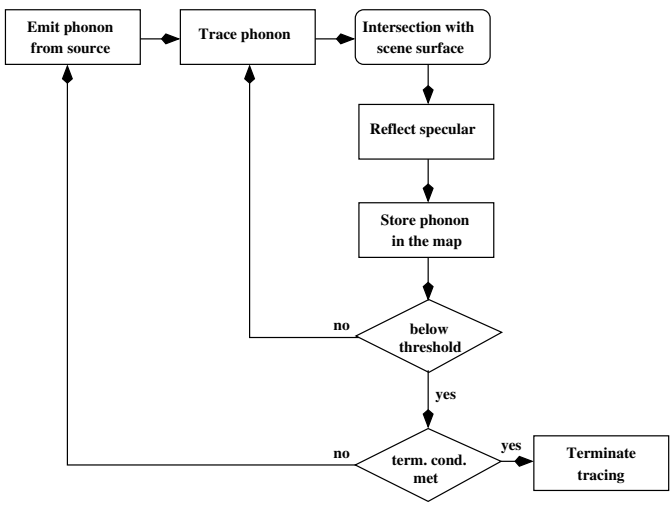


Figure 2: Flow chart diagram of the phonon emission step.

2.2.2 Phonon collection and filtering

The phonons are now collected in order to build the impulse response f_i at the listener position l_i. In the case of uniform absorption for all frequencies the contribution of a single phonon to the impulse response, corresponding to equation 1 reads:

$$p(t,x) = \frac{\rho_{tot}\, P_1}{d_{ph}}\, g\left(\angle\left(v_{ph}, l_i - q_{ph}\right) \right) \cdot \delta\left(t - \frac{d_{ph}}{c} \right) \tag{14}$$

where P_1 is the reference pressure at 1m from the source, ρ_{tot} is the product of the reflection coefficients along the phonon path and g is a Gaussian weighting function defined in equation 10 with $N = n_{ph}$.

Since most absorption coefficients α provided in the literature refer to energy, we must calculate the pressure related reflection coefficients:

$$\rho = \sqrt{1-\alpha}\ . \tag{15}$$

In the general case of frequency-dependent absorption, the unit pulse is subdivided as presented in [BDM+05]. The filter f_i then becomes a sum of wavelets (band pass filters) scaled and shifted as described above. Furthermore, the air absorption is considered, too.

3 Modeling diffuse reflections

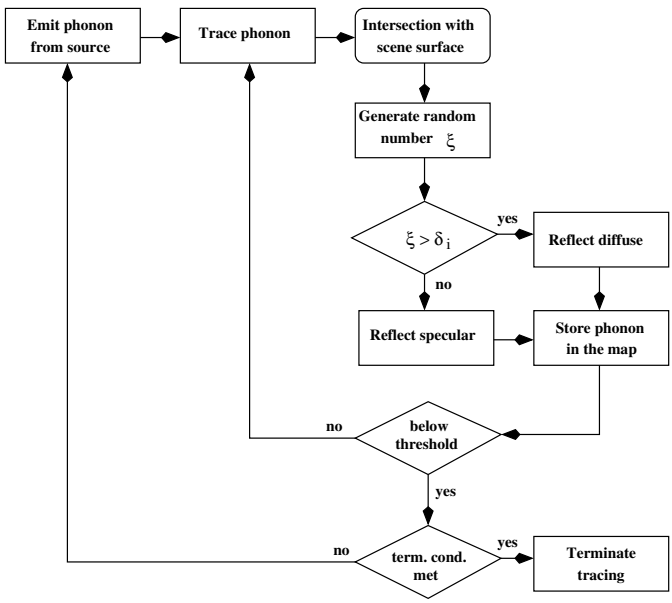


Figure 3: Flow chart diagram of the phonon emission step including diffuse reflections.

In the phonon tracing algorithm described in the previous section only specular reflections are considered. In fact for virtual acoustic applications surfaces are generally assumed to reflect the sound wave specularly. This assumption is true if the surface structure is significantly smaller in contrast to the wave length. Otherwise, the sound wave is reflected diffusely. This section describes the diffuse reflection model used in the modified phonon tracing algorithm.

In room acoustics, in order to model diffuse reflection, a scattering coefficient δ_i for each scene surface is introduced. The *phonon emission step* of the phonon tracing algorithm is then modified as follows. If the sound particle hits a scene surface, a number $\xi \in [0,1]$ is randomly chosen. If $\xi > \delta_i$ the phonon is reflected diffusely, otherwise the reflection is specular. In case of diffuse reflection the outgoing direction v_p is determined assuming ideal (Lambertian) diffuse reflection where the direction of the reflection is perfectly random over a hemisphere surrounding the phonon's position p_p (intersection point of the ray

with the surface) [Kut71]. The azimuthal angle θ is a random number ξ_1 in the interval $(-\pi, \pi)$ and the polar angle ϕ is given by the arc-cosine of the square root of a random number ξ_2 chosen in the interval $(0, 1)$:

$$\theta = \xi_1 \qquad \phi = \arccos(\sqrt{\xi_2}) \tag{16}$$

In the pressure phonon tracing algorithm the virtual source in equation 13 can now be calculated as follows:

$$q_p = p_p - d_p \cdot v_p \tag{17}$$

where p_p is the phonon's current position and d_p the traversed distance. Figure 3 depicts the *phonon emission step* including the diffuse reflection modeling.

Since the scattering coefficients δ_i are frequency dependent, we have to either perform a tracing step for each frequency band or use an average scattering coefficient for all frequencies. A better solution for this situation is subject to future research.

4 Acoustic level of detail

In computer graphics, there exist multiple techniques considering the level of detail representation of geometric objects [LRC+03]. In Virtual Reality for example the objects located in larger distance from the viewer position are rendered using fewer patches reducing rendering time. More details are added by the time the viewer is getting closer to the object providing more accurate representation. This section describes some different ideas how the phonon tracing algorithm can be modified in order to provide calculation of the room impulse response in different levels of detail depending on user requirements on computation time and computation accuracy.

4.1 Clustering

Before we discuss the different level of detail approaches we need to subdivide the particles stored in the phonon map into clusters which represent different wave-fronts which are being reflected by the walls. We use the information stored in the phonon map to accomplish the subdivision process. For early reflections all phonons within a cluster have the same trajectory, that is they are being reflected at the same plane and have equal absorption properties. Thus they satisfy the following criteria:

- equal numbers of reflections n_p
- and for each reflection:
  - equal material indices m_p (same object of the scene)

– equal surface normals n at the reflection position

Consequently phonons included in an early reflection cluster have equal virtual source position and pressure absorption factor.

At higher reflection orders, the clusters become smaller and smaller, until they contain only a single phonon. Hence, for cluster reconstruction at higher reflection order we disregard the constraint of equal trajectory and combine phonons having equal last reflection material. The cluster criteria are then as follows:

- equal material index m_p of last reflection
- equal surface normal n at the last reflection position

These clusters contain the residual phonons for different room surfaces after a prescribed number of first reflections. All the phonons within such residual cluster have different virtual sources and also different pressure decompositions.

4.2 LoD approaches

In the following, different variants of a level of detail method using the cluster information will be discussed.

The idea of providing different levels of the impulse response can be summarized in the following steps:

1. Find a representative phonon ph_r for each cluster
2. Calculate distance d_i of each phonon in the cluster to ph_r
3. Sort phonons according to d_i
4. Collect only phonons which are in a user prescribed distance d from ph_r

Now we will describe these individual steps in more detail.

Step 1. In the first step we determine the position pt_{ph_r} of the representative phonon ph_r for each cluster consisting of n particles:

$$pt_{ph_r} = \frac{\sum_{i=1}^{n} w_i \cdot pt_{ph,i}}{\sum_{i=1}^{n} w_i} \tag{18}$$

The weight w_i used for a single phonon depends on the phonon's pressure absorption factor due to reflections at the room walls. Since all phonons in an early reflection cluster have the same absorbtion factor the weights are all equal to 1 ($w_i = 1$). Whereas inside the late reflection clusters the weight w_i grows inversely proportional to this factor. Thus the representative phonon ph_r is shifted towards phonons having a amount of pressure absorbed due to reflections at least in one frequency band.

Step 2. Now we calculate the distance d_i of each phonon $pt_{ph,i}$ inside the cluster as Euclidean distance to ph_r:

$$d_i = |pt_{ph,i} - pt_{ph_r}| \tag{19}$$

The distances are used for the adaptive collection of particles in the next steps.

Step 3. In order to ensure a rapid collection of particles we sort them in ascending order with respect to the distance d_i.

Step 4. The information precalculated in the previous steps is now used for the collection of the particles. The user can prescribe a distance from the representative phonon ph_r. All phonons which are not farther away than this distance are collected. Another possibility is to predefine the number m of phonons to collect. Then only the m nearest particles to ph_r in the cluster are considered for the impulse response calculation. Thus, for a lowest level of detail only the representative phonon of each cluster is collected and all particles for the finest detail calculation.

5 Summary and Discussion

In this paper we have presented several modifications of the phonon tracing algorithm for room acoustics. The first modification was the improvement of the algorithm in order to trace pressure instead of energy allowing modeling of interference phenomena. Furthermore we extended the tracing step of the algorithm to consider diffuse reflections. We also presented a possibility to provide different levels of detail of the calculated impulse response depending on user input.

When including diffuse reflections one problem comes up. Since the scattering coefficients are frequency dependent one tracing step for all considered frequency bands is not enough. We either have to apply an average scattering coefficient and perform one tracing step, or we have to trace the phonons for each frequency band separately.

Concerning the level of detail we have presented a possibility to adapt the collection step of the algorithm depending on the users requirements on the accuracy of the resulting impulse response. Therefore we subdivided the particles in the phonon map into clusters and collected only a subset of the entire particles depending on the distance of the phonons to a representative phonon of the cluster. Further validations and verifications of this approach are necessary. In room acoustics early reflections are of greater interest as late reverberations, thus an approach incorporating this could be more appropriate. Additional benefits regarding the clustering should be investigated, for example the reduction of memory use for phonon map storage. All phonons inside early reflection clusters have equal total reflection coefficient, thus this information need to be stored only once for the cluster and not for each phonon separately. Also, the position of the virtual source is equal, so we can

store only the position of the virtual source and the pressure absorption coefficient for the cluster at the room surface. Statistical methods can then be used to perform the collection step, for example, collection of a prescribed number of particles equally distributed in the cluster. Therefore the regions of the clusters also have to be determined and stored. Regarding the residual particle clusters, more information has to be stored, due to the fact that the particles have both different pressure decomposition and different virtual source positions. Statistical approaches can be applied if the pressure as well as the virtual source positions distributions can be determined.

References

[AB79] J.B. Allen and A. Berkeley. Image method for efficiently simulating small-room acoustics. *J. Acoust. So. Amer.*, 65(4):943–950, Apr. 1979.

[BDM+05] M. Bertram, E. Deines, J. Mohring, J. Jegorovs, and H. Hagen. Phonon Tracing for Auralization and Visualization of Sound. In *IEEE Visualization*, Minneapolis, MN, October 2005.

[Bor84] J. Borish. Extension of the image model to arbitrary polyhedra. *J. Acoust. So. Amer.*, 75(6):1827–1836, 1984.

[FCE+98] Thomas A. Funkhouser, Ingrid Carlbom, Gary Elko, Gopal Pingali Mohan Sondhi, and Jim West. A Beam Tracing Approach to Acoustic Modeling for Interactive Virtual Environments. In *Computer Graphics (SIGGRAPH 98)*, pages 21–32, Orlando, FL, July 1998.

[HS64] J.H. Halton and G.B. Smith. Algorithm 247: Radical-Inverse Quasi-Random Point Sequence. *Communications of the ACM*, 7(12):701–702, December 1964.

[Kel96a] A. Keller. Quasi Monte Carlo Methods in Computer Graphics: The Global Illumination Problem. *Lecture in App. Math.*, 32:455–469, 1996.

[Kel96b] A. Keller. Quasi Monte Carlo Radiosity. *X. Pueyo and P. Schröder, editors, Rendering Techniques '96 (Proc. 7th Eurographics Workshop on Rendering)*, pages 101–110, 1996.

[KJM04] B. Kapralos, M. Jenkin, and E. Millios. Sonel Mapping: Acoustic Modeling Utilizing an Acoustic Version of Photon Mapping. In *IEEE International Workshop on Haptics Audio Visual Environments and their Applications (HAVE 2004)*, Ottawa, Canada, October 2-3 2004.

[Kro68] U. Krockstadt. Calculating the acoustical room response by the use of a ray tracing technique. *Journal of Sound and Vibrations*, 8(18):118–125, 1968.

[Kul84] U. Kulowski. Algorithmic Representation of the Ray Tracing Technique. *Applied Acoustics*, 18:449–469, 1984.

[Kut71] H. Kuttruff. Simulierte Nachhallkurven in Rechtecksräumen mit diffusem Schallfeld. *Acustica*, 25:333–342, 1971.

[LRC+03] D. Luebke, M. Reddy, J. Cohen, A. Varshney, B. Watson, and R. Huebner. *Level of Detail for 3D Graphics*. The Morgan Kaufmann Series in Computer Graphics, Elsevier Science, San Frncisco, CA, USA, 2003.

[Sve02] U.P. Svensson. Modelling Acoustic Spaces for Audio Virtual Reality. In *1st IEEE Benelux Workshop on Model based Processing and Coding of Audio (MPCA-2002)*, Leuven, Belgium, November 15 2002.

[Vor89] M. Vorländer. Simulation of the transient and steady-state sound propagation in rooms using a new combined ray-tracing/image-source algorithm. *J. Acoust. So. Amer.*, 86(1):172–178, 1989.

A Framework for Visualizing Multivariate Geodata

Ariane Middel
Computer Science Department
University of Kaiserslautern

Abstract: In urban planning, sophisticated simulation models are key tools to estimate future population growth for measuring the impact of planning decisions on urban developments and the environment. Simulated population projections usually result in bulky, large-scale, multivariate geospatial data sets. Millions of records have to be processed, stored, and visualized to help planners explore and analyze complex population patterns.
This paper introduces a database driven framework for visualizing geospatial multivariate simulation data from *UrbanSim*, a software-based simulation model for the analysis and planning of urban developments. The designed framework is extendable and aims at integrating methods from information visualization and cartography into planning processes.

1 Introduction

Estimates indicate that nowadays, approximately 80 percent of digital data is geospatially referenced [Nat03]. An ever-increasing availability of geodata raises the demand for new visual representations which are beyond classical 2D maps, particularly with regard to increasing data set size and dimensionality. This applies for acquirable real-world data as well as for estimated and projected data.
In urban planning, simulation models for predicting population growth have become highly complex. Model calculations result in large-scale, multidimensional spatial data sets. Projection data needs to be managed, manipulated, and visualized in order to enable planners to visually compare different planning scenarios and to evaluate simulated impacts of different land use policies. Especially the visualization task remains challenging, since traditional map-centered approaches lack support for high multivariability. The emerging research field Geovisualization (GeoVIS) takes advantage of techniques from cartography, geographic information science, and computer science to overcome visualization issues dealing with multi-attributed, large-scale spatial datasets.
Within the scope of this research domain, we develop a database driven framework to assist planners in their geovisual analyses of multidimensional simulation data. The framework uses *UrbanSim* demographic projection data for Maricopa County on a households-per-grid-cell basis. Additional residential building type data is derived from demographic household characteristics by statistical regression analysis. A geodatabase stores both simulated and estimated data for subsequent visualization. Within the database, modular scripts generate georeferenced scalable geometries calculated on the basis of multiple data attribute values for a map-based geovisualization.

2 Related Work

Visualizing geospatial datasets has long been a key issue in cartography. The cartographer Bertin established a basis for designing maps in his classical work "Semiology of Graphics" [Ber67] where he identified a set of fundamental visual variables and defined graphical rules for their appropriate use. Since then, Bertin's concepts have been constantly modified and extended. Modern cartography transfers design knowledge from 2D paper maps to new media. On-screen interactive maps are designed to assist in visual data exploration and analyses [AA99]. Cartographic visualization is also extended to abstract and non-geographic data by spatialization [SF03].

Whereas cartography primarily deals with representations constrained to a spatial domain, information visualization (InfoVIS) is mainly concerned with the display of large multivariate datasets. In the early 70's, Chernoff presented a technique to visualize trends in highly dimensional data by relating data to facial features [CR75]. Gradually over the years, new information visualization techniques were introduced, ranging from 2D scattergraphs to 3D treemaps. For a comprehensive overview of developments in Information visualization we refer to [SCM99] and [Tuf90].

Recently, efforts have emerged to combine techniques from both cartography and information visualization [Sku00, FS04]. Geographic visualization (GeoVIS) is a new, rapidly evolving domain, especially since the availability of geodata is increasing. In 1998, MacEachren compiled a first research agenda entitled "Visualization - Cartography for the 21st century" [Mac98] and addressed GeoVIS research challenges. Since then, cartographic and InfoVIS techniques have been applied to design integrated geovisualization tools frequently. Latest advances include multivariate analyses with self-organizing maps [GGMZ05, SH03], studies on human activity patterns using 3D space-time paths [MPJ04], and bivariate maps for public health studies [MGP$^+$04]. Most recent activities in geovisualization research are discussed in [Kra06].

Pinnel et al. [PDBB00] conducted a study on visualization designs for urban modeling. They found out that map-centered visualizations are the most useful portrayals for urban planning and analysis, since map layout encodes location information which is crucial for decision-making. A map-based visualization approach, "The Indicator Browser", was designed by Schwartzman et al. [SB07] to display *UrbanSim* simulation results. The browser uses comparative visualizations of 2D maps to satisfy multivariability which impedes human vision to recognize complex patterns across many dimensions.

Instead of encoding n-dimensional data in n maps, Tominski et al. [TSWS05] follow a different approach to display monthly health data. They visualize time dependent multivariate disease information as 3D pencil and helix icons geocoded on a 2D map. The research reported here helps to analyze complex patterns across multivariate, spatial, and temporal dimensions, but it lacks a powerful geodatabase and GIS functionality to manage, process, and distribute data. In the following, we will present a geovisualization framework to overcome those drawbacks.

3 A Framework for Visualizing Multivariate Geodata

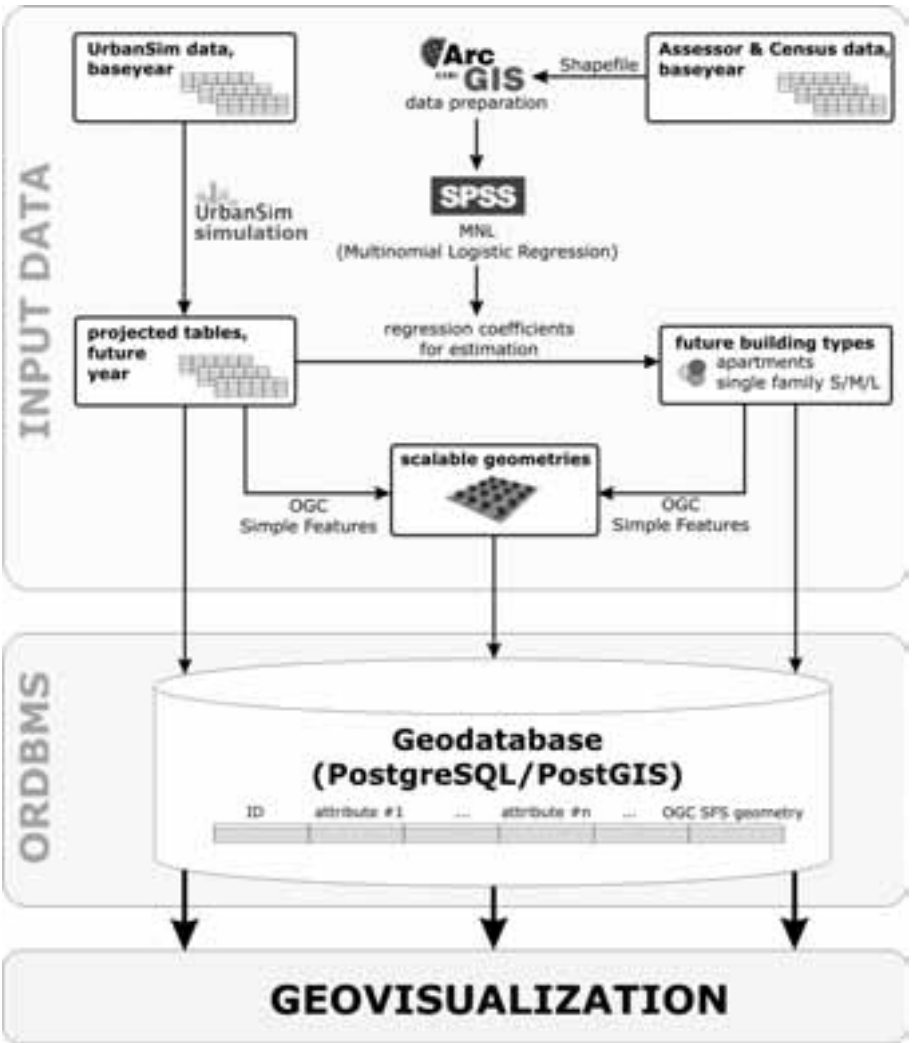


Figure 1: System Architecture (reproduced in color on p. 188)

The geovisualization framework mainly consists of three parts (cp. Fig. 1): an input data layer, an object-relational data base management system, and a visualization layer. Within the data layer, demographic data is aggregated from different sources and prepared in ArcGIS. An *UrbanSim* simulation projects demographic household characteristics for a predefined year in the future (see Section 3.1). For the same year, demographic data is aggregated to estimate future residential building types with multinomial logistic regression

in SPSS (see Section 3.2). The demographic simulation results and the estimated building types are projected into 3D space with scalable, georeferenced geometries (see Section 3.4). Data and geometry is stored and managed in an object-relational geodatabase (see Section 3.3) and finally visualized on top of map a. In the remainder of the paper, we will explain the system architecture in detail.

3.1 Data modeling with *UrbanSim*

UrbanSim [BW04] is a large scale land use and transportation simulation software to model the possible long-term effects of different policies on urban developments. More precisely, it simulates the interactions between transportation, land use, and public policy at household and job level. *UrbanSim* consists of numerous model components simulating different actors in the urban development process, e.g. discrete choice models for relocating households and jobs. The open source simulation model was developed by a research group in Washington and is currently implemented in the Digital Phoenix project [Sub07] at ASU, Arizona State University to predict population growth in the Phoenix Metropolitan area.

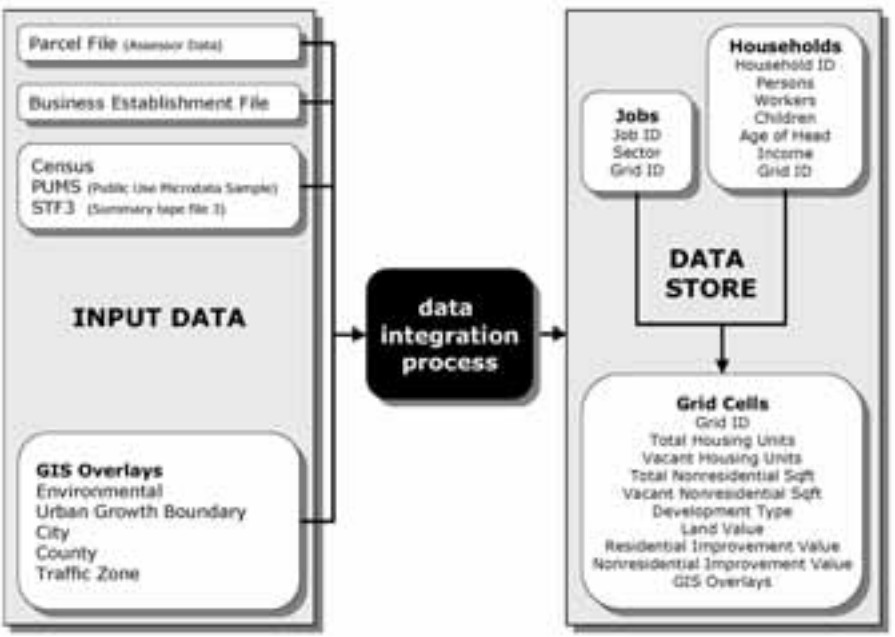


Figure 2: Data integration process (c.f. [Wad02])

UrbanSim input data is aggregated from various sources (see Fig. 2) and spatially mapped to a georeferenced grid cells file. Typical grid cell sizes are $150m \times 150m$ or 1 square

mile. Base year data includes information on parcels from the Assessor's office, employment data, and Census data. Additional input data on city, county, and urban growth boundaries as well as environmental and traffic information is overlaid in ArcGIS. The *UrbanSim* data store contains a grid cells table, a jobs table holding information on each job and its employment sector in the grid cells, and a household table. The latter is synthesized probabilistically and compiles demographic characteristics for each household in the metropolitan area.
Subsequent to the data integration process, the *UrbanSim* simulation is run for a predefined number of years. The output projection tables include data on future households with grid cell location and demographic characteristics, future jobs, and exogenous input data. The projection results can be integrated into a variety of analyses, e.g. the analysis of future population density, material use, or carbon footprints.

3.2 Estimating Residential Bulding Types from Demographic Data

Knowledge of future residential building types is essential for the above mentioned analyses and for visualization purposes. To derive the type of residential units from *UrbanSim* demographic data for each grid cell we assume a relationship between household characteristics and building types. Multinomial Logistic regression (MNL) is an analytically appropriate technique to link demographic attributes with dwelling types. This statistical method is widely used in social sciences and economics to discover hidden relationships between variables. Multinomial logistic regression is equivalent to the conditional discrete choice model, first presented and most notably influenced by McFadden [McF73, McF76, McF78]. MNL models the interaction between response and explanatory variables. The multinomial response is polytomous, it can have multiple unordered qualitative categories as outcomes. The parameter values of the regression are estimated according to the following equation [PX99]:

$$z_{ij} = \boldsymbol{x}_i' \boldsymbol{\beta}_j = \sum_{k=0}^{K} \beta_{jk} x_{ik} = \alpha_j + \sum_{k=1}^{K} \beta_{jk} x_{ik} \tag{1}$$

Here, $\boldsymbol{x}$ is the vector of predictors storing all relevant demographic variables and $\boldsymbol{\beta}$ denotes the regression parameter vector containing estimation coefficients.
With the calculated set of coefficients β_{jk}, the probability P_{ij} that a certain building type category j is chosen over any other building type category yields:

$$P(y_i = j|\boldsymbol{x}_i) = P_{ij} = \frac{e^{z_{ij}}}{\sum_{j=1}^{J} e^{z_{ij}}} = \frac{e^{\boldsymbol{x}_i' \boldsymbol{\beta}_j}}{\sum_{j=1}^{J} e^{\boldsymbol{x}_i' \boldsymbol{\beta}_j}} = \frac{e^{\boldsymbol{x}_i' \boldsymbol{\beta}_j}}{1 + \sum_{j=2}^{J} e^{\boldsymbol{x}_i' \boldsymbol{\beta}_j}} \tag{2}$$

Basically, we distinguish between multifamily dwellings and single family dwellings. Predictable categories for the regression analysis are apartments as well as single family houses with small, medium, and large lots. Those categories can be further refined if additional data is available, e.g. the number of storeys.

Variables explaining the building types are primary demographic characteristics corresponding to the *UrbanSim* output: household income, average household size, number of households, median age, presence of children, and percentage of minorities and Hispanics. Demographic data is compiled and synthesized to the *UrbanSim* grid cell base file from Assessor's data and Census block groups. Afterwards, the regression analysis is run in SPSS with the presented explanatory variables and building type categories. To predict future building types, we apply the estimated coefficients β_{jk} to *UrbanSim* simulation data. The resulting dwelling categories are stored together with demographic attributes from *UrbanSim* in a geodatabase, as explained in the next section.

3.3 Geodatabase

The inherent geospatial nature of *UrbanSim* gridded data requires the implementation of a spatial database. So-called geodatabases extend the database concept to storage, query, and editing of georeferenced objects. Accordingly, the projected demographic data from *UrbanSim* and the estimated building type data is stored in a geodatabase implemented in PostgreSQL [Pos07], an open source object-relational database management system (ORDBMS). In addition, a PostGIS module [SLRL05] serves as geospatial extension to the PostgreSQL backend server. PostGIS enables PostgreSQL to integrate spatial data structures in the database, query geographic objects, and serve them to GIS applications.
In general, each spatial table in the geodatabase represents a separate PostGIS layer. An ancillary table contains meta-data on the associated geodetic datum, here the coordinate reference system NAD 1983. Each distinct geographic object constitutes a record in a spatial table and associated attribute information is stored in data columns. In particular, the implemented geodatabase includes a table with records for *UrbanSim* grid cells and columns for *UrbanSim* projected demographic variables and the associated estimated building type category.
PostGIS provides a dedicated geometry column which contains geometric information for each feature in the form of point, line, or polygon data types. We will use this geometry column to store the scalable geometries generated from attribute data for the geovisualization.

3.4 Geometry Generation and Visualization

The designed PostgreSQL geodatabase provides the basis for a multivariate information visualization. *UrbanSim* simulation data and estimated building types stored in the database are encoded in scalable 3D geometries and visualized in a geospatial context on top of a map. Geometric objects are created for every grid cell directly within the geodatabase from geometric primitives.
PostGIS is fully OGC compliant which means it conforms to the "Simple Features for SQL" specification (SFS) from the OpenGIS Consortium [Ope07]. The OGC SFS defines standard geometric data types (see Fig. 3) and functions for manipulating geometry.

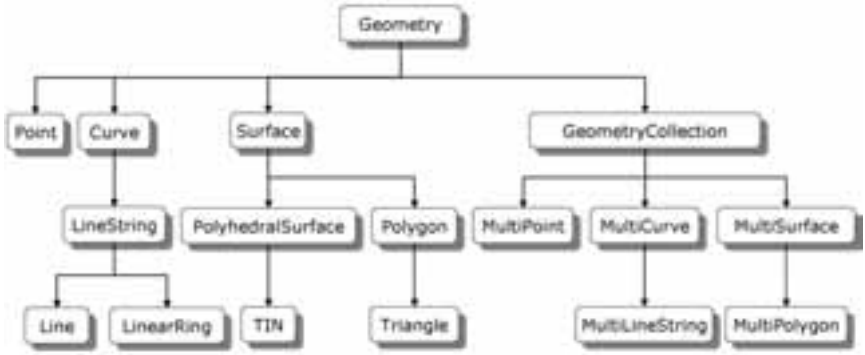


Figure 3: OGC Simple Feature Specification (cf. [Ope07])

PostGIS supports 1D and 2D geometric primitives but lacks geometric data types for volumes. Therefore, 3D objects have to be constructed using 3D polygons. Geometry calculations are performed using dedicated PostGIS geometry functions. For each record in the geodatabase, modular SQL scripts select the associated attributes to be visualized and generate user-specified geometries. At the same time, size and shape of the geometries are scaled according to normalized attribute values.
The geometry generating SQL modules can be implemented to calculate arbitrarily shaped discrete geometries. To visualize urban structures and household demographics, an implementation of iconized buildings seems obvious. In this case, shape parameters to vary by attribute values can be floor plan, roof height, ridge height or the number of chimneys for instance. Other possible geometries include stepped pyramids, discretized cones and tori. The geometric objects are generated for each *UrbanSim* grid cell and stored in the geodatabase. From there, the geometries can be geovisualized in a map context, e.g. with Google Earth.

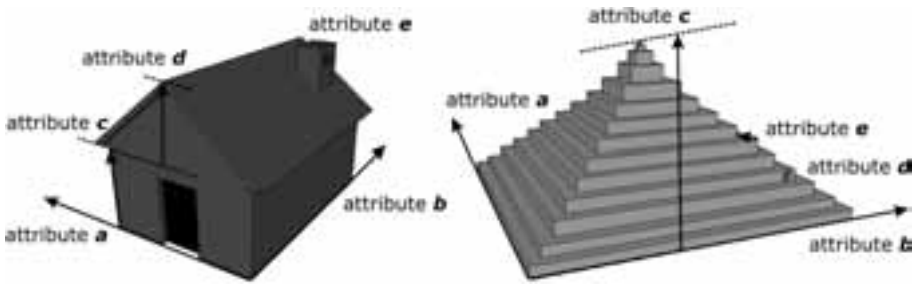


Figure 4: Examples for scalable geometries

4 Conclusions and Future Work

This paper presented an integrated framework for processing, storing, and visualizing multivariate geodata. The introduced framework uses simulation data from *UrbanSim*, a planning tool for the simulation, comparison, and evaluation of different planning scenarios to better assess the impacts of various policy decisions. Within the framework, residential building types are estimated from demographic household characteristics by means of multinomial logistic regression. The aggregated output data is stored in a PostgreSQL geodatabase and provides the basis for further visualization.

The developed visualization approach is a cross-disciplinary effort to integrate methods from InfoVIS and cartography for supporting space-related decision-making. Geovisualization is driven by the need to visualize geospatial data which is multivariate, large, and multidimensional at the same time. Our framework uses a map-centered visualization where data attributes are represented by georeferenced scalable 3D geometries. Geometric shapes are generated within the geodatabase and scaled according to normalized attribute values.

The presented geovisualization approach is extendable, since data representation is not restricted to specified geometric shapes. SQL modules can easily be added to generate various geometries, e.g. cones or tori instead of iconic dwellings. This makes the framework generic enough to be transferable to other application areas dealing with any kind of multidimensional geodata.

Besides extending the variety of geometry modules, future work includes the implementation of an interactive web-based 3D visualization. A PHP middleware will be designed, allowing the user to interactively choose the parameters for the geometries and the attributes he wishes to visualize from the geodatabase. A KMZ file will be generated from the database according to the user's geometry specifications and visualized in Google Earth on the fly. An internet based choice and display of data will provide the framework to a variety of user groups and increase the number of geovisualization tasks to be solved.

Acknowledgements This work was supported by the German Science Foundation (DFG, grant number 1131) as part of the International Graduate School (IRTG) in Kaiserslautern on "Visualization of Large and Unstructured Data Sets. Applications in Geospatial Planning, Modelling, and Engineering".

References

[AA99] Gennady L. Andrienko and Natalia V. Andrienko. Interactive Maps for Visual Data Exploration. In *International Journal Geographic Information Science*, volume 13, pages 355–374, June 1999.

[Ber67] Jacques Bertin. *Semiology of Graphics*. University of Wisconsin Press, 1967.

[BW04] Alan Borning and Paul Waddell. Integrated Land Use, Transportation, and Environmental Simulation: UrbanSim Project Highlights. In *dg.o '04: Proceedings of the 2004 annual national conference on Digital government research*, pages 1–2. Digital Government Research Center, 2004.

[CR75] H. Chernoff and M. H. Rizvi. Effect on Classification Error or Random Permutations of Features in Representing Multivariate Data by Faces. *Journal of American Statistical Association*, 70:548–554, 1975.

[FS04] Georg Fuchs and Heidrun Schumann. Visualizing Abstract Data on Maps. *IV '00*, 00:139–144, 2004.

[GGMZ05] Diansheng Guo, Mark Gahegan, Alan M. MacEachren, and Biliang Zhou. Multivariate Analysis and Geovisualization with an Integrated Geographic Knowledge Discovery Approach. *Cartography and Geographic Information Science*, 32(2):113–132, April 2005.

[Kra06] Menno-Jan Kraak. Visualization Viewpoints: Beyond Geovisualization. *IEEE Computer Graphics and Applications*, 26(4):6–9, Jul/Aug 2006.

[Mac98] Alan M. MacEachren. Visualization - Cartography for the 21st century. In *Proceedings of the 7th Annual Conference of Polish Spatial Information Assocaition*, Warsaw, Poland, May 1998.

[McF73] Daniel L. McFadden. Conditional Logit Analysis of Qualitative Choice Behavior. *Frontiers in Econometrics*, pages 105–142, 1973.

[McF76] Daniel L. McFadden. *Properties of the Multinomial Logit (MNL) Model*. Urban Travel Demand Forecasting Project, Institute of Transportation Studies, 1976.

[McF78] Daniel L. McFadden. *Modelling the Choice of Residential Location*. Institute of Transportation Studies, University of California, 1978.

[MGP+04] Alan M. MacEachren, Mark Gahegan, William Pike, Isaac Brewer, Guoray Cai, Eugene Lengerich, and Frank Hardisty. Geovisualization for Knowledge Construction and Decision Support. *IEEE Computer Graphics and Applications*, 24(1):13–17, Jan/Feb 2004.

[MPJ04] Kwan Mei-Po and Lee Jiyeong. Geovisualization of Human Activity Patterns Using 3D GIS: A Time-Geographic Approach. In Michael F. Goodchild M and Donald G. Janelle, editors, *Spatially Integrated Social Science: Examples in Best Practice*, pages 48–66, Oxford, 2004. Oxford University Press.

[Nat03] National Research Council. *IT Roadmap to a Geospatial Future*. National Academies Press, 2003.

[Ope07] Open Geospatial Consortium. OGC Simple Feature Specification. Available online at: <http://www.opengeospatial.org/standards>, 2007. (Last Accessed: 09-23-2007).

[PDBB00] L. Denise Pinnel, Matthew Dockrey, A. J. Bernheim Brush, and Alan Borning. Design of Visualizations for Urban Modeling. In *VisSym '00: Joint Eurographics – IEEE TCVC Symposium on Visualization*, Amsterdam, The Netherlands, May 2000.

[Pos07] PostgreSQL Global Development Group. PostgreSQL. Available online at: <http://www.postgresql.org>, 2007. (Last Accessed: 09-23-2007).

[PX99] Daniel Powers and Yu Xie. *Statistical Methods For Categorical Data Analysis*. Elsevier Science & Technology, 1999.

[SB07] Yael Schwartzman and Alan Borning. The Indicator Browser: A Web-Based Interface for Visualizing UrbanSim Simulation Results. In *HICSS '07: Proceedings of the 40th Annual Hawaii International Conference on System Sciences*, page 92, Washington, DC, USA, January 2007. IEEE Computer Society.

[SCM99] Ben Shneiderman, Stuart K. Card, and Jock D. Mackinlay. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann, San Francisco, CA, 1999.

[SF03] André Skupin and Sara I. Fabrikant. Spatialization Methods: A Cartographic Research Agenda for Non-Geographic Information Visualization. In *Cartography and Geographic Information Science*, volume 30, pages 95–115, 2003.

[SH03] André Skupin and Ron Hagelman. Attribute Space Visualization of Demographic Change. In *11th ACM GIS Symposium (GIS'03)*, New Orleans, Louisiana, USA, November 7-8 2003. ACM Press.

[Sku00] André Skupin. From Metaphor to Method: Cartographic Perspectives on Information Visualization. In *Proceedings of the IEEE Symposium on Information Visualization (InfoVIS 2000)*, pages 91–97, Salt Lake City, UT, USA, October 9-10 2000. Los Alamitos: IEEE Computer Society.

[SLRL05] Sandro Santilli, Mark Leslie, Paul Ramsey, and Jeff Lounsbury. PostGIS. Available online at: <http://postgis.refractions.net>, 2005. (Last Accessed: 09-22-2007).

[Sub07] Subhrajit Guhathakurta and Janet Holston and Yoshi Kobayashi and Tim Lant and Mookesh Patel. Digital Phoenix Project. Available online at: <http://www.digitalphoenix-asu.net>, 2007. (Last Accessed: 09-26-2007).

[TSWS05] Christian Tominski, Petra Schulze-Wollgast, and Heidrun Schumann. 3D Information Visualization for Time Dependent Data on Maps. In *IV '05: Proceedings of the Ninth International Conference on Information Visualisation (IV'05)*, pages 175–181, Washington, DC, USA, 2005. IEEE Computer Society.

[Tuf90] Edward Tufte. *Envisioning Information*. Graphics Press, 1990.

[Wad02] Paul Waddell. UrbanSim: Modeling Urban Development for Land Use, Transportation and Environmental Planning. *Journal of the American Planning Association*, 68(3):297–314, 2002.

Natural Neighbor Concepts in Scattered Data Interpolation and Discrete Function Approximation

Tom Bobach and Georg Umlauf

University of Kaiserslautern
Geometric Algorithms Group
Department of Computer Sciences
D-67653 Kaiserslautern, Germany
http://www-umlauf.informatik.uni-kl.de/{˜bobach | ˜umlauf}

Abstract: The concept of *natural neighbors* employs the notion of distance to define local neighborhoods in discrete data. Especially when querying and accessing large scale data, it is important to limit the amount of data that has to be processed for an answer. Because of its implicit definition on distances, the natural neighbor concept is extremely well suited to provide meaningful neighborhoods in spatial data with a scattered, inhomogeneous distribution.

This paper revisits some unique properties of natural neighbor based methods and summarizes important findings for their successful application to scattered data interpolation, and the computation of discrete harmonic functions.

1 Introduction

Many scientific areas deal with phenomena in a spatial context when modeling or investigating real world problems. Computational approaches, be it in the endeavor of data generation or data analysis, inherently deal with discrete data representations. Based on the spatial structure of the data which can range from completely regular to inhomogeneous and scattered, the way how we access the data model has big impact on space and time requirements of algorithms.

In the first part of this work we are concerned with smooth, local interpolants in large, inhomogeneous, and unstructured data, a challenging problem for data representation and -access. We focus on the local reconstruction of a real-valued function from data in an adequate spatial neighborhood. Often – in absence of further knowledge about the modeled problem – this neighborhood is best expressed in terms of natural neighbors, a neighborhood relation defined on the Voronoi diagram of the data sites.

Another problem that is related to locally defined functions is the computation of discrete harmonic functions. We discuss implications of a discretization domain that itself is a function of time and how natural neighbors are beneficial in this context.

In the remainder we first revisit previous work while discussing under which circumstances

natural neighbor concepts are feasible. We then turn to the algorithmic impact caused by natural neighbor concepts in scattered data interpolation. The issues that arise in the course of applying C^2-continuous natural neighbor interpolation are presented along with a brief sketch of available solutions. Leaving the area of function interpolation we then look at the computation of discrete harmonic functions over scattered point sets under continuous deformation.

2 The Natural Neighbor Concept

In a set of points $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$, we call $\mathbf{x}_i$ and $\mathbf{x}_j$ *natural neighbors* if there is a point that is closer to $\mathbf{x}_i$ and $\mathbf{x}_j$ than to any other point in $\mathbf{X}$. This notion of spatial proximity is formalized in the Voronoi diagram, also known as Dirichlet tessellation or Thiessen polygons [Aur91, OBSC00]. The discrete set $\mathbf{X}$ of given points is called the set of *Voronoi sites*, and its Voronoi diagram is the partition of space into so-called *Voronoi tiles* $\{\mathscr{T}_1, \ldots, \mathscr{T}_m\}$, such that

$$\mathbf{p} \in \mathscr{T}_i \;\Leftrightarrow\; d(\mathbf{p}, \mathbf{x}_i) \leq \min_{1 \leq j \leq n} d(\mathbf{p}, \mathbf{x}_j) \tag{1}$$

for every point $\mathbf{p}$ and a distance measure d. In this structure, natural neighbors are exactly those Voronoi sites whose tiles have a non-empty intersection. It is noteworthy that (1) can be generalized to encompass more complex shaped Voronoi sites such as curves, polyhedra etc. – as long as a distance measure can be defined.

One of our goals is the reconstruction of an unknown function that is locally defined by nearby data, where proximity is based on the distance measure. In this context natural neighbors provide an excellent notion of neighborhood.

Despite the flexible definition of Voronoi diagrams given in (1), only few choices for the shape of Voronoi sites and distance measures are algorithmically feasible. We present two of the most basic and elegant variants.

The *traditional Voronoi diagram* is uniquely defined by the point-shaped sites $\mathbf{x}_i \in \mathbf{R}^n$ and the Euclidean metric $d_2(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2$. It comes with properties that are typically required in locally operating geometric algorithms: invariance under rigid transformations and uniform scaling. Moreover, its tiles $\mathscr{T}_1, \ldots, \mathscr{T}_m$ are convex polyhedra, allowing easy and robust geometric access. The most important fact about Voronoi diagrams, from a computational point of view, is their duality to *Delaunay triangulations*. These data structures are well-understood with a large support in terms of fast and robust algorithms and computation libraries, and allow easy and fast access to the entities of the Voronoi diagram.

The Voronoi diagram can be generalized e.g. in terms of different distance measures, or by the shape of sites. Each generalization usually leads to more complex algorithms and data structures and causes the loss of one or the other desirable property of the traditional Voronoi diagram. One noteworthy generalization that retains most of the geometric elegance of the traditional Voronoi diagram is the *power-* or *Laguerre Voronoi diagram.*

Based on a non-uniform metric

$$(d_p(\mathbf{p}, \mathbf{x}_i))^2 = (d_2(\mathbf{p}, \mathbf{x}_i))^2 - w_i \tag{2}$$

that uses weights w_i associated with the sites $\mathbf{x}_i$, it still leads to convex Voronoi tiles and is dual to the *regular triangulation* of a weighted point set, which is only slightly more complicated to maintain than the Delaunay triangulation.

Generalizations in the shape of the sites usually do not dualize as easily to a triangulation and require more general graph structures to represent the Voronoi diagram.

Depending on the characteristics of a data set, natural neighbor concepts can prove valuable for local function approximation or interpolation, the key indicators being

- a spatial setting,
- proximity that is based on a distance measure with a geometric interpretation,
- a sought-after function that locally depends on known data.

3 Related Work

Although the general concept of natural neighbors is not restricted to interpolation and local coordinates, the latter are their most prominent applications. The introduction of natural neighbor based local coordinates which possess C^1 continuity almost everywhere was done in the pioneering work of [Sib80] under the nowadays misleading term "natural neighbor coordinates" and later applied to globally C^1 scattered data interpolation in [Sib81]. Further results on the properties of the C^1 coordinates followed in [Far90, Pip92].

Following the concept underlying the C^1-continuous coordinates, C^0-continuous coordinates have been independently introduced by [CFL82] in the context of random lattices in nuclear physics, by [BIK$^+$97] under the term "Non-Sibsonian" coordinates, and by [Sug99] as "Laplace natural neighbor" coordinates.

In [HS00b] an integral relation between C^0 and C^1 coordinates was shown and generalized to almost everywhere C^k-continuous natural neighbor coordinates.

The drawback of reduced (C^0) continuity which interpolants based on the coordinates show at the data sites was overcome in [Sib81, Far90] by devising globally C^1-continuous interpolants, and in [HS04] by devising a globally C^2-continuous interpolant.

A generalization of natural neighbor interpolation to line- and circle-shaped Voronoi sites was performed in [AMG98, GF99, HS00a].

The algorithmic implications of natural neighbor interpolation have been investigated in [BBU06, BS95, Hiy05], and acceleration approaches were presented in [FEK$^+$05, PLK$^+$06].

Support for the manipulation and access of generalized Voronoi diagrams by means of graphics hardware is due to [HCK$^+$99].

The implicit definition of higher order of continuity based on data that lacked explicit derivative information has been the goal of [Sib81, Cla96, Flö03, BBU06].

Finally, the non-trivial definition of natural neighbor coordinates in the tangent space of manifolds was investigated in [BC00, Flö03].

4 Smooth Data Interpolation with Natural Neighbors

The scattered data interpolation problem can be stated as follows. We assume a scattered, unstructured set of data sites $\mathbf{X}$ and a partially defined function $f : \mathbf{X} \to \mathbf{R}$ together with the first k terms of its Taylor expansion, $f^{(1)}, \ldots, f^{(k)}$, at each site. We seek to construct a function Φ whose Taylor expansion up to the k-th term agrees with that of f at every site and otherwise satisfies properties including, but not limited to, smoothness and variation minimization. In the following we consider the interpolation of scalar values and derivative information up to order two in $\mathbf{R}^2$, i.e. gradients $\nabla f(\mathbf{x}_i)$ and Hessians $\mathscr{H} f(\mathbf{x}_i)$, and concentrate on the evaluation of $\Phi(\mathbf{q})$ at a query position $\mathbf{q} \in \mathbf{R}^2$.

In the corresponding research area of scattered data interpolation, many efficient local and global schemes have been proposed, able to deal with a large variety of input data. Interpolation schemes with global support lead to better results in general than schemes with local support, at the expense of considerably increased computational complexity. Local schemes, on the other hand, depend on a definition of "local" that often amounts to a user-provided parameter, making especially inhomogeneously distributed data hard to deal with.

Natural neighbor scattered data interpolation determines the local support for the reconstruction from the set of natural neighbors, thus coming with implicit and automatic control over the neighborhood. The evaluation of natural neighbor schemes at a point $\mathbf{q}$ operates on the set $N_{\mathbf{q}} := \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ of natural neighbors in the Voronoi diagram of $\mathbf{X} \cup \{\mathbf{q}\}$, and involves the following steps:

1. computation of coordinates $\lambda(\mathbf{q}) = (\lambda_1(\mathbf{q}), \ldots, \lambda_n(\mathbf{q})) \in \mathbf{R}^n$ of $\mathbf{q}$ with respect to the points in $N_{\mathbf{q}}$,
2. if not provided, estimation of the first k terms $f^{(1)}, \ldots, f^{(k)}$ of the Taylor expansion with respect to $\mathbf{x}$ at the data points, and
3. setup and evaluation of a multivariate function φ in the coordinates to locally control the shape of the interpolant, giving the interpolant as $\Phi = \varphi(\lambda(\mathbf{q}))$.

The rest of this section is dedicated to the algorithmic efforts involved in the computation of C^0, C^1, and C^2-continuous interpolants. After pointing out in what respect the implementation of the considered C^2 interpolant is more complex, we devise an algebraic rather than geometric approach for the computation of coordinates to simplify the algorithmic realization. A similar difficulty can be observed for the estimation of higher order derivatives, for which we refer to a recursive approach with limited complexity.

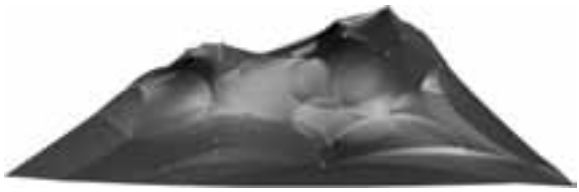


Figure 1: Discrete, scalar valued height field interpolated using the C^0-continuous interpolant Φ^0.

4.1 Natural neighbor schemes up to first order continuity

Natural neighbor coordinates as proposed in [Sib80, CFL82, Sug99, BIK$^+$97] are computed from sizes of geometric entities in the Voronoi diagram of the data sites. Thanks to the duality between Voronoi diagram and Delaunay triangulation, operations on the Voronoi diagram can in general be reduced to the traversal of adjacent elements in the Delaunay triangulation. We briefly sketch the steps involved in the computation of C^0 and C^1-continuous interpolants.

The set $N_{\mathbf{q}}$ of natural neighbors is identical to the set of edge-adjacent neighbors of $\mathbf{q}$ in the Delaunay triangulation of $\mathbf{X} \cup \{\mathbf{q}\}$. We can assume $N_{\mathbf{q}} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ to be ordered counter-clockwise around $\mathbf{q}$. The vertices $\mathbf{v}_1, \ldots, \mathbf{v}_n$ of $\mathbf{q}$'s Voronoi tile $\mathscr{T}_{\mathbf{q}}$ are the circumcenters of the triangles $\triangle(\mathbf{q}, \mathbf{x}_i, \mathbf{x}_{i+1}), 1 \leq i \leq n$, where we assume $N_{\mathbf{q}}$ to be cyclic, i.e. $\mathbf{x}_{n+1} := \mathbf{x}_1$.

Now, the C^0-continuous natural neighbor coordinates $\lambda^0(\mathbf{q}) = (\lambda_1^0(\mathbf{q}), \ldots, \lambda_n^0(\mathbf{q}))$ of $\mathbf{q}$ with respect to $\mathbf{x}_1, \ldots, \mathbf{x}_n$ are defined as

$$\lambda_i^0(\mathbf{q}) = \hat{\lambda}_i^0(\mathbf{q}) \;/\; \sum_{\mathbf{x}_j \in N_{\mathbf{q}}} \hat{\lambda}_j^0(\mathbf{q}), \qquad \hat{\lambda}_i^0(\mathbf{q}) = \|\mathbf{v}_{i-1} - \mathbf{v}_i\| \;/\; \|\mathbf{x}_i - \mathbf{q}\|.$$

From the above definitions it is obvious that the computation of $\lambda^0(\mathbf{q})$ amounts to the iteration of the one-ring of $\mathbf{q}$ after it has been inserted into the Delaunay triangulation of $\mathbf{X}$. In three-dimensions, the computation of $\lambda_i^0(\mathbf{q})$ requires the iteration around the Delaunay edge $(\mathbf{q}, \mathbf{x}_i)$ and the area computation of a convex polygon.

A C^0-continuous interpolant is now given by

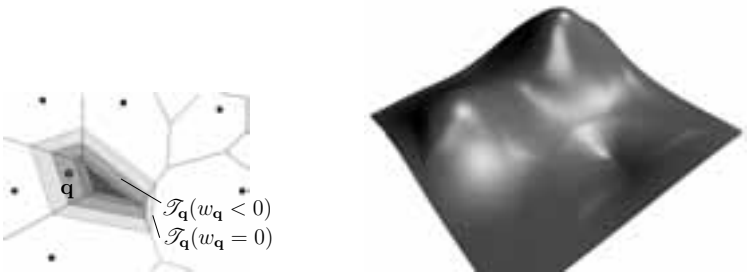
$$\Phi^0(\mathbf{q}) = \varphi(\lambda^0(\mathbf{q})) = \sum_{\mathbf{x}_i \in N_{\mathbf{q}}} \lambda_i^0(\mathbf{q}) f(\mathbf{x}_i).$$

An example of a height field interpolation based on Φ^0 is shown in Figure 1.

Natural neighbor coordinates with C^1-continuity in $\mathbf{R}^2 \setminus \mathbf{X}$, initially proposed by Sibson [Sib80], are defined by

$$\lambda_i^1(\mathbf{q}) = |\mathscr{T}_i \cap \mathscr{T}_{\mathbf{q}}| \;/\; |\mathscr{T}_{\mathbf{q}}|,$$

where $\mathscr{T}_i$ is the tile of $\mathbf{x}_i$ in the Voronoi diagram of $\mathbf{X}$, and $\mathscr{T}_{\mathbf{q}}$ is the tile of $\mathbf{q}$ in the Voronoi diagram of $\mathbf{X} \cup \{\mathbf{q}\}$. In [Wat92] it was observed that $|\mathscr{T}_i \cap \mathscr{T}_{\mathbf{q}}|$ is the sum of



(a) (b)

Figure 2: (a) The weight dependent Voronoi tile $\mathscr{T}_{\mathbf{q}}(w)$ in the power diagram. (b) Discrete, scalar valued height field with given derivatives interpolated using the C^1-continuous interpolant Φ^1.

signed areas of "dual triangles" which are defined on circumcenters of triangles formed by $\mathbf{q}$ and the vertices of Delaunay triangles that would locally be modified by the insertion of $\mathbf{q}$. The beauty of this lies in the generality of the resulting formula for the intersection volumes, which is directly applicable to higher dimensions, operating on "dual simplexes". It should be noted, however, that this approach is numerically unstable near and on edges (or $k-1$-simplexes) of the Delaunay triangulation due to an infinite volume of the dual triangle (simplex). A remedy to this can be found in [Hiy05].

To achieve C^1 continuity also at the data sites $\mathbf{X}$ by interpolating gradients, two constructions have been proposed in [Sib81] and [Far90] that define φ as a polynomial in $\lambda^1(\mathbf{q})$ such that the gradient of the interpolant matches that defined at the data site. The construction in [Far90] builds on cubic Bézier simplexes b^3 in n variables for which it is close to trivial to model derivatives at their vertices, which coincide with the data sites. For a thorough treatment of Bézier simplexes, see [dB87]. Consequently, the C^1 interpolant is

$$\Phi^1(\mathbf{q}) = \varphi^1(\lambda^1(\mathbf{q})) = b^3(\lambda^1(\mathbf{q})). \tag{3}$$

Figure 2(b) shows a discrete height field with given gradients interpolated using Φ^1.

In case of unknown gradients, [Sib81] proposed to estimate the gradient at $\mathbf{x}_i$ from the weighted least squares plane through $N_{\mathbf{x}_i}$. This, being a standard approach to derivative estimation, works remarkably well thanks to the utilization of λ^1 as weights in the least squares fit.

4.2 Natural neighbor coordinates with second order continuity

The evaluation of C^0 and C^1-continuous interpolants requires only the Delaunay triangulation of $\mathbf{X}$ and a simple traversal of vertex-adjacent elements. This section deals with the first step in globally C^2-continuous interpolation, the computation of local coordinates.

In [HS00b], a framework for the computation of natural neighbor coordinates with C^k-

continuity in $\mathbf{R}^2 \setminus \mathbf{X}$ was proposed that contained above mentioned C^0 and C^1 coordinates as special cases. The definition of these is based on the concept of power diagrams, which differ from the ordinary Voronoi diagram in the use of the custom distance measure (2). The power diagram shares all properties of the Voronoi diagram with additional control of the Voronoi tile sizes by means of the site weights w_i, where tiles can vanish for small enough values of w_i.

Interestingly, the bisectors bounding a tile $\mathscr{T}_i$ in the power diagram are linearly displaced depending on w_i. Thus, for a uniform choice of $w_i = 0$ at all sites except $\mathbf{q}$, the corresponding weight $w_\mathbf{q} \in [-w_{\max}, 0]$, where $\mathscr{T}_\mathbf{q}(-w_{\max}) = \emptyset$, continuously blends between the ordinary Voronoi diagram of $\mathbf{X}$ and that of $\mathbf{X} \cup \{\mathbf{q}\}$, as illustrated in Figure 2(a).

These last facts led to the observation that the areas $\mathscr{T}_\mathbf{q} \cap \mathscr{T}_i$ used in the computation of Sibson's coordinates are swept by the edges of the weight-dependent tile $\mathscr{T}_\mathbf{q}(w_\mathbf{q})$ as $w_\mathbf{q}$ runs from $-w_{\max}$ to 0. This constitutes an integral relation between the variable length of the Voronoi edge and the area of overlap $\mathscr{T}_\mathbf{q} \cap \mathscr{T}_i$ that was generalized to

$$\lambda_i^k(\mathbf{q}) = \hat{\lambda}_i^k(\mathbf{q}) \;/\; \sum_j \hat{\lambda}_j^k(\mathbf{q}), \qquad \hat{\lambda}_i^k(\mathbf{q}) = \hat{\lambda}_i^k(\mathbf{q}, 0), \tag{4}$$

$$\hat{\lambda}_i^k(\mathbf{q}, u) = \int_{-\infty}^{0} \hat{\lambda}_i^{k-1}(\mathbf{q}, v)\mathrm{d}v, \qquad \hat{\lambda}_i^0(\mathbf{q}, u) = l_i(\mathbf{q}, u)/r_i,$$

where $l_i(\mathbf{q}, u)$ is the length of the tile edge separating $\mathscr{T}_\mathbf{q}(u)$ and $\mathscr{T}_i(u)$, and $r_i = \|\mathbf{x}_i - \mathbf{q}\|$.

Figure 2(a) shows that $l_i(\mathbf{q}, u)$ is a piecewise linear function, making $\hat{\lambda}_i^k(\mathbf{q}, u)$ a piecewise polynomial on support intervals that are determined by the geometry of the Voronoi diagram. The implementation of the integral expression (4) involves rather complex geometric operations and requires careful treatment of degenerate cases.

In [BBU06] the authors devised an algebraic approach to the determination of $l(\mathbf{q}, u)$ that naturally deals with degenerate situations and generalizes more easily to 3D. A sketch of the approach follows.

The weight-dependent tile $\mathscr{T}_\mathbf{q}(u)$ in our consideration is always a convex polytope which has an alternate representation as an intersection of half-spaces whose representation in Hessian normal form can easily be derived from the Delaunay triangulation,

$$\mathscr{T}_\mathbf{q}(u) = \bigcap_{\mathbf{x}_i \in N_\mathbf{q}} H_i(u), \qquad H_i(u) = \{\mathbf{p} \mid (\mathbf{p} - \mathbf{q})^T(\mathbf{x}_i - \mathbf{q}) \le b_i(u)\}, \tag{5}$$

where $b_i(u)$ is a linear function.

In [Las83] a recursive algorithm was proposed for the computation of the volume of convex polytopes in half-space representation such as (5). Each recursion expresses the d-dimensional volume as a function of $d-1$-dimensional volumes, until only intersections of real-valued intervals are to be computed at the deepest level. We exploit this by applying the recursive algorithm to (3) while keeping track of the modifications done to the linear functions $b_i(u)$. After $d-1$ recursions, the 1-dimensional volumes correspond to $l_i(\mathbf{q}, u)$.

For $\mathscr{T}_\mathbf{q}(u) \in \mathbf{R}^2$, this leads to a linear programming problem whose solution corresponds to the piecewise linear function required for the computation of $\lambda^k(\mathbf{q})$ in (4), and can still

(a) (b) (c)

Figure 3: (a) Data sample from $\cos(\|\mathbf{x}\|)$. (b) Hiyoshi's global C^2 interpolant based on gradients and Hessians estimated from values in the natural neighborhood of each site, (c) the gradients at the natural neighbors were fitted in a first stage and taken into account when fitting the Hessians.

be solved in a decent fashion for $\mathscr{T}_{\mathbf{q}}(u) \in \mathbf{R}^3$, where this time roots of second order polynomials are involved.

4.3 Natural neighbor interpolant with second order continuity

Based on the C^k coordinate construction and the Bézier simplex idea introduced in [Far90], a globally C^2 continuous interpolation scheme was proposed in [HS04], representing φ by a quintic Bézier simplex b^5 over $\lambda^2(\mathbf{q})$. While the C^1 approach in [Far90] interpolates given gradients at the nodes, the C^2 approach in [HS04] additionally interpolates Hessians. The evaluation of the interpolant is rather costly in terms of floating point operations, yet the quality of the results is very high.

However, if only function values are provided as input, $\nabla f(\mathbf{x}_i)$ and $\mathscr{H} f(\mathbf{x}_i)$ must be estimated from the data. To estimate $\nabla f(\mathbf{x}_i)$, [Sib81] used Sibson's coordinates as weights for the least squares plane through the natural neighbors, yielding good results and the reproduction of spherical quadratics (see also [Flö03]). If also second order derivative information shall be extracted from the data, the set $N_{\mathbf{q}}$ is of insufficient size, which renders the estimation unstable most of the time as shown in Figure 3(b). We now sketch the recursive scheme for the estimation of higher order derivatives devised by the authors in [BBU06].

Starting with a first iteration over the data, intermediate gradients $\hat{\nabla} f(\mathbf{x}_i)$ are fitted based on each site's natural neighborhood. The second iteration fits $\nabla f(\mathbf{x}_i)$ and $\mathscr{H} f(\mathbf{x}_i)$, which are the first two terms of the Taylor expansion of f at $\mathbf{x}_i$, to the $f(\mathbf{x}_i)$ and $\hat{\nabla} f(\mathbf{x}_i)$ such that both are approximated as closely as possible. The result of this improved method is shown in Figure 3(c). The approach readily generalizes to higher order derivative estimation.

5 Discrete Harmonic Functions in Time-Dependent Point Sets

We now turn our attention to the approximation of harmonic functions. By definition, a function f is harmonic on a domain Ω if it satisfies the Laplace equation $\Delta f|_\Omega = 0$, where $\Delta f = \nabla^2 f$. In case of a boundary value problem with Dirichlet boundary conditions of the form $f|_{\partial\Omega} = u$ for a given u, a fundamental result from harmonic function theory states that there exists a unique f satisfying the Laplace equation, c.f. [ABR01].

5.1 Discrete Harmonic Functions

In case of a discrete domain $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subset \Omega$, the characterization of a discrete harmonic function $f : \mathbf{X} \to \mathbf{R}$ utilizes an approximation of $\Delta f(\mathbf{x}_i)$, which is commonly modeled as a weighted sum of differences

$$\Delta f_i := \Delta f(\mathbf{x}_i) = \sum_{\mathbf{x}_j \in N_{\mathbf{x}_i}} \lambda_{ij}(\mathbf{x}_i - \mathbf{x}_j),$$

where $N_{\mathbf{x}_i}$ is some nearby neighborhood and $\lambda_{ij} \in \mathbf{R}^+$ reflect the approximation of the metric. This is a generalization of the sum of second derivatives in finite differences.

This definition directly corresponds to that of the graph Laplacian on a directed, weighted graph $G = (\mathbf{X}, E, \Lambda)$ over the nodes $\mathbf{X}$, with edges $E = \{e_{ij}\}_{ij} \subset \mathbf{X} \times \mathbf{X}$ and edge weights $\Lambda : E \to \mathbf{R}$, where $\lambda_{ij} = \Lambda(e_{ij})$. Under the assumption that the weights Λ are positive and G is connected, a similar statement about existence and uniqueness of a discrete harmonic function exists for a set of fixed function values.

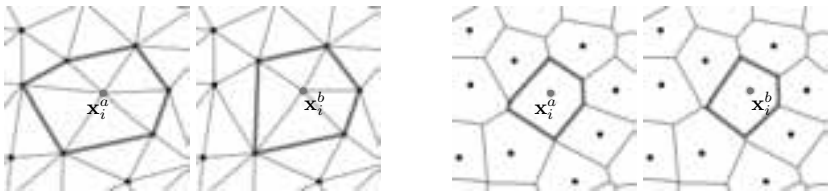
In order to approximate the continuous Laplacian, a reasonable choice of E are the edges in the Delaunay triangulation of $\mathbf{X}$. This leaves the choice of λ_{ij}, which can be chosen such that the discrete Laplacian reproduces certain properties of the continuous Laplacian, namely $\Delta \mathrm{id} = 0$, which translates component-wise into

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

This is trivially fulfilled if λ_{ij} are chosen to be generalized barycentric coordinates of $\mathbf{x}_i$ in the one-ring neighborhood of the Delaunay triangulation, since by definition of barycentric coordinates,

$$0 = \sum_{e_{ij} \in E} \lambda_{ij}(\mathbf{x}_i - \mathbf{x}_j), \qquad 1 = \sum_{e_{ij} \in E} \lambda_{ij}, \qquad 0 \leq \lambda_{ij}.$$

Obviously, λ_{ij} is only defined in the interior of the convex hull $\mathscr{C}(\mathbf{X})$ of $\mathbf{X}$. Therefore, the Dirichlet conditions of the boundary value problem are automatically required at convex hull vertices of $\mathbf{X}$.



(a) (b)

Figure 4: (a) The enclosing one-ring polygon of a vertex in the Delaunay triangulation in before and after an edge flip caused by the motion of $\mathbf{x}_i^a$ to $\mathbf{x}_i^b$. (b) The same constellation in the Voronoi diagram. Note the difference in the amount of change that happens in (a) and (b).

With this in hand, the discrete harmonic function that is the solution to the boundary value problem

$$f|_{\partial\mathscr{C}(\mathbf{X})} = u, \qquad \Delta f_i = 0$$

is found as the solution of a sparse linear system.

When ignoring the limit behavior of the above introduced Laplacian approximation the authors confirmed in [BHFU07] that the computed discrete harmonic functions approximate continuous harmonic functions very well.

5.2 Deforming Domain

If the domain discretization $\mathbf{X}$ is now a continuous function of time $\mathbf{X}(t)$, say in the course of a dynamic simulation, then we intuitively expect the Laplacian approximation and with it the computed discrete harmonic function to continuously reflect the deformation that occurs with time.

The authors investigated this aspect in [BHFU07]. It turns out that among the many choices of barycentric coordinates that render acceptable approximations of the Laplacian in the static case, only natural neighbor coordinates lead to a Laplacian approximation that continuously depends on the deformation $\mathbf{X}(t)$.

The reason for this is the continuity of local coordinates with respect to their defining neighborhood. The majority of generalized barycentric coordinates are defined with respect to an enclosing polygon, which is derived from the connectivity of the Delaunay triangulation. The connectivity, however, must change at some point of an arbitrary deformation for any triangulation to remain valid. These changes lead to discontinuous changes of the polygonal neighborhood, and consequently to discontinuous jumps in generalized polygonal barycentric coordinates, which is illustrated in Figure 4(a).

Natural neighbor coordinates, on the other hand, are zero exactly when edge flips occur in the Delaunay triangulation, which comes from their relation to sizes of entities in the Voronoi diagram which themselves continuously depend on the positions of the Voronoi

sites. This behavior is illustrated in Figure 4(b). Furthermore, the continuity of λ_{ij} with respect to $\mathbf{q}$, as discussed in Section 4, carries over to the Laplacian approximation.

6 Conclusion

We motivated that for scattered data methods in a spatial context, the definition of "neighborhood" plays an important role. Without prior knowledge about the data, spatial proximity is a valid criterion for neighborhood definition. A completely automatic determination of such a neighborhood is given in terms of natural neighbors in the Voronoi diagram of a set of data sites.

By pointing out the advantages and algorithmic implications of natural neighbor based methods we provided insight into key indicators for their application.

This has been further supported by sketching the robust implementation of C^2 natural neighbor interpolation, derivative estimation and the application of natural neighbor coordinates to the computation of discrete harmonic functions.

Acknowledgements Financial support by the German Research Foundation DFG within the International Research Training Group 1131 'Visualisation of Large and Unstructured Data Sets. Applications in Geospatial Planning, Modeling, and Engineering' is gratefully acknowledged by the authors.

References

[ABR01] S. Axler, P. Bourdon, and W. Ramey. *Harmonic Function Theory*. Springer, 2nd edition, 2001.

[AMG98] Francois Anton, Darka Mioc, and Christopher Gold. Local coordinates and interpolation in a Voronoi diagram for a set of points and line segments. In *Proceedings of the 2nd Voronoi Conference on Analytic Number Theory and Space Tillings*, pages 9–12, 1998.

[Aur91] Franz Aurenhammer. Voronoi Diagrams - A survey of a fundamental geometric data structure. *ACM Computing surveys*, 23(3):345–405, 1991.

[BBU06] T. Bobach, M. Bertram, and G. Umlauf. Issues and Implementation of C^1 and C^2 Natural Neighbor Interpolation. In *Proceedings of the* 2^{nd} *International Symposium on Visual Computing*, Nov. 2006.

[BC00] J. Boissonnat and F. Cazals. Natural neighbour coordinates of points on a surface. Technical Report 4015, INRIA-Sophia., 2000.

[BHFU07] T. Bobach, D. Hansford, G. Farin, and G. Umlauf. Discrete Harmonic Functions from Local Coordinates. In *Mathematics of Surfaces XII*, 2007.

[BIK+97] V. V. Belikov, V. D. Ivanov, V. K. Kontorovich, S. A. Korytnik, and A. Yu. Semenov. The Non-Sibsonian interpolation: A new method of interpolation of the values of a

function on an arbitrary set of points. *Computational Mathematics and Mathematical Physics*, 37(1):9–15, 1997.

[BS95] Jean Braun and Malcolm Sambridge. A numerical method for solving partial differential equations on highly irregular grids. *Nature*, 376:655–660, 1995.

[CFL82] N. H. Christ, R. Friedberg, and T. D. Lee. Weights of links and plaquettes in a random lattice. *Nuclear Physics B*, 210(3):337–346, 1982.

[Cla96] Kenneth L. Clarkson. Convex Hulls: Some Algorithms and Applications. Presentation at Fifth MSI-Stony Brook Workshop on Computational Geometry, 1996.

[dB87] Carl de Boor. B-form basics. *Geometric Modelling - Algorithms and New Trends*, pages 131–148, 1987.

[Far90] Gerald Farin. Surfaces over Dirichlet Tessellations. *Computer Aided Geometric Design*, 7:281–292, Jun 1990.

[FEK+05] Quanfu Fan, Alon Efrat, Vladlen Koltun, Shankar Krishnan, and Suresh Venkatasubramanian. Hardware-assisted Natural Neighbor Interpolation. In *Proc. 7th Workshop on Algorithm Engineering and Experiments (ALENEX)*, 2005.

[Flö03] Julia Flötotto. *A coordinate system associated to a point cloud issued from a manifold: definition, properties and applications*. PhD thesis, Université de Nice-Sophia Antipolis, Sep 2003. http://www.inria.fr/rrrt/tu-0805.html.

[GF99] Lee Gross and Gerald E. Farin. A transfinite form of Sibson's interpolant. *Discrete Applied Mathematics*, 93:33–50, 1999.

[HCK+99] K. Hoff, T. Culver, J. Keyser, M. Lin, and D. Manocha. Fast Computation of Generalized Voronoi Diagrams using Graphics Hardware. In *Proceedings of ACM SIGGRAPH 1999*, 1999.

[Hiy05] Hisamoto Hiyoshi. Stable computation of natural neighbor interpolation. In *Proceedings of the* 2^{nd} *International Symposium on Voronoi Diagrams in Science and Engineering*, pages 325–333, Oct. 2005.

[HS00a] Hisamoto Hiyoshi and Kokichi Sugihara. An Interpolant Based on Line Segment Voronoi Diagrams. In *JCDCG*, pages 119–128, 2000.

[HS00b] Hisamoto Hiyoshi and Kokichi Sugihara. Voronoi-based interpolation with higher continuity. In *Symposium on Computational Geometry*, pages 242–250, 2000.

[HS04] Hisamoto Hiyoshi and Kokichi Sugihara. Improving the Global Continuity of the Natural Neighbor Interpolation. In *ICCSA (3)*, pages 71–80, 2004.

[Las83] J. B. Lasserre. An analytical expression and an algorithm for the volume of a convex polytope in R^n. *Journal of Optimization Theory and Applications*, 39(3):363–377, 1983.

[OBSC00] Atsuyuki Okabe, Barry Boots, Kokichi Sugihara, and Sung Nok Chiu. *Spatial Tessellations: Concepts and applications of Voronoi diagrams*. Wiley series in probability and statistics. John Wiley & Sons Ltd, 2000.

[Pip92] Bruce R. Piper. Properties of Local Coordinates Based on Dirichlet Tesselations. In *Geometric Modelling*, pages 227–239, 1992.

[PLK+06] Sung W. Park, Lars Linsen, Oliver Kreylos, John D. Owens, and Bernd Hamann. Discrete Sibson Interpolation. In *IEEE Transactions on Visualization and Computer Graphics 12*, volume 2, pages 243–253, 2006.

[Sib80] R. Sibson. A vector identity for the Dirichlet tessellation. *Mathematical Proceedings of Cambridge Philosophical Society*, 87:151–155, 1980.

[Sib81] R. Sibson. A brief description of natural neighbor interpolation. *Interpreting Multivariate Data*, pages 21–36, 1981.

[Sug99] Kokichi Sugihara. Surface interpolation based on new local coordinates. *Computer Aided Design*, 13(1):51–58, 1999.

[Wat92] David F. Watson. *Contouring - A guide to the analysis and display of spatial data*. Pergamon, 1st edition, 1992.

Comparative Tensor Visualisation within the Framework of Consistent Time-Stepping Schemes

R. Mohr[1], T. Bobach[2], Y. Hijazi[2], G. Reis[2], P. Steinmann[1], and H. Hagen[2]

[1] University of Kaiserslautern
Chair of Applied Mechanics
Department of Mechanical and Process Engineering
D-67653 Kaiserslautern, Germany
http://mechanik.mv.uni-kl.de

[2] University of Kaiserslautern
Computer Graphics Group
Department of Computer Sciences
D-67653 Kaiserslautern, Germany
http://www-hagen.informatik.uni-kl.de

Abstract: Nowadays, the design of so-called consistent time-stepping schemes that basically feature a physically correct time integration, is still a state-of-the-art topic in the area of numerical mechanics. Within the proposed framework for finite elasto-plasto-dynamics, the spatial as well as the time discretisation rely both on a Finite Element approach and the resulting algorithmic conservation properties have been shown to be closely related to quadrature formulas that are required for the calculation of time-integrals. Thereby, consistent integration schemes, which allow a superior numerical performance, have been developed based on the introduction of an enhanced algorithmic stress tensor, compare [MMS06]-[MMS07c].
In this contribution, the influence of this consistent stress enhancement, representing a modified time quadrature rule, is analysed for the first time based on the spatial distribution of the tensor-valued difference between the standard quadrature rule, relying on a specific evaluation of the well-known continuum stresses, and the favoured non-standard quadrature rule, involving the mentioned enhanced algorithmic stresses. This comparative analysis is carried out using several visualisation tools tailored to set apart spatial and temporal patterns that allow to deduce the influence of both step size and material constants on the stress enhancement. The resulting visualisations indeed confirm the physical intuition by pointing out locations where interesting changes happen in the data.

1 Motivation

It is well-known in literature that the performance of classical time integration schemes for structural dynamics, as for instance developed in [New59], is strongly limited when dealing with highly nonlinear systems. In a nonlinear setting, sophisticated numerical techniques are required to satisfy the classical balance laws, as for instance balance of

linear and angular momentum or the classical laws of thermodynamics. Nowadays, energy and momentum conserving time integrators for dynamical systems, like multibody systems or elasto-dynamics, are well-established in the computational dynamics community, compare e.g. [ST92]. In contrast to the commonly used time discretisation based on Finite Differences, one-step implicit integration algorithms relying on Finite Elements in space and time were developed, for instance, in Betsch and Steinmann [BS01]. Therein, conservation of energy and angular momentum have been shown to be closely related to quadrature formulas required for numerical integration in time. In this context, specific algorithmic energy conserving schemes for hyperelastic materials can be based on the introduction of an enhanced stress tensor for time shape functions of arbitrary order, compare Gross *et al.* [GBS05]. Recently, a generalisation of these Galerkin-based concepts to finite elasto-plasto-dynamics has been worked out by Mohr *et al.* [MMS06]-[MMS07c].

However, it has been shown by many authors that the introduction of a modified stress tensor represents an appropriate tool to design specific conserving respectively consistent time-stepping schemes, compare e.g. [Arm06, Gonz00, GBS05, ML02a, MMS07c, NSP06]. Nevertheless, in our opinion the influence of this stress enhancement is not completely understood yet. One very interesting aspect that has not been addressed in the literature so far, is for instance the spatial distribution of the difference tensor between the stresses of the continuum model and the enhanced stresses for the time-stepping. In this context, we have already encouraged some basic discussions in [MMS06] based on an 'ad hoc' visualisation approach that provides very limited information. In this contribution, several more sophisticated techniques that offer different levels of detail regarding the included information, have been developed to visualise the difference between both second-order tensor fields. It will be demonstrated by means of representative parameter studies that the proposed concepts indeed represent an effective tool to better understand the numerical behaviour of the underlying time-stepping scheme.

2 Finite Elasto-Plasto-Dynamics

First, the nonlinear deformation map $\boldsymbol{\varphi}(\boldsymbol{X},t) : \mathcal{B}_0 \times [0,T] \rightarrow \mathcal{B}_t$ is introduced as a mapping from the material to the spatial configuration, whereby $\mathcal{B}_{0/t} \subset \mathbb{R}^{2/3}$. In the context of finite plasticity, the resulting deformation gradient $\boldsymbol{F} := \nabla_{\boldsymbol{X}} \boldsymbol{\varphi}(\boldsymbol{X},t)$ is assumed to be multiplicatively decomposed into an elastic and a plastic part:

$$\boldsymbol{F} \doteq \boldsymbol{F}_e \cdot \boldsymbol{F}_p \tag{1}$$

In contrast to the modelling of elasticity, additional internal variables $\boldsymbol{\kappa}$ are included in the Helmholtz energy density $\psi(\boldsymbol{F},\boldsymbol{\kappa})$ for the plastic case to model the loading history. Moreover, it is accepted to introduce the so-called conjugated thermodynamical forces $\boldsymbol{\beta} := -\nabla_{\boldsymbol{\kappa}}\psi$ which render the dissipation inequality, namely $\mathcal{D} = \left\langle \boldsymbol{\beta}, \dot{\boldsymbol{\kappa}} \right\rangle \geq 0$. In view of a thermodynamically consistent modelling this dissipation inequality has to be respected not only by the continuum model, but also by the applied numerical integration scheme. In a next step, we apply a standard Finite Element discretisation in space for the material

configuration of a solid continuum body. Using the spatial approximations, the semi-discrete deformation map can be written by means of the spatial shape functions $N_A(\boldsymbol{X})$ in the form: $\boldsymbol{\varphi}(\boldsymbol{X},t) = \sum_{A=1}^{n_{node}} \boldsymbol{q}_A(t)\, N_A(\boldsymbol{X})$. Consequently, the approximations in space of the spatial velocity $\boldsymbol{v} := \sum_{A=1}^{n_{node}} \dot{\boldsymbol{q}}_A\, N_A$ and the right Cauchy-Green tensor

$$\boldsymbol{C} := \boldsymbol{F}^{\mathrm{t}} \cdot \boldsymbol{F} = \sum_{A,B=1}^{n_{node}} \boldsymbol{q}_A \cdot \boldsymbol{q}_B\, \nabla N_A \otimes \nabla N_B \tag{2}$$

can be computed straightforwardly. To obtain a semi-discrete system of equations of motion, we combine the placements of the spatial nodes $\boldsymbol{q} = [\boldsymbol{q}_1, ..., \boldsymbol{q}_{n_{node}}]^{\mathrm{t}}$ and the nodal generalised momenta $\boldsymbol{p} := \mathbb{M} \cdot \dot{\boldsymbol{q}} = [\boldsymbol{p}_1, ..., \boldsymbol{p}_{n_{node}}]^{\mathrm{t}}$ with the mass matrix $\mathbb{M}$ to the vector $\boldsymbol{z} := [\boldsymbol{q}, \boldsymbol{p}]^{\mathrm{t}}$. Furthermore, the sum of the kinetic energy $T(\boldsymbol{p}) = \frac{1}{2}\, \boldsymbol{p} \cdot \mathbb{M}^{-1} \cdot \boldsymbol{p}$, the free energy $\Psi = \int_{\mathcal{B}_0} \psi \, \mathrm{d}V$ and possibly an external potential V^{ext} is defined as $H(\boldsymbol{q}, \boldsymbol{p}; \boldsymbol{\kappa}) := T + \Psi + V^{ext}$. Inspired by the purely elastic case, the resulting equations of motion can still be written in a compact format of Hamilton-type

$$\dot{\boldsymbol{z}}(t) = \mathbb{J} \cdot \nabla_{\boldsymbol{z}} H(\boldsymbol{z}; \boldsymbol{\kappa}) \qquad \text{with} \qquad \nabla_{\boldsymbol{z}} H = \begin{bmatrix} \boldsymbol{F}^{int} - \boldsymbol{F}^{ext} \\ \\ \mathbb{M}^{-1} \cdot \boldsymbol{p} \end{bmatrix}, \tag{3}$$

wherein we have incorporated the symplectic matrix $\mathbb{J}$ and the internal load vector $\boldsymbol{F}^{int}(\boldsymbol{S})$, involving the Piola Kirchhoff stresses $\boldsymbol{S} = 2\, \nabla_{\boldsymbol{C}}\, \psi$. Next, the time discretisation of the semi-discrete system of equations of motion (3) is considered. We start with a decomposition of the time interval $[0, T] = \bigcup_{n=0}^{N} [t_n, t_{n+1}]$ and a map of each sub-interval to the reference time interval $[0, 1]$ via the function $\alpha(t) := [t - t_n]/h_n$ based on the time-step size $h_n = t_{n+1} - t_n$. For the approximation in time a continuous Galerkin method – abbreviated by: cG(k)-method – is applied. Therefore, the time approximations of the unknown function $\boldsymbol{z}^h = \sum_{j=1}^{k+1} M_j(\alpha)\, \boldsymbol{z}_j$ and the test function $\delta \boldsymbol{z}^h = \sum_{i=1}^{k} \widetilde{M}_i(\alpha)\, \delta \boldsymbol{z}_i$ are introduced [1]. In a compact notation the resulting weak form in time is given by

$$\int_0^1 \left[\mathbb{J} \cdot \delta \boldsymbol{z}^h \right] \cdot \left[\mathrm{D}_\alpha \boldsymbol{z}^h - h_n\, \mathbb{J} \cdot \nabla_{\boldsymbol{z}} H(\boldsymbol{z}; \boldsymbol{\kappa}) \right] \mathrm{d}\alpha = 0\,. \tag{4}$$

Obviously, Equation (4) involves time-integrated internal load vectors, which will be referred to as $\bar{\boldsymbol{F}}^{int}_{A\,i}$ related to the spatial node A. As discussed for instance in Mohr *et al.* [MMS06]-[MMS07c], the crucial aspect for the conservation properties of the resulting time-stepping schemes is the approximation of these highly nonlinear time integrals. Of course, one potential option concerning the approximation is the application of a standard Gauss quadrature rule represented by

$$\bar{\boldsymbol{F}}^{int}_{A\,i} \approx \sum_{l=1}^{n_{gpt}} \sum_{B=1}^{node} w_l\, \widetilde{M}_i(\zeta_l)\, \boldsymbol{q}^h_B(\zeta_l) \left[\int_{\mathcal{B}_0} \nabla N_A \otimes \nabla N_B : \boldsymbol{S}\, \mathrm{d}V \right] \Big|_{\zeta_l}, \tag{5}$$

[1] It is important to emphasise that the time shape functions $M_j \in \mathcal{P}^k$ are polynomials of degree k, whereas the reduced shape functions $\widetilde{M}_i \in \mathcal{P}^{k-1}$ are only of degree $k-1$.

using the Gauss points ζ_l and the Gauss weights w_l. The foregoing discretisations render a completely discrete system of equations, representing a time-stepping scheme with the following conservation properties. If we assume vanishing external loads, the resulting integration scheme allows the conservation of linear momentum as well as the conservation of angular momentum. Nevertheless, it can be shown that such a standard quadrature rule is not able to guarantee the conservation of total energy for elastic deformations, despite being an essential feature which has to be captured by the integrator regarding the claimed thermodynamical consistency. Consequently, we introduce the nonstandard quadrature rule

$$\bar{\boldsymbol{F}}^{int}_{A\,i} \approx \sum_{l=1}^{n_{gp_t}} \sum_{B=1}^{node} w_l\, \widetilde{M}_i(\zeta_l)\, \boldsymbol{q}^h_B(\zeta_l) \left[\int_{\mathcal{B}_0} \nabla N_A \otimes \nabla N_B : \boldsymbol{S}^{alg}\, \mathrm{d}V \right] \Big|_{\zeta_l} , \tag{6}$$

wherein the so-called *elastic-enhanced algorithmic* stress tensor $\boldsymbol{S}^{alg} := \boldsymbol{S} + {}^{el}\boldsymbol{S}^{enh}$ has been applied based on the enhancement

$${}^{el}\boldsymbol{S}^{enh}(\boldsymbol{S}) = 2\, \frac{\psi_{\alpha=1} - \psi_{\alpha=0} - \int_0^1 \boldsymbol{S} : \frac{1}{2}\, \mathrm{D}_\alpha \boldsymbol{C}^h \ \mathrm{d}\alpha}{\int_0^1 || \,\mathrm{D}_\alpha \boldsymbol{C}^h ||^2 \ \mathrm{d}\alpha}\, \mathrm{D}_\alpha \boldsymbol{C}^h \ . \tag{7}$$

This approach follows the enhanced Galerkin methods – or short: eG(k)-methods – that have been proposed originally by Gross *et al.* [GBS05] in the context of hyperelasticity. Based on this specific nonstandard quadrature rule, the resulting time integrators guarantees additionally a conservation of the total energy $H_{\alpha=1} - H_{\alpha=0} = 0$ when the deformation is elastic. In combination with a strictly positive dissipation in the plastic case, a monotonic decrease of the total energy $H_{\alpha=1} - H_{\alpha=0} < 0$ and, consequently, a thermodynamically consistent time-integration can apparently be featured, offering superior performance in comparison to standard integration schemes. In this context, we want to point out once more that the key to thermodynamical consistency exclusively relies on a modified approximation of the corresponding time-integrals based on the elastic-enhanced algorithmic stress tensor.

3 Comparative Tensor Visualisation

In the previous section, the essential ingredients for a thermodynamically consistent time-integration have been presented. Thereby, the crucial difference between the standard Gauss quadrature rule and the more sophisticated nonstandard quadrature rule is directly related to the tensor-valued difference between the standard stresses of the continuum model $\boldsymbol{S}$ and the algorithmic stresses $\boldsymbol{S}^{alg}$, involving the enhancement tensor (7). One interesting aspect not addressed in literature so far is the spatial distribution of the corresponding difference tensor field. We are optimistic that such a comparison between both tensor fields provides a much deeper insight into the numerical behaviour of the related time-stepping schemes. In this context important issues are for instance: the correlation between the corrections and the underlying deformation, the influence of the time-step size

or the material properties, the evolution of the corrections in time, the existence of characteristic patterns within the difference tensor field, etc. However, a satisfying visualisation is a non-trivial task, dealing with two different tensor fields and a large number of time steps. A further difficulty is the fact that a direct physical interpretation of the enhancement term ${}^{el}\boldsymbol{S}^{enh}$ and the algorithmic stress tensor $\boldsymbol{S}^{alg}$, respectively, is not valid since it represents only a numerical tool to support the quadrature rule for time-integration. In the following, we focus on the development and the comparison of various visualisation approaches to better understand the influence of the correction on the time-quadrature rule.

To generate a benchmark data set, we calculated the motion of a 'Flying L' based on 36 4-node Finite Elements in space, using linear Finite Elements in time. For further set-up details we refer to Mohr *et al.* [MMS06]. In view of the abovementioned issues, the calculations have been performed with stiff/non-stiff material properties, involving $[\lambda, \mu] = [10000, 5000]/[1000, 500]$, and with large ($h_n = 0.4$) respectively small ($h_n = 0.04$) time-step sizes. Since the considered tensor fields are both defined in the reference configuration $\mathcal{B}_0$ only the undeformed configuration is of interest and, consequently, the actual deformation of the body is not shown, compare Figures 1- 10.

Figure 1 shows a visualisation obtained with Matlab by representing the tensor with its two orthogonal, normalised eigenvectors based on the spectral decompositions

$$\boldsymbol{S}^{alg} = \sum_{i=1}^{2} {}^{\boldsymbol{S}}\lambda_i^{alg}\, \boldsymbol{N}_i^{alg} \otimes \boldsymbol{N}_i^{alg} \qquad \text{and} \qquad \boldsymbol{S} = \sum_{i=1}^{2} {}^{\boldsymbol{S}}\lambda_i\, \boldsymbol{N}_i \otimes \boldsymbol{N}_i\,. \tag{8}$$

This figure was our initial motivation for experimenting with more advanced visualisations, especially to overcome the occlusion problem. Moreover, Figure 1 shows the two tensor fields and not its difference. However, a natural possibility to reduce the complexity of information is to find an appropriate representation of the difference field, since basically the corrections are of particular interest.

Our goal is to provide multiple different visualisation tools to support the understanding of both the spatial distribution of the algorithmic enhancement terms and their effect on the stress field. Therefore we examine the data in a spatial context from different points of view, one focusing on the magnitude of numerical differences in the stress tensors $\boldsymbol{S}$ and $\boldsymbol{S}^{alg}$ such as tensor invariants, another focusing on differences in extracted entities like principal stress directions. We combine basic visualisation techniques such as colour coding, transparency effects, and scaling together in order to provide the most helpful tools, thereby applying Information Visualisation [Jac99] techniques. Very little work has been carried out for difference tensor visualisation; most results relate to the tensor visualisation itself, which already is a challenge. Some examples include the visualisation of stress and strain tensors [GGH*97, NJP05] and the visualisation of diffusion tensor MRI [WMN*02].

We analyse a particular time step from abovementioned simulation for the combination of stiff/non-stiff material and fine/coarse time resolution, resulting in four different configurations. These are shown in Figure 2, showing the principal directions $[\boldsymbol{N}_i^{alg}, \boldsymbol{N}_i]$ in [blue, red] [2] similar to Figure 1. Since principal stress directions are not oriented, we use

[2] This is replaced by a [lighter,darker] grey in the b/w print.

line segments scaled by the magnitude of the corresponding eigenvalues to show the stress distribution over the elements. The scaling has been chosen to avoid the visual clutter seen in Figure 1, yet although both stresses are visualised, the absolute difference in these intuitive representations is too small to be seen, justifying the direct analysis of difference terms as proposed in the subsequent sections.

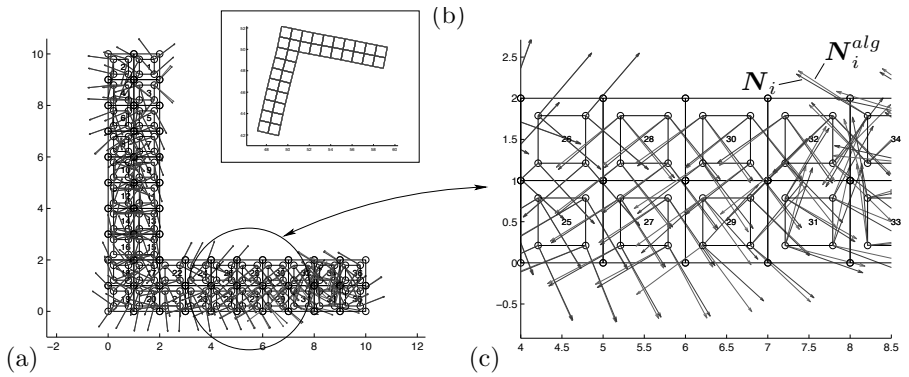


Figure 1: (a) reference configuration $\mathcal{B}_0$ with the eigenvectors $[\boldsymbol{N}_i^{alg}, \boldsymbol{N}_i]$ of the elastic-enhanced algorithmic stress tensor $\boldsymbol{S}^{alg}$ & the Piola Kirchhoff stress tensor $\boldsymbol{S}$, (b) deformed configuration $\mathcal{B}_t$ after $10s$, (c) zoom of the principal directions $[\boldsymbol{N}_i^{alg}, \boldsymbol{N}_i]$.

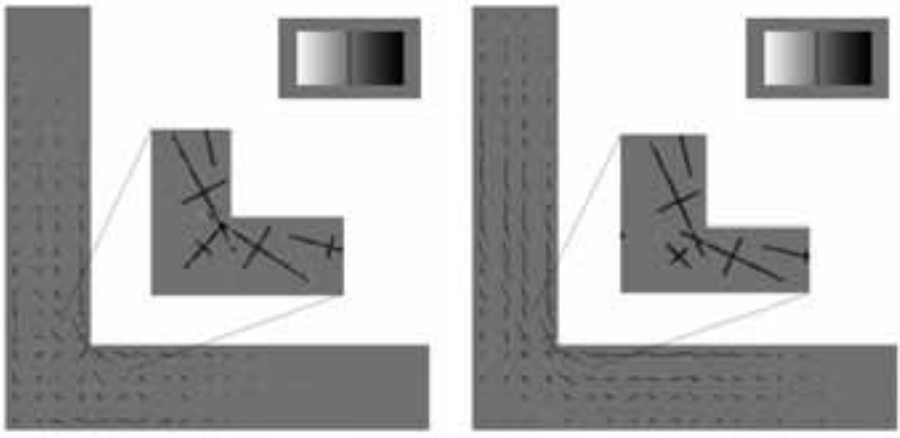
3.1 Interpreting the symmetric difference tensor field as a $3d$ vector field

We are interested in a way of representing the difference tensor field. Notice that the considered tensors are all symmetric so we have three independent components, i.e.

$$\boldsymbol{S} = \begin{bmatrix} S_{11} & S_{12} \\ S_{12} & S_{22} \end{bmatrix} \tag{9}$$

which we can represent as a $3d$ vector $s = [S_{11}\ S_{22}\ S_{12}]^{\mathrm{t}}$, similar to the classical Voigt notation in the Finite Element context.

We have chosen this approach since we find it much more intuitive to compute the difference between two vectors than computing the difference between two tensors. We then connected the $3d$ vectors of each Gauss point, four by four, to create patches resulting in a quad-patch for every calculation element. Even if the resulting patches are indeed $3d$ we find it useful to simply visualise their $2d$ projection, as it shows the deformations. Note that this type of deformation is not related to the physical deformation of the considered body. Based on this visualisation, we compensated the loss of one dimension by adding circles at each Gauss point whose radii are the Euclidean norm of the $3d$ difference vectors, namely $r = \sqrt{\Delta {S_{11}}^2 + \Delta {S_{22}}^2 + \Delta {S_{12}}^2}$. Figures 3(a,b) and 4(a,b) respectively illustrate



(a) stiff data set, $h_n = 0.4$, time step 11

(b) stiff data set, $h_n = 0.04$, time step 110

(c) non-stiff data set, $h_n = 0.4$, time step 11

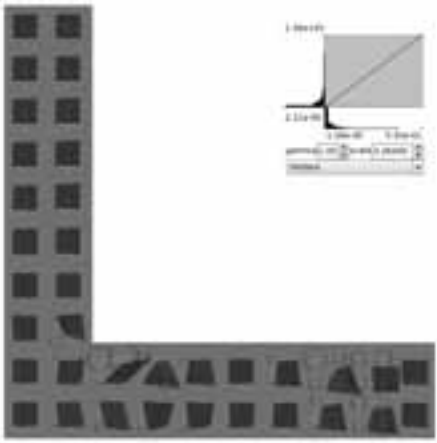
(d) non-stiff data set, $h_n = 0.04$, time step 110

Figure 2: The stress configurations in question, where the lines depict the principal stress directions scaled by the magnitude of the corresponding eigenvalues.

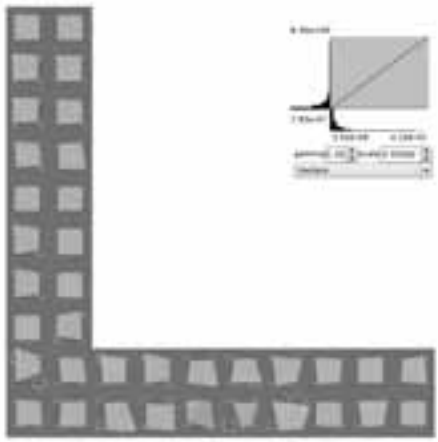
those visualisations for stiff and non-stiff data sets[3].

The results clearly demonstrate that the proposed approach is considerably well-suited to highlight regions of the body in which large corrections occur, compare Figure 3(a). Moreover, it is obvious that the corrections are higher when large time-step sizes are involved, compare e.g. Figure 3(a) and Figure 3(b).

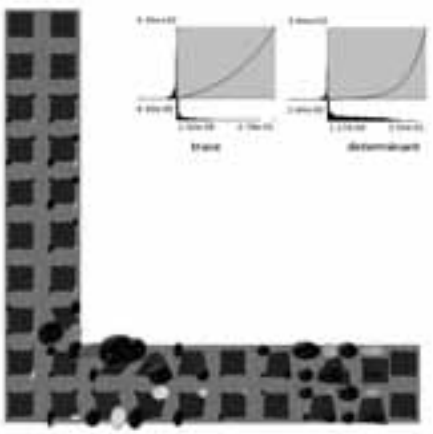
[3]Note that we used a linear scaling to avoid occlusion.



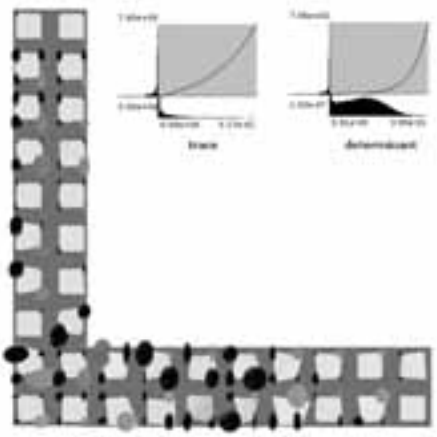
(a) stiff data set, $h_n = 0.4$, time step 11



(b) stiff data set, $h_n = 0.04$, time step 110



(c) stiff data set, $h_n = 0.4$, time step 11



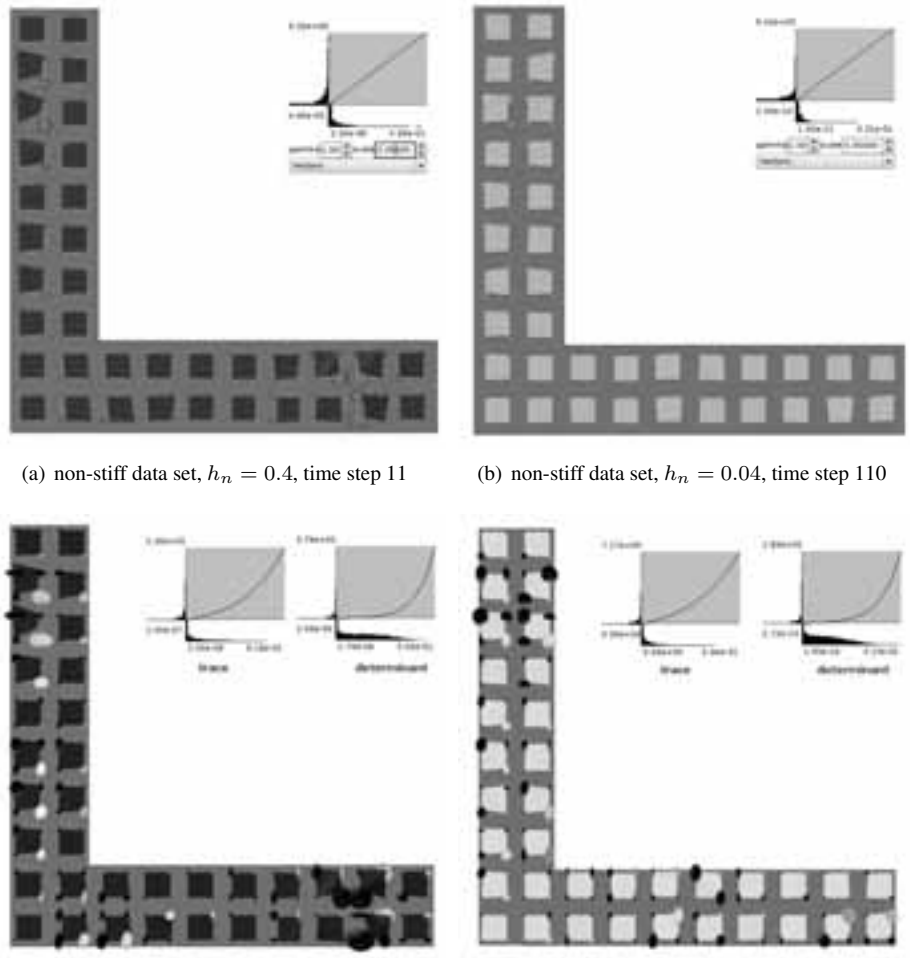
(d) stiff data set, $h_n = 0.04$, time step 110

Figure 3: Circle- and ellipsoid-based visualisation using *stiff* material properties.

3.2 Visualising the tensor invariants through ellipsoids

Another approach - more rigorous this time - consists of visualising the tensor invariants as ellipsoids. Despite looking very similar to Kindlmann's tensor glyphs [Kin04], our ellipsoids don't involve the tensor eigenvectors at all. The characteristic function of a tensor $\boldsymbol{S}$ is given by

$$\chi(\boldsymbol{S}) = |\boldsymbol{S} - \lambda \boldsymbol{I}| = \lambda^2 - [S_{11} + S_{22}]\lambda + [S_{11}S_{22} - {S_{12}}^2] \tag{10}$$



(a) non-stiff data set, $h_n = 0.4$, time step 11

(b) non-stiff data set, $h_n = 0.04$, time step 110

(c) non-stiff data set, $h_n = 0.4$, time step 11

(d) non-stiff data set, $h_n = 0.04$, time step 110

Figure 4: Circle- and ellipsoid-based visualisation using *non-stiff* material properties.

and provides two invariants, namely the trace and the determinant of the tensor:

$$I_1 = \text{tr}(\boldsymbol{S}) = S_{11} + S_{22} \quad \text{and} \quad I_2 = \det(\boldsymbol{S}) = S_{11}S_{22} - {S_{12}}^2. \tag{11}$$

The ellipsoid is built using the components' basis

$$(x, y, z) = (\Delta I_1, \Delta I_2, \frac{\Delta I_1 + \Delta I_2}{2}), \tag{12}$$

where ΔI_i is the difference between the invariants of both tensor fields.

Figures 3(c,d) and 4(c,d) respectively illustrate the ellipsoid-based visualisations for data sets that have been calculated by means of stiff and non-stiff material properties. Moreover, Figure 5 shows the evolution of the differences between the invariants over time. To investigate a potential correlation between the deformation and the corrections, the norm of the physical strain field based on the right Cauchy-Green strain tensor $\boldsymbol{C}$ has been additionally incorporated, where the following colour-coding has been used: from blue [lighter] to red [darker] for increasing strain norms. Note that we used a logarithmic scaling here, as opposed to a linear scaling, since the differences are much greater than in the circle-based visualisation[4]. In comparison to the previous approach, the corresponding plots provide an essentially better view on the spatial distribution of the corrections, since the regions with extremely large corrections are not so dominant due to the mentioned logarithmic scaling. In this context, it becomes obvious that the locations of the corrections are, especially for the stiff data set shown in Figure 3(c,d), more homogeneously distributed when a smaller time-step size is applied. Also very interesting is the clustering of large corrections in certain regions of the 'L' particularly where the norm of the strains is high, as pictured in Figure 5.

3.3 Interpreting the differences of physical measures

So far, focus has been on the display of abstract measures derived from numerical representations of the stress tensors that might be hard to interpret. We now turn our attention to changes in more physically motivated measures given by the eigenvectors $[\boldsymbol{N}_i^{alg}, \boldsymbol{N}_i]$, and eigenvalues $[{}^{\boldsymbol{S}}\lambda_i^{alg}, {}^{\boldsymbol{S}}\lambda_i]$ of both stress tensors $[\boldsymbol{S}^{alg}, \boldsymbol{S}]$, as already introduced in Equation (8). From a mechanical point of view, such a spectral decomposition is quite intuitive, since the results can be interpreted as principal stresses and principal stress directions respectively. Nevertheless, keep in mind that a direct physical interpretation of the algorithmic stress tensor by itself is critical. However, the actual goal is to show how the stress distribution is modified by the introduction of the correction term, related to Equation (7). We therefore consider rotation of the principal stress directions, and changes in the stress magnitude along these, whereby particularly the rotation seems to be an interesting issue in this context.

The computation of a rotation angle α_i between the principal directions requires the association of each direction in $\boldsymbol{N}_i^{alg}$ with a direction in $\boldsymbol{N}_i$. This can be based either on the minimisation of the variation in the corresponding eigenvalues or the minimisation of the angle between the directions. If these two criteria don't agree, say, both major eigenvalues are close but the major principal direction of the enhanced tensor aligns with the minor principal direction of the untouched tensor, the choice is arbitrary and leads to different magnitudes in angle or principal stress. In our considerations, we favour closely aligned principal directions over similar eigenvalues, where the largest eigenvalue of $\boldsymbol{S}$ determines the major principal direction. Finally, the difference in the absolute eigenvalues amounts to the root-mean-square error between the ordered pair of absolute eigenvalues in both the

[4]The redistribution caused by the logarithmic transfer function can be seen in the histograms displayed in the top-right corner.

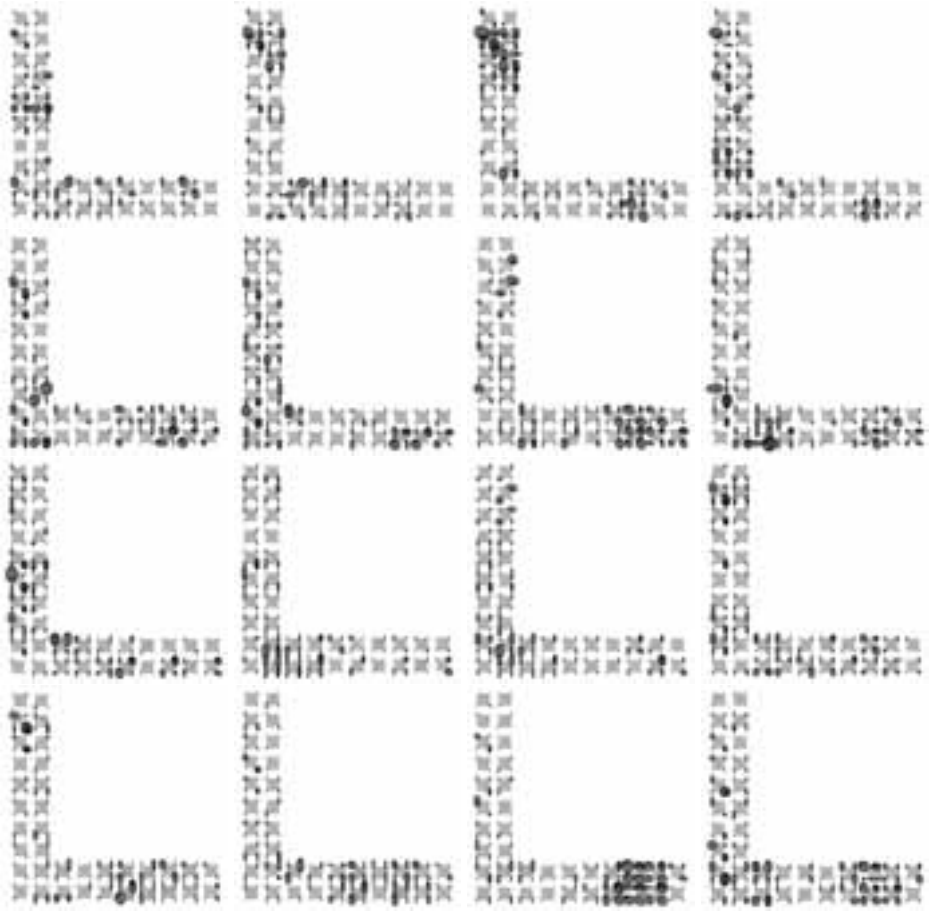


Figure 5: Ellipsoid-based visualisation over time. From top-left to bottom-right: time step 220 to 235.

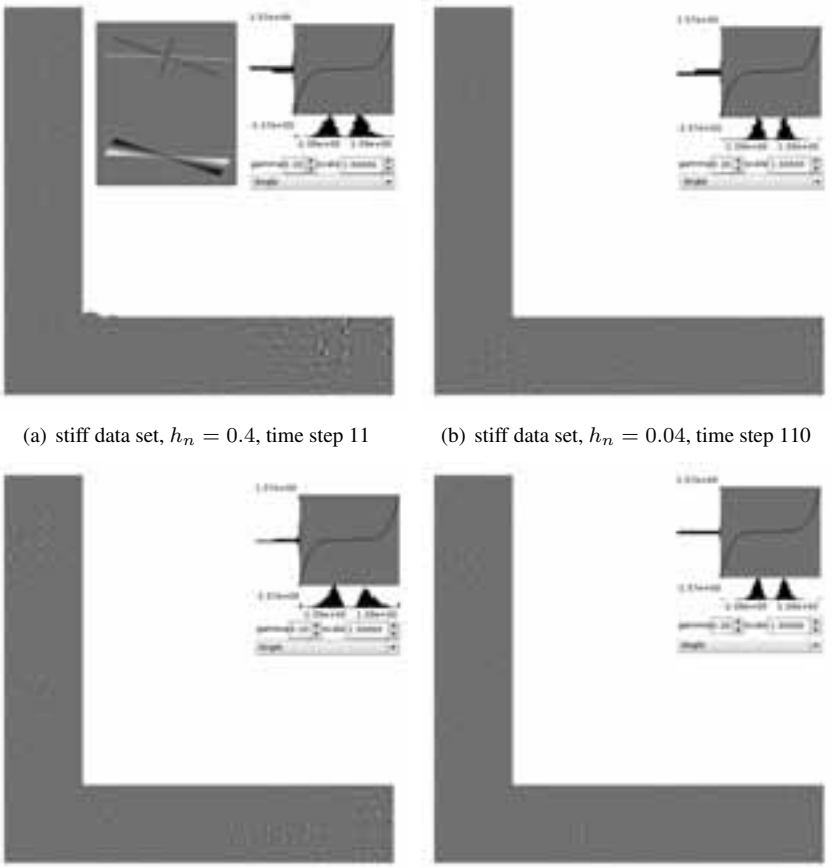
enhanced and the basic stresses, rendering

$$\Delta\lambda := \left[(|^{\boldsymbol{S}}\lambda_1^{alg}| - |^{\boldsymbol{S}}\lambda_1|)^2 + (|^{\boldsymbol{S}}\lambda_2^{alg}| - |^{\boldsymbol{S}}\lambda_2|)^2 \right]^{\frac{1}{2}} . \tag{13}$$

3.3.1 Wedges as rotation indicators

As seen in Figure 2, the angular difference α_i between $[\boldsymbol{N}_i^{alg}, \boldsymbol{N}_i]$ is too small for direct visualisation purposes. We therefore propose an exaggerated display of those using wedges that indicate the rotation direction with a colour sweep as seen in Figures 6(a-d). Prior to display, the computed angles are equalised based on the transfer function

$$\tilde{\alpha}_i = \text{sign}(\alpha_i) \ \cdot \ \lambda|\alpha_i|^\gamma, \qquad \lambda, \gamma \in \mathbb{R}^+ . \tag{14}$$



(a) stiff data set, $h_n = 0.4$, time step 11

(b) stiff data set, $h_n = 0.04$, time step 110

(c) non-stiff data set, $h_n = 0.4$, time step 11

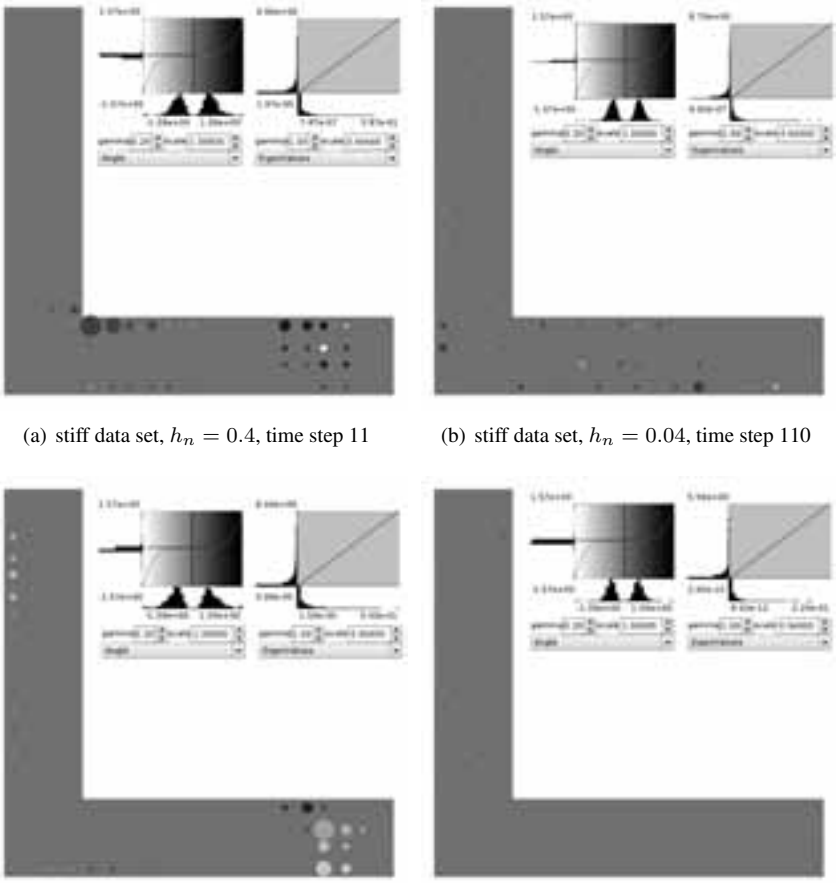
(d) non-stiff data set, $h_n = 0.04$, time step 110

Figure 6: Wedge visualisation for *stiff* and *non-stiff* data at different time resolutions. The angles have been scaled such as to provide an exaggerated display of the qualitative behaviour.

Note that a choice of $\gamma = 1$ corresponds to linear scaling by λ, while for $\gamma < 1$, stretching occurs in the interval $[-1, 1]$ - which is a reasonable behaviour as $\alpha_i \in [-\pi/4, \pi/4]$. In the present example a choice of $\lambda = 1$ and $\gamma = 0.2$ led to the best results, the corresponding equalised angle distribution is shown in the top-right corner.

The visualisation in Figures 6(a-d) employs a uniform scaling of the major eigenvectors, because the screen space occupied by one such glyph directly correlates with the user-perceived importance. As the main objective is to indicate rotation, the change in the magnitude of eigenvalues has been mapped linearly to circles, which we found least distracting from the main visualisation goal.

Once more, the results confirm the fundamental influence of the applied time-step size: a



(a) stiff data set, $h_n = 0.4$, time step 11

(b) stiff data set, $h_n = 0.04$, time step 110

(c) non-stiff data set, $h_n = 0.4$, time step 11

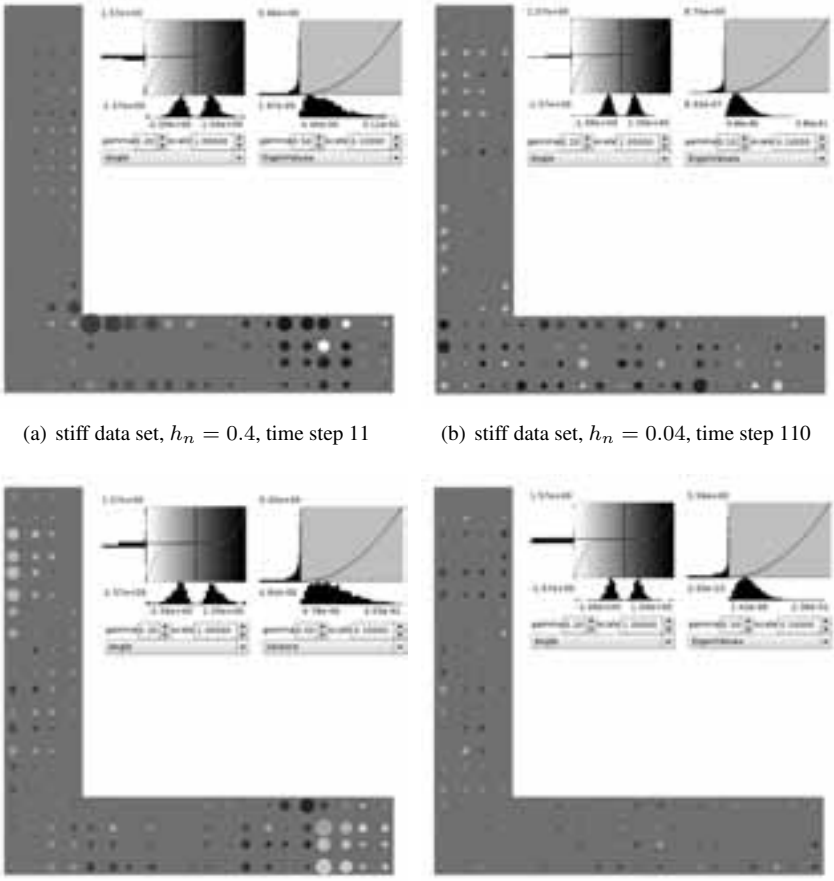
(d) non-stiff data set, $h_n = 0.04$, time step 110

Figure 7: Colour-coded discs displaying the difference in the eigenvalues via diameter and angle modification via blue ($\alpha_i < 0$) and red ($\alpha_i > 0$) colour. In the b/w print, this is replaced by [lighter,darker] grey.

larger time-step size requires also larger corrections. Moreover, we can see that in some elements large modifications of the angle are combined with small changes in the eigenvalues and vice versa, compare Figure 6(a). Analogously to Figure 3(a,b) and Figure 4(a,b), it is obvious that for the present example the largest modifications are needed when stiff material properties are combined with large time-step sizes.

3.3.2 Colour-coded discs as indicators of change in eigenvalue magnitude

The wedge-based visualisation obviously lacked clarity with respect to the rotation direction due to the small size of the glyphs. A simple yet very effective visualisation of



(a) stiff data set, $h_n = 0.4$, time step 11

(b) stiff data set, $h_n = 0.04$, time step 110

(c) non-stiff data set, $h_n = 0.4$, time step 11

(d) non-stiff data set, $h_n = 0.04$, time step 110

Figure 8: The same setting as in Figure 7, but now equalising the eigenvalues for all data sets based on the same transfer function. All but (b) exhibit better visible details.

sign and magnitude is achieved by colour-coding, where blue indicates negative rotation and red positive[5], see Figures 7 and 8. Here again an equalisation of the value range is necessary to address nuances in the angle distribution close to the origin.

We aim as a second aspect for the depiction of differences in the tensor eigenvalues, corresponding to the difference in the principal stress magnitude along $[\boldsymbol{N}_i^{alg}, \boldsymbol{N}_i]$. Figure 7 displays coloured discs with linearly scaled differences as radii. As can be seen in Figure 7(a,c), regions with dominant corrections can be immediately captured, especially for the large time-step size. However, the resulting visualisation only indicates changes in few tensors because of the large variation in magnitude of the displayed values, therefore

[5]This is replaced by a [lighter,darker] pattern in the b/w print.

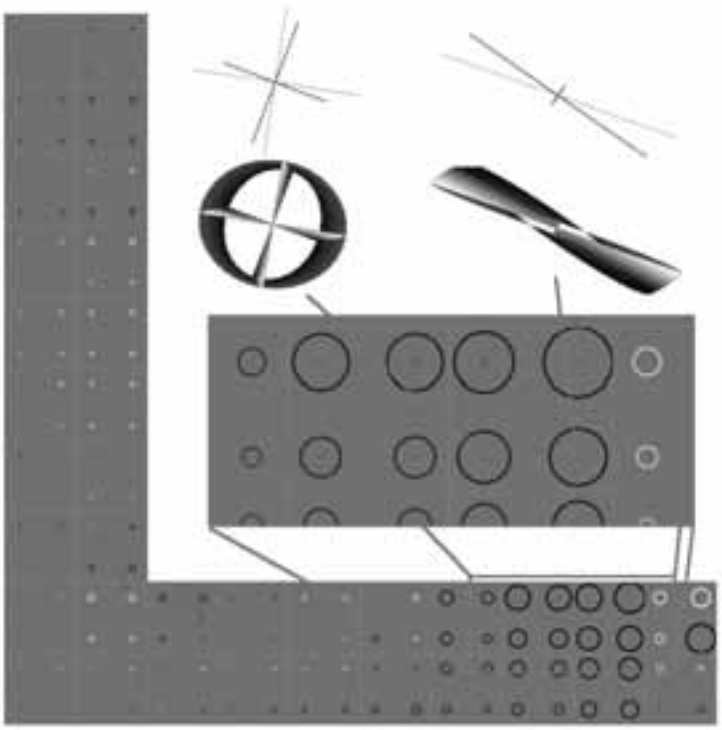


Figure 9: Level-of-detail visualisation, showing more qualitative facts at a large scale while depicting the concrete deformation on a per-tensor basis after an appropriate zoom. The pixel clutter visible in the angle of the 'L' is actually a swept ellipse similar to the one depicted in the second top row, yet very big and very thin (reproduced in colour on p. 189).

loosing more subtle details to the scaling.

This is taken care of by again equalising the value range as shown in Figure 8. The respective transfer functions are displayed in the top right corner of each figure. It can be seen in Figure 8(b) that the equalisation, if not properly scaled, obfuscates details as the result of a too uniform value distribution. Nevertheless, especially Figure 8 (a,c) enables a very interesting perspective on the spatial distribution of the modifications. Obviously, homogeneously coloured 'correction clusters' occur within the 'L', similar to the results of the ellipsoid-based visualisation.

3.3.3 Level-of-detail investigation using complex glyphs

The last approach we present is based on a multiresolution paradigm, providing qualitative information at a global scale while allowing to look at concrete behaviour in detail at an adequate zoom level. The large scale visualisation in Figure 9 combines the difference in

eigenvalues via the circle diameters, and the angle and direction of rotation via the circle colours. The detail level depicts the magnitude and angle modification as a sweep between individual representations of $\boldsymbol{S}^{alg}$ and $\boldsymbol{S}$.

The employed glyphs are created in the following way. Each pair $[\boldsymbol{N}_i^{alg}, \boldsymbol{N}_i]$ of principal directions, scaled by the eigenvalues $[{}^{\boldsymbol{S}}\lambda_i^{alg}, {}^{\boldsymbol{S}}\lambda_i]$, gives raise to a cross of two orthogonal lines. The eigenvalues are also the radii of an ellipse aligned with the principal directions. If different colours for the combination of cross and ellipse are assigned for $\boldsymbol{S}^{alg}$ and $\boldsymbol{S}$, the linear sweep between the corresponding curves provides the complex glyph visualisation in Figure 9.

This visualisation tool indeed incorporates most of the advantages of the different techniques that have been discussed before. It allows, on the one hand, an excellent detection of regions in which large modifications occur. On the other hand, a detailed physical-based insight can be obtained by zooming-in, offering information both on magnitude and rotation. The motivation for nested visualisation can be verified by looking at Figure 10, which demonstrates that no single visualisation is usually capable of giving sufficient insight into all interesting aspects present in the data.

4 Conclusions & Outlook

In the first part of this paper, we have presented the essential ingredients for a thermodynamically consistent time-stepping scheme for finite elasto-plasto-dynamics, whereby the conservation properties are directly related to the approximation of related time-integrals. In this context, a modified quadrature rule has been applied based on a so-called elastic-enhanced algorithmic stress tensor. In the second part, special emphasis has been placed on the investigation of the spatial distribution of the resulting difference between the stresses of the continuum model and the enhanced stresses for the time-stepping. Thereby, it has been shown in previous work that an 'ad hoc' visualisation is not able to provide satisfying information. Therefore, we have devised visualisations of both abstract and physically based measures in the spatial context of the simulated domain. The results help revealing the intrinsic qualities of the data, especially by pointing out regions of interest. Indeed, the developed visualisation approaches provide a deeper insight into the numerical behaviour of the algorithmic stress tensor and, consequently, enable a better understanding of the discussed integration algorithms.

In future work, the discussed results, like influence of the time-step size or clustering of the corrections, should be verified for further data sets. Moreover, we plan to incorporate the time dimension, looking at the evolution of the corrections based on the here proposed visualisation techniques. Thereby, especially the question of time continuity of the difference between both tensor fields seems to be essential.

Acknowledgements Financial support by the German Research Foundation DFG within the International Research Training Group 1131 'Visualisation of Large and Unstructured Data Sets. Applications in Geospatial Planning, Modeling, and Engineering' is gratefully acknowledged by the authors.

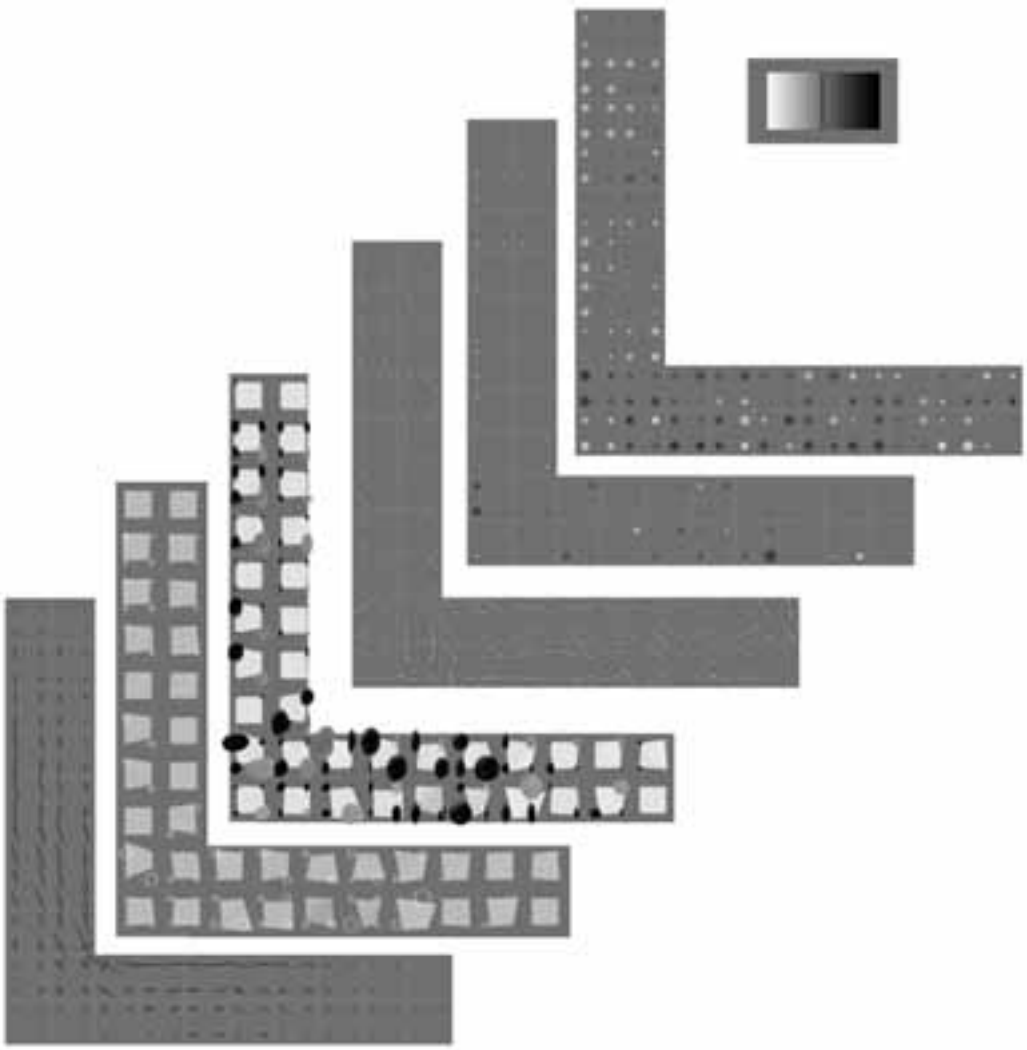


Figure 10: A collection of all presented visualisation tools applied to the same data set and time step (reproduced in colour on p. 190).

References

[Arm06] F. Armero. Energy-dissipative momentum-conserving time-stepping algorithms for finite strain multiplicative plasticity. *Computer Methods in Applied Mechanics and Engineering*, 195:4862–4889, 2006.

[BS01] P. Betsch, P. Steinmann. Conservation properties of a time FE method. Part II: Time-stepping schemes for nonlinear elastodynamics. *International Journal for Numerical Methods in Engineering*, 50:1931–1955, 2001.

[Gonz00] O. Gonzalez. Exact energy and momentum conserving algorithms for general models in nonlinear elasticity. *Computer Methods in Applied Mechanics and Engineering*, 190:1763–1783, 2000.

[GBS05] M. Gross, P. Betsch, and P. Steinmann. Conservation properties of a time FE method. Part IV: Higher order energy and momentum conserving schemes. *International Journal for Numerical Methods in Engineering*, 63:1849–1897, 2005.

[GGH*97] H. Guo, H. Gao, R. W. Hendricks, R. K. Batra, Y. Lavin, Y. Levy, and L. Hesselink. Visualization of Triaxial Residual Stress Tensors Near Welds in HSLA-100 Steel. *International Conference on Residual Stresses-5*, Linkoping, Sweden, June 1997.

[Jac99] R. Jacobson. *Information Design*. The MIT Press, 1999.

[ML02a] X. Meng, T. Laursen. Energy consistent algorithms for dynamic finite deformation plasticity. *Computer Methods in Applied Mechanics and Engineering*, 191:1639–1675, 2002.

[Kin04] G. Kindlmann. Visualization and Analysis of Diffusion Tensor Fields. *PhD thesis*, University of Utah, 2004.

[MMS06] R. Mohr, A. Menzel, and P. Steinmann. Galerkin-based time integrators for geometrically nonlinear elasto-plastodynamics – Challenges in modeling and visualization. In *Visualization of Large and Unstructured Data Sets, GI-Edition Lecture Notes in Informatics (LNI)*, S-4:185–194, 2006.

[MMS07a] R. Mohr, A. Menzel, and P. Steinmann. Conservation properties of Galerkin-based time integrators for geometrically nonlinear elasto-plasto-dynamics. In Proceedings of *IMPLAST – Symposium on Plasticity and Impact Mechanics*, Bochum, 2007.

[MMS07b] R. Mohr, A. Menzel, and P. Steinmann. Conservation properties of Galerkin-based time-stepping schemes for finite elasto-plasto-dynamics. In Proceedings of *COMPLAS IX – International Conference on Computational Plasticity*, Barcelona, 2007.

[MMS07c] R. Mohr, A. Menzel, and P. Steinmann. A consistent time-FE method for large strain elasto-plasto-dynamics. Submitted for publication, 2007.

[New59] N. Newmark. A method of computation for structural dynamics. *ASCE Journal of the Engineering Mechanics Division*, 85:67–94, 1959.

[NJP05] A. Neeman, B. Jeremic, and A. Pang. Visualizing Tensor Fields in Geomechanics. *IEEE Visualization*, 2005.

[NSP06] L. Noels, L. Stainier, and J. Ponthot. An energy momentum conserving algorithm using the variational formulation of visco-plastic updates. *International Journal for Numerical Methods in Engineering*, 65:904–942, 2006.

[ST92] J. Simo, N. Tarnow. The discrete energy-momentum method. Conserving algorithms for nonlinear elastodynamics. *Zeitschrift fuer Angewandte Mathematik und Physik (ZAMP)*, 43:757–792, 1992.

[WMN*02] C.-F. Westin, S. E. Maier, H. Mamata, A. Nabavi, F. A. Jolesz, and R. Kikinis. Processing and visualization of diffusion tensor MRI. *Medical Image Analysis*, 6(2):93–108, 2002.

A Framework for the Visualization of Brain Structures

Sebastian Thelen
s_thelen@informatik.uni-kl.de
University of Kaiserslautern, Germany

Torsten Bierz
bierz@informatik.uni-kl.de
International Research Training Group
University of Kaiserslautern, Germany

Britta Müller
bmuelle@rhrk.uni-kl.de
University of Kaiserslautern, Germany

Hans Hagen
hagen@informatik.uni-kl.de
University of Kaiserslautern, Germany

Achim Ebert
ebert@informatik.uni-kl.de
University of Kaiserslautern, Germany

Eckhard Friauf
eckhard.friauf@biologie.uni-kl.de
University of Kaiserslautern, Germany

Jörg Meyer
jmeyer@uci.edu
University of California, Irvine, USA

Abstract: Nowadays biologists investigate different causes for deafness. One reason is a damage in a particular region of the auditory brain stem. Anatomical differences were discovered when investigating brain slices of different laboratory mice. However, these slices are only a two dimensional representation of a part of the brain. The arising question was how these differences of structure affect the three dimensional representation of this region. Therefore, an interdisciplinary framework was developed, which allows even unexperienced users to investigate and compare these regions.

1 Introduction

The *Superior Olivary Complex* (SOC) is a part of the auditory brain stem playing an important role in hearing. The SOC consists of several *cores*, each responsible for a particular aspect of sound processing. It was discovered that genetic suppression of a certain calcium channel in mouse brains results, among other things, in deafness by birth. Cross-sections of the SOC revealed anatomic differences in a core called *Lateral Superior Olivary* (LSO). LSOs of mice with genetic suppression (knock-out mice) differ in form and size, respectively number and density of cells from the LSOs of healthy mice. Further studies of these differences should be supported by a framework, that can visualize the cores of the SOC three dimensionally (especially the LSO) and determine characteristic parameters, such as volumes and surface areas of the cores. Of course professional applications used in *Magnetic Resonance Imaging* (MRI) or *Computed Tomography* (CT) can accomplish this task. Unfortunately their acquisition goes far beyond the financial scope of most research groups. Therefore this paper focuses on a low cost alternative, that does not require expen-

sive hardware. Instead the framework processes grayscale images of cross-sections, that can easily be obtained by using a camera and a microscope.
Section 2 provides an overview of related work in the field of brain visualization. In Section 3 the reconstruction of cores, which is based on contour information taken from digital grayscale images of the brain slices is presented. Section 4 describes methods for volume- and surface estimation used in the analysis of the cores. Finally Section 5 presents implementational aspects of the framework. The paper ends with a discussion and perspective on future work.

2 Related work

Nowadays numerous different applications deal with the visualization and analysis of cerebral data sets for various purposes. Some representative examples are given:
The *Allen Brain Explorer* [Al07], a product of the *Allen Institute for Brain Science*, provides a detailed cellular-resolution, genome-wide map of gene expression in the mouse brain. Because there is a high degree of similarity between the human genome and that of a mouse ($\sim 90\%$), the project offers the opportunity to further understanding of human disorders and diseases (e.g. Alzheimer's, Parkison's, epilepsy etc). In 2006, the explorer was able to visualize the expression of approximately 20.000 genes in a three dimensional model of the brain.
AnatQuest [An07] is based on the *Visual Human Project* [NLM07], a collection of digital images of complete human male and female cadavers in MRI, CT and anatomical modes. The images of the Visual Human data set are used by AnatQuest to create three dimensional models of anatomical objects within these slices. The *Insight Toolkit* (ITK) [ITK07] also supports the Visual Human Project and employs leading-edge segmentation and registration algorithms in two, three, and more dimensions. Developing these algorithms is of great importance in many branches of medicine, e.g. in the automatic identification of tumors.
Nearly all applications dealing with the visualization of biological data sets rely on a method to reconstruct the models to display. In the current approach this is done based on contour information. Besides the approach applied in this paper, which was first presented in [EPO91], various other authors dedicated their work to this topic. A detailed summary of previous work can be found in [MS92].
In order to achieve good results in the reconstruction process, it is advisable to align all slices in a preceding step. For cross-sectional data sets, pin holes in the preparations can simplify this task [SM01]. Unfortunately this method was not applicable in the current case, because the preparations were too fragile.

3 Core reconstruction

A typical data set to be processed consists of 6-10 grayscale images of anatomical cuts of the SOC, sliced at approximately 30 μm intervals. All images were segmented by manually marking the contours of the cores to be reconstructed (e.g. the LSO) with unique colors. Although it would be of great benefit to segment all pictures automatically, the manual marking is the only appropriate way at the moment. Most regions are so diffuse that it takes an experienced person with biological background to identify them correctly. All steps of the reconstruction process are described subsequently. After the extraction of contour information from the input files, an alignment procedure compensates transformations caused by the manual treatment of preparations. The preprocessed contours can then be used to identify correspondences between points of consecutive contours, which will provide the information needed to construct a geometric model of a core.

3.1 Contour extraction

First all core contours are extracted from the grayscale images. In a mathematical sense, a contour is an ordered sequence of two dimensional points. Obviously some caution is needed to identify this sequence correctly. Traversing the image pixelwise row by row will ignore the predecessor-successor order between contour pixels. *Contour Tracing Algorithms*, like the one of *Pavlidis* [Pa82], provide a solution to this problem. Pavlidis' algorithm extracts the point set of a contour in counterclockwise direction by always searching for the next rightmost contour pixel in moving direction. The idea is illustrated in Figure 1, where the contour pixels of a gray object are identified one after another in correct order. Solid arrows indicate directions in which a pixel is searched for. Directions with dashed arrows do not have to be checked anymore, because a suited pixel was already found before.

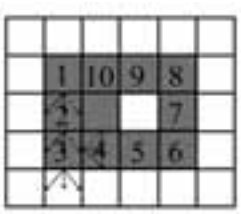
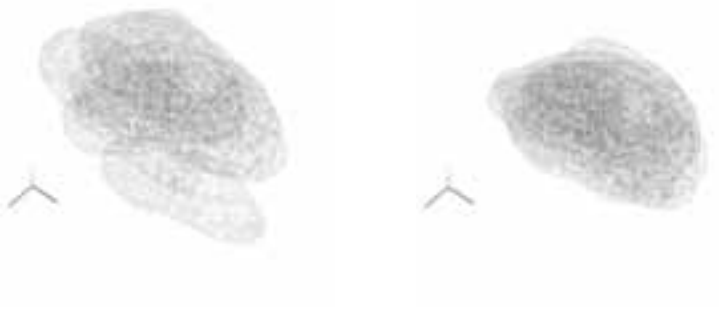


Figure 1: Contour extraction of a gray object by Pavlidis' algorithm. Arrows indicate directions where to search for the next pixel. Directions with dashed arrows do not have to be checked anymore. All pixels belonging to the contour are identified in correct order.



(a) (b)

Figure 2: (a) Situation before the alignment with barycenters mutually displaced an eigenvector crosses twisted against each other. (b) Situation after the alignment. Transformations due to manual placement are compensated.

3.2 Alignment

The next steps in the reconstruction process are based on the extracted contours. However, a misalignment of the slices due to manual placement onto glass object carriers prevents a direct use. Slices can be displaced, twisted and even distorted. The former two result from an imprecise placement, whereas the latter one is due to the varying pressure of a brush, used to place the preparations. Therefore, contours must be aligned in order to compensate these transformations. By performing a *Principal Component Analysis* (PCA) it is possible to treat translations and rotations between successive contours [La06]. Identifying the principal components of a two dimensional contour is done by calculating the eigenvectors and eigenvalues of the points' covariance matrix. This yields to two perpendicular vectors in the barycenter of each contour, i.e. the principal components. Contours are aligned by making the crosses formed by the eigenvectors coincident through mutual shifting and twisting, as shown in Figure 2. Provided that contours are quite similar and not too much transformed, this technique produces useful results in most situations.
However, results should be interpreted with some care, for it is possible that the structure of a core gets lost during the alignment. One can imagine a core where the centers of its contours are mutually displaced. By making these centers coincident, the natural structure of the core will be destroyed. The reason is the impossible determination whether the displacement of barycenters is due to the core's natural structure or to the manual positioning onto object carriers. Finding an appropriate alternative will be subject of future work.

3.3 Linking

The aligned contours are used to identify corresponding points in successive contours. Information gained in this step will be used to form the triangle meshes of a core's wire

frame model. The applied method was first presented by Ekoule et al. [EPO91]. A *correspondence*, or *linkage*, can be expressed by a function $Z : \mathcal{P} \rightarrow \mathcal{Q}$ from the set of points of the first contour into the set of points of the following one. The mapping of Z should be "natural", as illustrated in Figure 3. Here the point p is mapped to $Z(p)$ which seems

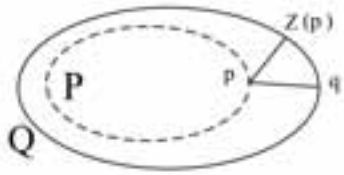


Figure 3: Defining a linkage between points of consecutive contours. Due to the geometry of both contours, a link between p and q would seem most "natural".

quite unnatural. Regarding the geometry, q would be a more suitable candidate. Ekoule et al. determine corresponding points by the use of a *local distance criterion* for *convex* contours. If a pair of points $(p_i, Z(p_i))$ is selected, $Z(p_{i+1})$ is searched for among the successors of $Z(p_i)$ within a local neighborhood, so that the distance between p_{i+1} and $Z(p_{i+1})$ becomes minimal.
Nonconvex contours are treated by projecting them onto their convex hulls and determining a linkage between the projected contours. The projection procedure must take into account the local deformation of a contour. For this purpose, contours are subdivided recursively into concavities which are projected onto the convex hull of the superior concavity.
At the moment the framework is able to identify *single branching* correspondences. For each core there is exactly one contour in every slice of the data set. *Multiple branching* situations arise when a core splits up. Implementing a correct multiple branching detection and treatment will enhance the frameworks functionality in future versions. Approaches how to do this can be found in [EPO91].

3.4 Triangulation

After linking points of consecutive contours, the triangle meshes of a core's model are constructed. The order of pixels within a contour and the mapping of $Z : \mathcal{P} \rightarrow \mathcal{Q}$ already define some triangle edges. However, assuming without loss of generality $|\mathcal{P}| \leq |\mathcal{Q}|$ holds (otherwise they are switched), there can be points in $\mathcal{Q}$ that are not linked to any point in $\mathcal{P}$. These must be taken into account when constructing the meshes. The method used is described by Linsen [Li97]. Unlinked points are assigned to points in $\mathcal{P}$ and possible gaps are closed. It is verified that the orientation of points is consistent for all triangles of a model. This is important for a correct visualization with OpenGL. As an example of the whole reconstruction process, the wire frame model of a Lateral Superior Olivary is shown in Figure 4.

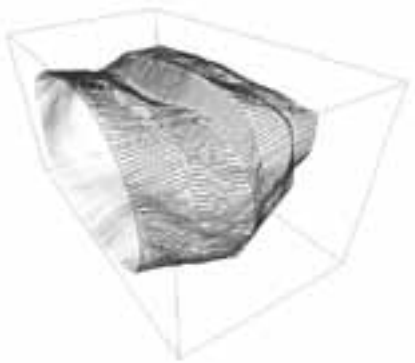


Figure 4: Wire frame model of a LSO. The model was reconstructed from 7 slices of a healthy mouse brain.

4 Analysis

As stated in the introduction, the LSOs of knock-out mice and wild-type mice differ from each other. Several parameters characterize these differences. The framework supports the comparison of the two mouse groups by providing an automated estimation of the volume- and surface values for the reconstructed cores.

4.1 Volume estimation

The volume of a core can be calculated by figuring out the integral $\int_Z A(z)dz$, where $A(z) =$ *contour area at depth* z [RH90]. This integral is approximated by

$$\sum_{i=0}^{n-2}(z_{i+1} - z_i)\left[\frac{A(z_i) + A(z_{i+1})}{2}\right],$$

a discrete sum of trapezoidal areas beneath the graph of A. The parameter n describes the number of contours, z_i their depths in z-direction and $A(z_i)$ the contour area at position z_i. A comparison of methodologies shows a high degree of similarity between the results of automatic estimations and the ones of conventional manual calculations. The resemblance underlines the correctness of this approach.
An interesting fact is that the volumes are independent from the method of alignment (as far as only translations and rotations are treated). This is guaranteed by *Cavalieri's Principle*, which claims that "*if, in two solids of equal altitude, the sections made by planes parallel to and at the same distance from their respective bases are always equal, then the volumes of the two solids are equal*" [We03]. Hence, possible variations in the method of alignment will not affect volume estimation.

4.2 Surface estimation

Surface estimation of a core is straight forward compared to its volume estimation. It is done by summing up all triangle areas of a core's model. Obviously this estimation is strongly affected by the method of alignment, so alterations in future versions will lead to different results.

5 Implementation

Implementing a prototypic software application requires deliberate decisions regarding the tools to be used. The framework is designed for WindowsXP and was implemented choosing C++ as a programming language. C++ is widely spread and comes along with robust and efficient compilers for nearly every platform. Microsoft's Windows is the most convenient operating system and is best known by most users. It provides various tools for handling a development environment (e.g. Microsoft Visual Studio 2005). In order to create an appealing visualization and comfortable Graphical User Interface (GUI), OpenGL and Trolltech's Qt4 were used, which are freely available for a multitude of systems. The framework was tested on an Intel Centrino 1.7 GHz system with 1024 MB RAM and a 128 MB ATI 9700 graphic card. At least 50 MB of free disk space are required.

5.1 Basic architecture

The framework is built of four modules which implement the methods described in the previous sections. The modules depend on each other an are organized in a pipeline as follows:

ContourMaker This component extracts contours from image files in the *Portable Network Graphics* format (PNG) by executing an implementation of Pavlidis' algorithm. All contour pixels within a slice are identified and saved in *xls*-files.

SliceConstructor The *xls*-files produced by the ContourMaker are processed by the SliceConstructor, which performs a principal component analysis on the point sets and identifies translation vectors and rotation angles for the following alignment. All alignment information is stored in *xls*-files for further use.

VolumeConstructor This component is responsible for the reconstruction of core models. Points in consecutive contours are linked as described as described above triangle meshes are created. Results are saved as *Obj/Wavefront*-files, which can be displayed by various viewers.

Viewer3D Finally a three dimensional representation of the SOC and its cores is created and displayed to the user. This module offers the possibility to examine data sets

in a virtual environment and to interact with them by adjusting the perspective as required (through translation, rotation and zooming).

All components are controlled by *configuration files* which mainly contain information about the input files to process and the output files to create.

5.2 Graphical User Interface

When starting the framework, the application's main window, shown in Figure 5, is presented to the user. From here it is possible to perform all provided operations. It mainly consists of two graphic windows (instances of Viewer3D) which can display different data sets simultaneously. Thus it is possible to directly compare knock-out and wild-type mice. Beneath each window several control elements allow the user to take influence on the pre-

Figure 5: The framework's main window consists of two graphic windows. The user intuitively interacts with data sets by using the mouse to adjust his view. The design facilitates the direct comparison of wild-type and knock-out mice.

sentation of data sets (distance between slices, current time step in a series of data sets, parameters for artificial end pieces). Further operations are supported by elements of the main menu. Results of the automated analysis are presented in form of histograms. A histogram always displays parameter estimations for the cores in the left and right graphic window, which simplifies the task of recognizing regularities and differences in data sets. Samples of scenes showing a reconstructed Lateral Superior Olivary are presented in Figure 6. Figure 6(a) illustrates the model of a LSO, reconstructed from seven cross-sections which were sliced at intervals of 30 μm. In order to investigate the core together with the embedding tissue, the original data set (grayscale images of the preparations) has been added to the scene in Figure 6(b). By rotating, translating and zooming the user can adjust

the point of view as required. A color-mapped version of Figure 6(b) can be found in the colorplate on p. 191. Color-mapping is achieved by applying a transfer function to the

(a) (b)

Figure 6: A sequence of scenes generated by the framework. (a) Surface model of a LSO. (b) Surface model embedded into the surrounding tissue.

original data set. They are a common technique in scientific visualization and are used to map scalar values to optical features. The framework uses transfer functions to map grayscale values to RGB-colors. Thus this it gets possible to emphasize certain features of a data set, like darker regions with lots of cells, by highlighting them with different colors. Transfer functions are defined in a separate dialog (see colorplate) that allows the declaration of up to three disjoint intervals in the set of grayscale values over which RGB-colors are interpolated linearly.

6 Conclusion and Future Work

In this paper the development of an interdisciplinary framework for the visualization of brain structures was described. The framework can visualize a particular region of the auditory brain stem of a mouse. Data sets consist of cross-sections containing contour information about the cores to be reconstructed. The reconstruction process was discussed in detail and difficulties with the alignment were pointed out. Different opportunities to analyze the structures were described.
Considering the specific needs of the involved biologists, the framework is used in the validation of a deafness model by providing ways to visualize and quantify differences between two mouse groups.
Future work will focus on enhancing the alignment of contours and increasing the framework's flexibility by providing ways to handle branching structures. Another extension is the automatic estimation and calculation of the amount of cells and cell cores, because biologists still try to count their occurrences by hand.

References

[Al07] Allen Brain Atlas - Neuroscience Gateway. http://www.brainatlas.org/aba/. 2007.

[An07] Anatquest - Anatomic Images Online. http://anatquest.nlm.nih.gov/. 2007.

[CS78] Christiansen, H. und Sederberg, T.: Conversion of complex contour line definitions into polygonal element mosaics. In: *Computer Graphics*. volume 12. pp. 187–192. ACM. 1978.

[EPO91] Ekoule, A., Peyrin, F., und Odet, C.: A triangulation algorithm from arbitrary shaped multiple planar contours. In: *Transactions on Graphics*. volume 10. pp. 182–199. ACM. Juli 1991.

[Fr07] Friauf, E. Vom GehËr zum Gehirn: die Neurobiologie des HËrens. http://www.uni-kl.de/wcms/agf_hoeren.html. 2007.

[ITK07] Insight Toolkit (itk). http://www.itk.org/index.htm. 2007.

[Ka99] Kaballo, W.: *EinfÂhrung in die Analysis III*. Spektrum. 1999.

[La06] Lacour, F.: Large-scale biomedical image registration using the principal component analysis. Project thesis. University of Kaiserslautern. 2006.

[Li97] Linsen, L.: Schnitt und Vereinigung von Kontrollnetzen. Diploma thesis. University of Karlsruhe. 1997.

[MS92] Meyers, D. und Skinner, S.: Surfaces from contours. In: *Transactions on Graphics*. volume 11. pp. 228–258. ACM. Juli 1992.

[NLM07] The Visible Human Project. http://www.nlm.nih.gov/research/visible/visible_human.html. 2007.

[Pa82] Pavlidis, T.: *Algorithms for Graphics and Image Processing*. Computer Science Press. Maryland. 1982.

[RH90] Rosen, G. und Harry, J.: Brain volume estimation from serial section measurements: A comparison of methodologies. In: *Journal of Neuroscience Methods*. volume 35. pp. 115–124. 1990.

[SM01] Shulga, D. und Meyer, J.: Aligning large-scale medical and biological data sets: Exploring a monkey brain. In: *Visualization, Imaging and Image Processing (VIIP 2001)*. volume 3. pp. 434–439. The International Association of Science and Technology for Development (IASTED). ACTA Press. September 2001.

[Th07] Thelen, S.: Entwicklung eines Frameworks zur Visualisierung, Exploration und Komparation biologischer Datensâtze. Diploma thesis. University of Kaiserslautern. 2007.

[We03] Weisstein, E. Cavalieri's Principle. From MathWorld - A Wolfram Web Resource. http://mathworld.wolfram.com/CavalierisPrinciple.html. 2003.

GPU Accelerated Gesture Detection for Real Time Interaction

Torsten Bierz
University of Kaiserslautern
International Research Training Group
P.O. Box 3049
67653 Kaiserslautern, Germany
Email: bierz@informatik.uni-kl.de

Achim Ebert
University of Kaiserslautern
P.O. Box 3049
67653 Kaiserslautern, Germany
Email: ebert@informatik.uni-kl.de

Jörg Meyer
University of California Irvine
644E Engineering Tower
Irvine, CA 92697-2625, USA
Email: jmeyer@uci.edu

Abstract: Over the past years, the interaction between humans and computers (HCI) evolved to one of the most important research topics in computer science. Therefore, finding a way for an intuitive, easy and affordable interaction is the main challenge. Optical markerless tracking using consumer hardware can satisfy these problems. However, in order to be able to interact in an efficient way, the tracking and furthermore the interaction must be handled in real time. This leads to efficient use of current GPUs in order to speed up the tracking and furthermore the gesture recognition. Involving these issues, an approach for an implementation and system will be presented in the following paper.

1 Introduction

Nowadays, humans are surrounded by a highly sophisticated environment and technology. From these circumstances, the interaction and in addition the human-computer interaction (HCI) is arising to a very important research topic among scientists. Interacting with computers can be handled in many different fashions and furthermore with a huge amount of different interaction devices. Consequently, the interaction should be as intuitive and simple as possible in order to provide an efficient and satisfying way for interaction even for untrained persons .

In recent years, many different interaction types and sophisticated hardware input devices have been developed. In this regard, one of the intuitive options for interaction commands are poses and gestures, which can be detected in a satisfying manner by additional hardware, e.g. data gloves or markers. However, most untrained or novel users are not familar with wearing additional hardware devices and behave unnatural. Although, the devices

have become smaller and more lightweight, they might still be uncomfortable to wear. Therefore, the question arises whether it is possible and sufficient to track only the poses and gestures of the users and therefore their hands by means of skin detection in order to obtain information for interaction purposes.

Commercially available optical tracking systems are often quite expensive and usually require markers or other visual means for optical reference. Therefore, the question arises whether we really need those tracking devices or whether off-the-shelf components like webcams or digital cameras can also be sufficient for this type of interaction.

Based on these ideas, a system has been developed which facilitates optical tracking without optical markers for human gesture-controlled real-time interaction with a software system. The real-time capability is accomplished by using a GPU-accelerated computation method.

2 State of the Art

Gaining insight into current state of the art research in human-computer interaction (HCI), many different interaction methods and approaches as well as hardware devices exist. These methods can be categorized in different tracking methods:

- Pointing devices, which are tracked in context of the users, and allow a three dimensional interaction, e.g., wands [CB03], or the dragonfly device [SR03],
- Spot pointing devices, e.g. laser pointers [ATK+05, ON01], where the resulting spot is used for selecting or moving of objects or writing on the screen,
- Or trackable interaction devices, e.g. the Cubic Mouse [FPW+00], or data gloves [Imm07], where the position needs to be tracked in order to obtain interaction information for the applications.

However, the user is always dependent on the use of a specific device for the interaction, keeping his or her hands occupied, which may not always be feasible, for instance, in a sterile medical environment.

In order to be independent of additional devices, interaction should be kept as intuitive and natural as possible. One good solution is the tracking of the user and the use of eye movements to control navigation or detect the user's focus [BDDG03] (gaze tracking). However, a serious drawback to this method is the anthropogenic fact that humans tend to focus on different targets in very short time intervals, which results in difficulties for deciding whether a change of the eye position should result in a scene change or not [Zha03]. When considering which human bodily expressions can or should be tracked, usually the hands come to mind first, as they are certainly the most natural and intuitive interaction devices.

In recent years, different approaches for using the hands for human-computer interaction have been studied and developed. Strickon and Paradiso [SP98] tried to track hands mov-

ing over a low-cost laser-scanning rangefinder. With this device, the user interacts with a laser in front of a screen. Varona et al. [VRL05] and Thayananthan et al. [TSTC06] presented other approaches for tracking of hands and faces trying to classify gestures by a hierarchical Bayesian filter, which consisted of different rotations of the gesture. They demonstrated that building up different rotated gestures in preprocessing can be avoided by using rotation invariant gestures.

Nowadays, quite complex and furthermore expensive camera solutions and systems exist, which are able to track a person or interaction devices. However, these devices often need a specific setup and may require additional hardware. Furthermore, they often require reflectors or other optical indicators, so called markers or targets, which must be attached to the desired tracking object [Gmb07]. In this article, we develop other tracking options, employing consumer hardware such as webcams or digital cameras and develop a markerless gesture-based input method. In addition, the computational power of the GPU is used to accelerate computations and to obtain real-time capability of the detection and tracking system. Modern available webcams or digital cameras have a sufficiently high resolution of at least 640x480 pixels and a frame rate of 25 to 30 frames per second. Furthermore, current computers are mostly equipped with satisfying graphic cards, that can take care of the necessary computations in order to match the real-time constraints, which comes along with the requirements for smooth interaction.

This article presents an efficient way of interacting with virtual environments by using consumer hardware. In Section 3, the basic concepts and ideas of the system will be presented. Within this Section, a short system overview will be given, followed by a description of the image capturing and the necessary image processing steps ofskin detection, noise filtering, and outline extraction. For acceleration purposes, these steps are computed on the graphics processing unit (GPU). In section 4, the pose and gesture detection is explained. Therefore, some basic operations on images must be explained first. These are the so-called image moments and their computation, which will be used for the comparison of the captured regions to poses, which are stored in a database. The paper concludes with results from the skin detection and gesture recognition algorithms and with an outlook on future work.

3 Markerless Interaction

In the following section first a short overview of the system architecture and the usage of the algorithms on the GPU is presented. Then, the different steps of capturing an image, applying skin detection, noise removal and silhouette reduction are described. The results are then used for pose and gesture detection, as described in Section 4.

3.1 System Architecture

In many applications, the real-time responce of the system for the interaction task is most important providing an immediate feedback to the user. Consequently, tracking and recognition must be handled as fast as possible, preferably at the same rate as images are displayed or obtained. Otherwise, the user could be easily confused or irritated and would not know whether he or she has already initiated an action. As a result, it is important to handle all necessary calculations within a given time frame. Although modern CPUs are becoming increasingly faster, it is sometimes difficult to process all incoming and outgoing data in real-time. Many researches now explore the potentials of the current graphic processing units (GPUs). Furthermore, by now the GPU is not only utilized for graphical purposes, but also for performing highly parallel tasks efficiently. The usage of graphic cards for such a purpose is defined as general purpose computation on the GPU (GPGPU for short) [Fun05, FM04]. Textures or in this case image data, as obtained from digital cameras or webcams, are well-suited for the computation and execution of multiple operations in parallel on the GPU.

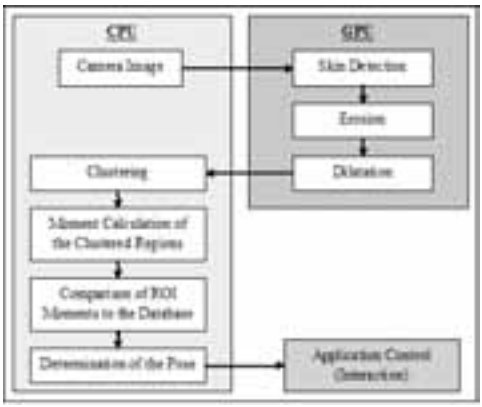


Figure 1: Scheme of the algorithm

The current implementation of the system uses advanced features and acceleration capabilities of the GPU. An outline of the system architecture is given in Figure 1. First, the camera image is captured and transmitted to the GPU as a texture. Next, the skin detection, which is presented in 3.2, is performed on the GPU implemented as a fragment shader. The opening of the image, which is separated in an erosion and a dilation part, reduces noise in the binary image. This algorithm can also be efficiently computed on the GPU. The results from the filter kernels are stored as alpha values in the camera image. All of these algorithms are implemented as fragment shaders. After their computation, the image is transmitted to the CPU where the remaining tasks are performed. The clustering as well as pose and gesture detection based on image moments, which will be handled in Section 4, cannot be implemented as a highly parallel GPU algorithm and therefore are carried out on the CPU.

3.2 Skin detection

Common skin detection methods classify pixels according to a specific color space, e.g., RGB, normalized RGB, HSV, or YCrCb. Other methods try to classify skin regions based on probability maps, which are generated using training sets. This process needs the manual selection of the regions containing skin, and is therefore expensive. Neural networks can be used to simplify it. A more detailed overview of current state-of-the-art approaches are given by Martinkauppi et al. [MSP03] and by Kakumanu et al. [KMB07].

For the detection of skin in form of a binary classification, Gomez and Moralez proposed an approach based on normalized RGB values [GM02]. The normalization of the RGB values reduces the ambient illuminance and offers the advantage of being more robust against varying light conditions. The decision whether a pixel is classified as "skin" or "not skin" is made by logically combining the following three thresholding equations:

$$\frac{r}{g} > 1.185 \quad and \quad \frac{rb}{(r+g+b)^2} > 0.107 \quad and \quad \frac{rg}{(r+g+b)^2} > 0.112 \qquad (1)$$

where $r, g, b \in [0, 1]$ are the normalized RGB values.

If the query succeeds in all three cases, the pixel is classified as "skin", otherwise as "not skin". Consequently, when applying this skin detection method to a captured image, the result will be a binary coded image. This separates the image into regions containing either "skin" or "no skin" (see Figure 2).

(a) Captured Image (reproduced in color on p. 192) (b) Image after skin detection

Figure 2: Applying the skin detection to a captured image

3.3 Noise reduction

When using consumer off-the-shelf cameras, a well known problem is their susceptibility to noise. In order to improve the captured image, one can smooth it using a discrete

Gaussian filter. For this, the image is convoluted with a Gaussian filter mask. However, when the noise reduction is applied to the captured image, it is already in a binary format (see Figure 2). Applying a Gaussian filter to a binary image would result in blurred edges and increased pixel errors. To avoid this, morphological operators are employed as they are more efficient on binary images. We use an opening operation, which is a combination of an erosion and a dilation filter.

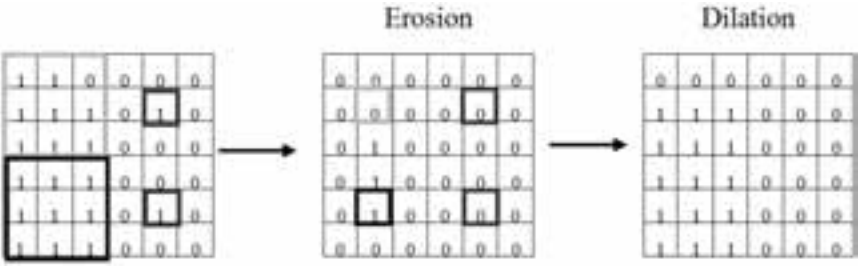


Figure 3: Opening of a binary image (reproduced in color on p. 192)

The erosion as well as the dilation steps are applied as filter kernels of a 3x3 matrix. An example is shown in Figure 3. Additional information on image processing techniques and morphological operations can be found in [Jäh97].

In order to receive the contour or silhouettes of the segmented image, algorithms and methods for edge detection must be applied. Multiple approaches have been described in the literature, e.g., gradient-based methods like Sobel or Canny operators [Jäh97]. However, for binary images, the computation can be done in a more efficient way by applying a modified hit-miss operator. The defined hit-miss operator operates as a search operator for a 4-connected neighborhood of the current pixel with the value ”1”. If no zero is found in the neighborhood of the pixel, the value of the pixel is set to zero. As a result, only the outline of an object remains visible (represented by ”1” pixels).

(a) Original image and detected contours (reprduced in color on p. 192)

(b) Result from sample image

Figure 4: The modified hit-miss operator

4 Gesture Detection

In the following section the clustering of the resulting image is described. After that, the concept of image moments is explained. They are calculated for the regions previously found by the clustering and are compared to the moments in the stored database. Finally, the action assigned to the detected pose is performed.

4.1 Clustering

After the computation of the outlines of the skin region done on the GPU, these regions of interest are separated into clusters. Without region clustering a recognition of poses is not feasible. This can be seen in Figure 4. This image consists of several regions, e.g. the eyes, the head, and the hand. It should be noted that there is some noise present in the image. If the whole image was compared to a pose (e.g. as seen in Figure 5), there would be no success, because the image does not exclusively contain the pose. There are still some other objects present in the image, which lead to incorrect results. Therefore, the respective discrete regions must be found and clustered. In order to gather the regions, region growing [Ver91] is used. An 8-connected neighborhood is considered for determining connected regions. The boundary of each region is returned as a result.

Comparing gestures or poses is challenging. Several solutions exist, e.g. performing a pixel-wise comparison of two images or image regions. However, this method can only be used if the reference image and the searched region have the same scale, the same rotation, and the same position. Therefore, when dealing with poses or gestures, a pixel-based comparison approach is inefficient and will not provide sufficient results. The next section describes a more appropriate solution to this problem. The method is called "image moments".

4.2 Image Moments

The concept of moments is well known in physics and mathematics. Image moments are specific weighted averages of image pixels. These moments, or in this context the values of the moments, represent the feature which is supposed to be recognized. In our case, the feature is a gesture or a pose. Consequently, a definition of moments is required which is invariant to translation, rotation or scale. In order to be able to define and compute such moments, we define raw moments:

$$M_{ij} = \sum_{x=0}^{w-1} \sum_{y=0}^{h} x^i y^j I(x, y), \tag{2}$$

where $I(x, y)$ is the pixel intensity at location (x,y), w and h the width and the height of the image, and $i, j \in \mathbb{N}$. The raw moments are neither invariant to translation nor to

scale or rotation. In order to devise a translation invariant representation, the region is moved to the origin. These moments are called central moments. For this, the centroid $(x_c, y_c) \in \mathbb{R}^2$ can easily be calculated $(x_c, y_c) = \left(\frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}}\right)$. The central moments are calculated by:

$$\mu_{ij} = \sum_x \sum_y (x - x_c)^i (y - y_c)^j I(x, y). \tag{3}$$

Central moments are still not scale and rotation variant. For scale invariance, the central moments are dividedby the moment μ_{00}, which is the same as M_{00}. These moments are

$$\eta_{ij} = \frac{\mu_{ij}}{\mu_{00}^{\left(1+\frac{i+j}{2}\right)}} \quad for \quad i + j \geq 2. \tag{4}$$

In order to have a rotation invariant representation of these moments, Hu proposed a set of seven rotation invariant moments [Hu62]. Flusser proved in [Flu00] that a set of six moments is sufficient because of dependencies of two rotation invariant moments. These moments are

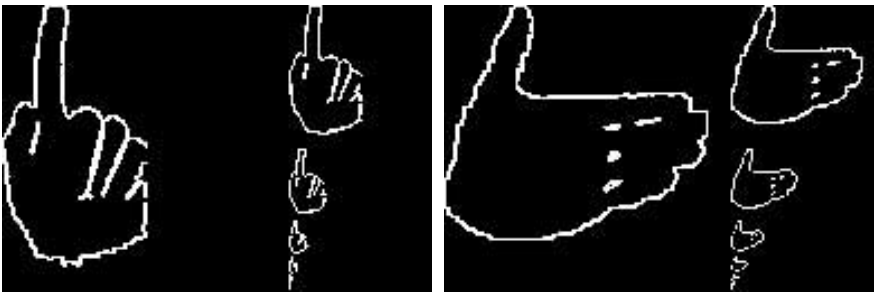
$$\begin{aligned}
\psi_1 &= \eta_{20} + \eta_{02}, \\
\psi_2 &= (\eta_{30} + \eta_{12})^2 + (\eta_{03} + \eta_{21})^2, \\
\psi_3 &= (\eta_{20} - \eta_{02})((\eta_{30} + \eta_{12})^2 - (\eta_{03} + \eta_{21})^2) + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{03} + \eta_{21}), \\
\psi_4 &= \eta_{11}((\eta_{30} + \eta_{12})^2 - (\eta_{03} + \eta_{21})^2) - (\eta_{20} - \eta_{02})(\eta_{30} + \eta_{12})(\eta_{03} + \eta_{21}), \\
\psi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})((\eta_{30} + \eta_{12})^2 - 3(\eta_{03} + \eta_{21})^2), \\
&\quad + (3\eta_{21} - \eta_{03})(\eta_{03} + \eta_{21})(3(\eta_{30} + \eta_{12})^2 - (\eta_{03} + \eta_{21})^2), \\
\psi_6 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})((\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2), \\
&\quad - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})(3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2).
\end{aligned}$$

The moments of the poses are represented by this set of six values and are invariant to rotation, translation and scale. They can be computed as a prepossessing step for the stored poses in the gesture database.

4.3 Hierarchical Moments

In order to achieve higher recognition rates, the image moments are hierarchically build. The hierarchical moments can be described as moments of the poses in different resolutions. An example for the poses is presented in Figure 5.

The poses and their corresponding values are stored in the database. This reduces the computation time during gesture recognition.



(a) Pointing Gesture (b) Rotation Gesture

Figure 5: Two Sample Poses

4.4 Pose Detection

The resulting regions of the clustering, given by their bounding boxes, are used for the pose detection. For this purpose, hierarchical moments for the regions as previously defined are calculated and compared to the stored moments in the database. If the results are within a certain threshold range, the region is recognized as containing a pose. Two examples of the original image and the resulting poses are shown in Figures 6.

4.5 Following of Hands

The regions given by the clustering might contain poses. The detection based on the image moments provides the information, whether the region contains a pose. These regions usually move slowly. When comparing the resulting regions from one frame to another, the corresponding regions of the two frames usually overlap. Therefore, a simple overlapping test for axes-aligned bounding boxes can prove this issue. Having found two overlapping regions, the changing of their area usually remains in a certain threshold. In the current implementation, the threshold is set to 90%, which lead to sufficient results. So, if a pose was detected in a frame, and is not detected in the following frame, it is possible to keep track of the position of the hands.

5 Results and Future Work

Most cameras have a frame rate of either 25 or 30 frames per second, and the computation has to be done within a time frame of about 0.03 seconds to achieve interactive response of the visualization system. With the presented GPU accelerated approach the computation can be handled within this time interval, which is important for realtime interaction.

The recognition of poses and gestures has been tested under different lighting conditions.

The overall recognition rates depending on the normalized RGB values were sufficient. However, in dark rooms as well as under direct sunlight exposure, the pose and gesture recognition was insufficient and further research has to be done to calibrate the camera. In dark rooms it is quite difficult to detect skin. This may become an issue if the gestures are supposed to control an application projected onto a screen in a dark room. In this context simple fluorescent gloves in combination with an ultraviolet lamp or an additional infrared light source might lead to improved recognition results. For direct sunshine, it might be possible to compute the average of the region and an offset, which can be substracted in order to obtain higher recognition rates. Another option is an automated recalibration of the camera to compensate for different exposure situations.

In order to accelerate the process of detection, following of the hands can be used as a prediction model, where only a smaller or specific region of the image needs to be examined. In this case, the moments, which are currently computed for all occurring regions, can be reduced to the regions, where a gesture or pose was detected first.

Currently, interaction with the system is based on the first recognized pose or gesture. Two-handed interaction is currently not supported. One of the next steps will be the implementation of the two- instead of single-handed interaction, which will result in advanced possibilities for interaction. Furthermore, the current system is designed for single-user interaction. Another aspect is multiuser interaction. For this, suitable metaphors or mechanism must be developed to facilitate this type of interaction.

Acknowledgements The authors would like to thank the German Research Foundation (DFG) for supporting the International Research Training Group (IRTG 1131). In addition, the members of the Computer Graphics and Visualization Research Groups at the University of Kaiserslautern, Germany, and the University of California, Irvine, as well as the members of the IRTG 1131 for their cooperation and assistance.

References

[ATK+05] Benjamin A. Ahlborn, David Thompson, Oliver Kreylos, Bernd Hamann, and Oliver G. Staadt. A practical system for laser pointer interaction on large displays. In *VRST: Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pages 106–109, 2005.

[BDDG03] Patrick Baudisch, Doug DeCarlo, Andrew T. Duchowski, and Wilson S. Geisler. Focusing on the essential: considering attention in display design. *Communications of the ACM ACM*, 46(3):60–66, 2003.

[CB03] Xiang Cao and Ravin Balakrishnan. VisionWand: Interaction techniques for large displays using a passive wand tracked in 3D. In *Proceedings of the 16th annual ACM symposium on User interface software and technology*, pages 173–182. ACM Press, 2003.

[Flu00] Jan Flusser. On the independence of rotation moment invariants. *Pattern Recognition*, 33(9):1405–1410, 2000.

[FM04] James Fung and Steve Mann. Using Multiple Graphics Cards as a General Purpose Parallel Computer: Applications to Computer Vision. In *ICPR '04: Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 1*, pages 805–808, Washington, DC, USA, 2004. IEEE Computer Society.

[FPW+00] Bernd Fröhlich, John Plate, Jürgen Wind, Gerold Wesche, and Martin Göbel. Cubic-Mouse-Based Interaction in Virtual Environments. *IEEE Computer Graphics and Applications*, 20(4):12–15, 2000.

[Fun05] James Fung. *GPU Gems 2 : Programming Techniques for High-Performance Graphics and General-Purpose Computation (Gpu Gems) (Hardcover)*. Addison-Wesley Professional, March 2005.

[GM02] G. Gomez and E. Morales. Automatic feature construction and a simple rule induction algorithm for skin detection. In *Proceedings of Workshop on Machine Learning in Computer Vision*, pages 31–38, 2002.

[Gmb07] Advanced Realtime Tracking GmbH. Advanced Realtime Tracking GmbH: Finger Tracking and Markers, 2007. http://www.ar-tracking.de.

[Hu62] Ming-Kuei Hu. Visual pattern recognition by moment invariants. *Information Theory, IEEE Transactions on*, 8(2):179–187, Feb 1962.

[Imm07] Immersion. Wireless Data Glove: The CyberGlove®II System by Immersion, 2007.

[Jäh97] Bernd Jähne. *Digital Image Processing: Concepts, Algorithms, and Scientific Applications*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 6. edition edition, 1997.

[KMB07] P. Kakumanu, S. Makrogiannis, and N. Bourbakis. A survey of skin-color modeling and detection methods. *Pattern Recogn.*, 40(3):1106–1122, 2007.

[MSP03] B. Martinkauppi, M. Soriano, and M. Pietikainen. Detection of skin color under changing illumination: a comparative study. In *Image Analysis and Processing, 2003.Proceedings. 12th International Conference on*, pages 652–657, 17-19 Sept. 2003.

[ON01] Dan R. Olsen and Travis Nielsen. Laser pointer interaction. In *CHI '01: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 17–22. ACM Press, 2001.

[SP98] Joshua Strickon and Joseph Paradiso. Tracking hands above large interactive surfaces with a low-cost scanning laser rangefinder. In *CHI '98: CHI 98 conference summary on Human factors in computing systems*, pages 231–232, New York, NY, USA, 1998. ACM Press.

[SR03] Oliver Stefani and Jörg Rauschenbach. 3D input devices and interaction concepts for optical tracking in immersive environments. In *EGVE '03: Proceedings of the workshop on Virtual environments 2003*, pages 317–318, New York, NY, USA, 2003. ACM Press.

[TSTC06] Arasanathan Thayananthan, Bjorn Stenger, Philip H. S. Torr, and Roberto Cipolla. Model-Based Hand Tracking Using a Hierarchical Bayesian Filter. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(9):1372–1384, 2006.

[Ver91] David Vernon. *Machine vision: automated visual inspection and robot vision*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1991.

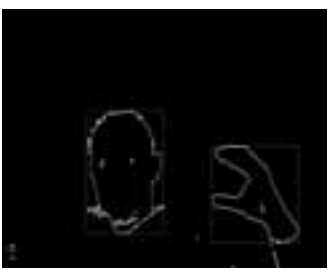
[VRL05] Xavier Varona, Jose Maria Buades Rubio, and Francisco J. Perales López. Hands and face tracking for VR applications. *Computers & Graphics*, 29(2):179–187, 2005.

[Zha03] Shumin Zhai. What's in the eyes for attentive input. *Communications of the ACM*, 46(3):34–39, 2003.

(a) Captured Image of a pointing gesture (reproduced in color on p. 192)

(b) Resulting image with recognized pointing gesture

(c) Captured Image of a grabbing gesture (reproduced in color on p. 192)



(d) Resulting image with recognized grabbing gesture

Figure 6: Examples of recognized poses

Deriving Global Material Properties of a Microscopically Heterogeneous Medium – Computational Homogenisation and Opportunities in Visualisation

C. B. Hirschberger[1], S. Ricker[1], P. Steinmann[2], N. Sukumar[3]

[1] University of Kaiserslautern
Chair of Applied Mechanics, Department of Mechanical and Process Engineering
P.O. Box 3049, 67653 Kaiserslautern, Germany
bhirsch@rhrk.uni-kl.de

[2] University of Erlangen-Nuremberg
Chair of Applied Mechanics, Department of Mechanical Engineering
Egerlandstraße 5, 91058 Erlangen, Germany
steinmann@ltm.uni-erlangen.de

[3] University of California, Davis
Department of Civil & Environmental Engineering
One Shields Avenue, Davis, CA 95616. U.S.A.
nsukumar@ucdavis.edu

Abstract: In order to derive the overall mechanical response of a microscopically material body, both the theoretical and the numerical framework of multi scale consideration coined as computational homogenisation is presented. Instead of resolving the actual heterogeneous microstructure in all detail for its simulation, representative micro elements are considered which provide the material properties for the coarse or rather scale. This procedure allows for a smaller and less inexpensive computation. However both the chance and challenge of visualising the decisive features arise on two scales.

1 Introduction

If a material—although it may appear macroscopically homogeneous—at a closer look possesses a heterogeneous microstructure, we would rather account for this microstructure than posing a inaccurate macroscopic constitutive assumption. Such microscopic inhomogeneities may for instance be voids, small inclusion, micro cracks [NNH99] as for instance those that occur in metal alloy systems, polymer blends, porous and cracked media, polycrystalline materials, or composites [KBB01].

Approaches to obtain the overall characteristics of such heterogeneous materials have been

developed since the 1960s by e. g. Hashin and Shtrikman, Hill, Willis and Walpole [HS62, Hil63, Hil72, Has64, Wal66, Wil77, Wil81, Has83] and are comprised in the recent monograph of Nemat-Nasser and Hori [NNH99]. They provide a framework to relate overall aggregate properties to micro-properties, thereby they involve the concept of a representative volume element (RVE) over which an averaging of key quantities is performed [NNH99]:

> An RVE for a material point of continuum mass is a material volume which is statistically representative of the infinitesimal material element.

Within the framework of computational mechanics involving multiple scales, these concepts have been applied to the numerical solution during the last decade by several researchers [TIK98, SBM98, MSS99, MMS99, THKK00, FC00, TK01, KBB01, MMS01, Kou02, KGB02, MK02, MD04]. Rather than resolving the single heterogeneities individually, they have made use of the theoretical foundations and developed algorithms which allow to obtain macroscopic material properties from underlying microstructures during the computation as these evolve under the deformation.

In this contribution, we adopt a multiscale framework within the context of large deformations. The notation is adopted along the lines of our earlier contributions [HKS06, HKS07a, HKS07b, HKS07c], whereby we now restrict ourselves to a classical continuum without additional degrees of freedom on both the macro and the micro level.

We first describe the theoretical framework (Section 2) which involves the continuum description for both the macro and the micro scale, the latter being occupied by the representative volume element, and the relations between the two scales. Then the approach to the computational homogenisation is explained (Section 3). The paper closes with numerical examples (Section 4) and a brief conclusion (Section 5).

2 Homogenisation

We review the theoretical framework which relates the continuum on the macro scale, for which no constitutive assumption is made, with the underlying microstructure advocated by the representative volume element. Thereby the continuum mechanical framework of finite deformation is pursued analogous to the contributions of [Hil72, KGB02, Kou02, KGB02, Mie03]. Note that throughout the paper quantities on the macro scale are denoted by $\bar{\bullet}$, while quantities within the RVE are denoted by plain letters. Body forces will be omitted on both scales.

2.1 Governing equations on the macro scale

As already indicated, the mechanical behaviour on the macro scale is described within a standard continuum mechanics framework, whereby no constitutive assumption is posed.

2.1.1 Continuum framework

On the macro level we consider the behaviour of a physical point $\bar{\mathcal{P}}$ within a body $\mathcal{B}$. It has the initial placement $\bar{\boldsymbol{X}}$ in the material configuration $\bar{\mathcal{B}}_0$ and the current placement $\bar{\boldsymbol{x}}$ in the spatial configuration $\bar{\mathcal{B}}_t$, respectively. The deformation map $\bar{\boldsymbol{\varphi}}$ and deformation gradient $\bar{\boldsymbol{F}}$ are defined as

$$\bar{\boldsymbol{x}} = \bar{\boldsymbol{\varphi}}(\bar{\boldsymbol{X}}, t) \qquad\qquad \bar{\boldsymbol{F}} = \nabla_{\bar{\boldsymbol{X}}} \bar{\boldsymbol{\varphi}} \tag{1}$$

The weak formulation of the balance of momentum, derived from the Dirichlet principle[1], reads

$$\int_{\bar{\mathcal{B}}_0} \bar{\boldsymbol{P}} : \delta \bar{\boldsymbol{F}} \mathrm{d}V = \int_{\partial \bar{\mathcal{B}}_0} \bar{\boldsymbol{t}}_0 \cdot \delta \bar{\boldsymbol{\varphi}} \mathrm{d}A \tag{2}$$

whereby $\bar{\boldsymbol{P}}$ denotes the Piola stress and $\bar{\boldsymbol{t}}_0$ the corresponding traction vector. Hereby the terms on the left represent the internal and those on the right the external virtual work, respectively. The resulting strong or rather local form of the balance of momentum and the corresponding boundary conditions read

$$\mathrm{Div}\, \bar{\boldsymbol{P}} = \boldsymbol{0} \quad \text{in } \bar{\mathcal{B}}_0 \qquad\qquad \bar{\boldsymbol{P}} \cdot \bar{\boldsymbol{N}} = \bar{\boldsymbol{t}}_0 \quad \text{on } \partial \bar{\mathcal{B}}_0 \tag{3}$$

for the considered static case. No constitutive assumption is made on the macro level, but rather obtained from the underlying micro structure.

2.2 Governing equations on the micro scale

The set of governing equations on the RVE level comprises both the continuum framework and a constitutive formulation.

2.2.1 Continuum framework

The classical continuum framework as introduced for the macro scale, is also adopted for the micro scale. Particularly, the deformation map $\boldsymbol{\varphi}$ and the deformation gradient $\boldsymbol{F}$ read

$$\boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{X}, t) \qquad\qquad \boldsymbol{F} = \nabla_{\boldsymbol{X}} \boldsymbol{\varphi}\,. \tag{4}$$

The balance of momentum and the corresponding boundary conditions

$$\mathrm{Div}\, \boldsymbol{P} = \boldsymbol{0} \quad \text{in } \mathcal{B}_0\,, \qquad\qquad \boldsymbol{P} \cdot \boldsymbol{N} = \boldsymbol{t}_0 \quad \text{on } \partial \mathcal{B}_0 \tag{5}$$

[1]The Dirichlet principle requires the energy density to be stationary for the system to be in equilibrium, i e., $\mathrm{D}_\delta(\int_{\bar{\mathcal{B}}_0} \bar{U}_0(\bar{\boldsymbol{\varphi}}, \bar{\boldsymbol{F}}; \bar{\boldsymbol{X}}) \mathrm{d}V - \int_{\partial \bar{\mathcal{B}}_0} \bar{v}_0(\bar{\boldsymbol{\varphi}}; \bar{\boldsymbol{X}})) = 0$, whereby D_δ denotes a variation with respect to all kinematic quantities at fixed material placement $\bar{\boldsymbol{X}}$. Furthermore U_0 denotes the total bulk energy density and v_0 the surface energy density.

have to be fulfilled for the RVE to be in static equilibrium. Herein $\boldsymbol{P}$ is the Piola stress within the RVE and $\boldsymbol{t}_0$ denotes the spatial traction vector on the material surface of the RVE. The weak form is stated in complete analogy to (2) as

$$\int_{\mathcal{B}_0} \boldsymbol{P} : \delta\boldsymbol{F}\mathrm{d}V = \int_{\partial\mathcal{B}_0} \boldsymbol{t}_0 \cdot \delta\boldsymbol{\varphi}\mathrm{d}A\,. \tag{6}$$

2.2.2 Constitutive formulation

Any suitable constitutive behaviour can be assigned with the micro level. In the present contribution for the sake of simplicity, we use a hyperelastic material which obeys the neo-Hooke formulation for the energy density

$$U_0(\boldsymbol{F};\boldsymbol{X}) = \tfrac{1}{2}[\lambda \ln^2(\det(\boldsymbol{F})) + \mu[\boldsymbol{F}:\boldsymbol{F} - n^{\dim} - 2\ln(\det(\boldsymbol{F}))]]\,, \tag{7}$$

wherein μ and λ are the Lamé material constants and $n^{\dim}$ denotes the spatial dimension of the problem. With this formulation the Piola stress $\boldsymbol{P} := \mathsf{D}_{\boldsymbol{F}} U_0$ can be evaluated.

2.3 Micro-to-macro transition

The relation between a macro continuum point and the underlying microstructure is based on the averaging of the decisive quantities over the corresponding representative volume element.

2.3.1 Averaging of quantities over the RVE

The volume averages over the RVE of the deformation gradient, the Piola stress, and the virtual work

$$\langle\boldsymbol{F}\rangle = \frac{1}{V}\int_{\mathcal{B}_0} \boldsymbol{F}\mathrm{d}V\,, \qquad \langle\boldsymbol{P}\rangle = \frac{1}{V}\int_{\mathcal{B}_0} \boldsymbol{P}\mathrm{d}V\,, \qquad \langle\boldsymbol{P}:\delta\boldsymbol{F}\rangle = \frac{1}{V}\int_{\mathcal{B}_0} \boldsymbol{P}:\delta\boldsymbol{F}\mathrm{d}V \tag{8}$$

are recalled here and transformed to boundary integrals[2]:

$$\langle\boldsymbol{F}\rangle = \frac{1}{V}\int_{\partial\mathcal{B}_0} \boldsymbol{\varphi}\otimes\boldsymbol{N}\mathrm{d}A\,, \quad \langle\boldsymbol{P}\rangle = \frac{1}{V}\int_{\partial\mathcal{B}_0} \boldsymbol{t}_0\otimes\boldsymbol{X}\mathrm{d}A\,, \quad \langle\boldsymbol{P}:\delta\boldsymbol{F}\rangle = \frac{1}{V}\int_{\partial\mathcal{B}_0} \boldsymbol{t}_0\cdot\delta\boldsymbol{\varphi}\,\mathrm{d}A\,. \tag{9}$$

These will be related to the macroscopic quantities next.

[2]The following relations are utilised to derive the representation of the volume averages as surface integrals over the boundary:

$$\begin{aligned}
\boldsymbol{F} &= \nabla_{\boldsymbol{X}}\boldsymbol{\varphi} = \nabla_{\boldsymbol{X}}\boldsymbol{\varphi}\cdot\boldsymbol{I} + \boldsymbol{\varphi}\cdot\mathrm{Div}\,(\boldsymbol{I}) = \mathrm{Div}\,(\boldsymbol{\varphi}\otimes\boldsymbol{I})\\
\boldsymbol{P}^t &= \boldsymbol{I}\cdot\boldsymbol{P}^t + \boldsymbol{X}\otimes\mathrm{Div}\,\boldsymbol{P} = \nabla_{\boldsymbol{X}}\boldsymbol{X}\cdot\boldsymbol{P}^t + \boldsymbol{X}\otimes\mathrm{Div}\,\boldsymbol{P} = \mathrm{Div}\,(\boldsymbol{X}\otimes\boldsymbol{P})\\
\boldsymbol{P}:\delta\boldsymbol{F} &= \mathrm{Div}\,(\boldsymbol{\varphi}\cdot\boldsymbol{P}) - \mathrm{Div}\,\boldsymbol{P}\cdot\delta\boldsymbol{\varphi} = \mathrm{Div}\,(\delta\boldsymbol{\varphi}\cdot\boldsymbol{P})\,.
\end{aligned}$$

2.3.2 Equivalence of averages and macro quantities

It is postulated that the macroscopic deformation gradient, Piola stress and virtual work are equivalent to the averages:

$$\bar{\boldsymbol{F}} \equiv \langle \boldsymbol{F} \rangle \,, \qquad \bar{\boldsymbol{P}} \equiv \langle \boldsymbol{P} \rangle \,, \qquad \bar{\boldsymbol{P}} : \delta\bar{\boldsymbol{F}} \equiv \langle \boldsymbol{P} : \delta\boldsymbol{F} \rangle \,. \tag{10}$$

With this framework at hand, boundary conditions stemming from the macroscopic quantities are imposed on the RVE using (9), whereby the Hill condition

$$\langle \boldsymbol{P} \rangle : \langle \delta\boldsymbol{F} \rangle \doteq \langle \boldsymbol{P} : \delta\boldsymbol{F} \rangle \tag{11}$$

must be fulfilled.

2.3.3 Boundary conditions on the RVE imposed by macro quantities

There are three ways to apply boundary conditions on the RVE based on the macroscopic quantities, which meet the Hill condition (11): The first is referred to as *linear displacement boundary conditions*, the second as an imposition of *periodic displacements and antiperiodic tractions*. A third option is to apply *constant traction* on the boundary. While the first two options are depicted in Figure 1, we omit the latter case here for the sake of brevity. They all are based on the assumption that the origin of the coordinate system is placed in the volumetric centre of the RVE. For linear displacement boundary conditions, the displacement on the RVE boundary is a linear mapping of the reference placement by means of the macro deformation gradient:

$$\boldsymbol{\varphi} = \bar{\boldsymbol{F}} \cdot \boldsymbol{X} \quad \text{on } \partial\mathcal{B}_0 \,. \tag{12}$$

In case of periodic boundary conditions, deformations of opposite boundaries allow for periodicity, while from equilibrium the tractions on opposite edges will be antiperiodic:

$$[\boldsymbol{\varphi}^+ - \boldsymbol{\varphi}^-] = \bar{\boldsymbol{F}} \cdot [\boldsymbol{X}^+ - \boldsymbol{X}^-] \,, \qquad \boldsymbol{t}_0^+ - \boldsymbol{t}_0^- = \boldsymbol{0} \quad \text{on } \partial\mathcal{B}_0^\pm \,. \tag{13}$$

Thereby the positive and negative boundaries, $\mathcal{B}_0^+$ and $\mathcal{B}_0^-$, respectively, are located on opposite edges of the RVE.

3 Computational Homogenisation

The homogenisation concept introduced previously is embedded into a nonlinear finite element framework. Particularly, the macro specimen is discretised by a finite element mesh. During the simulation of a boundary value problem, at each elemental integration point of each element, the material properties are evaluated by means of an underlying RVE along the lines of the contributions by Miehe, Geers, and others [MSS99, FC00, KBB01, KGB02]. Due to the geometrically and physically nonlinear framework, this multi-scale computation requires an iterative solution procedure on both the macro and the micro level, as we illustrate in the simplified flowchart of Figure 2.

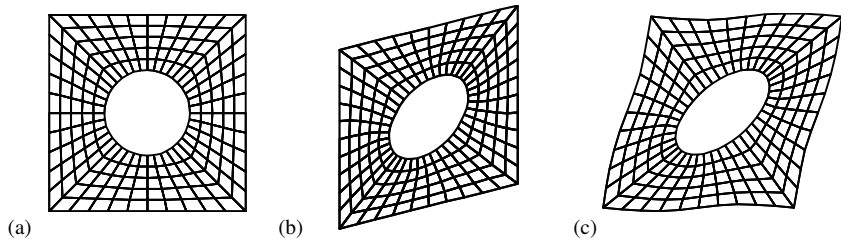


Figure 1: Representative volume elements: (a) undeformed mesh and deformed mesh under (b) linear displacement on the boundary, (c) periodic displacement on the boundary.

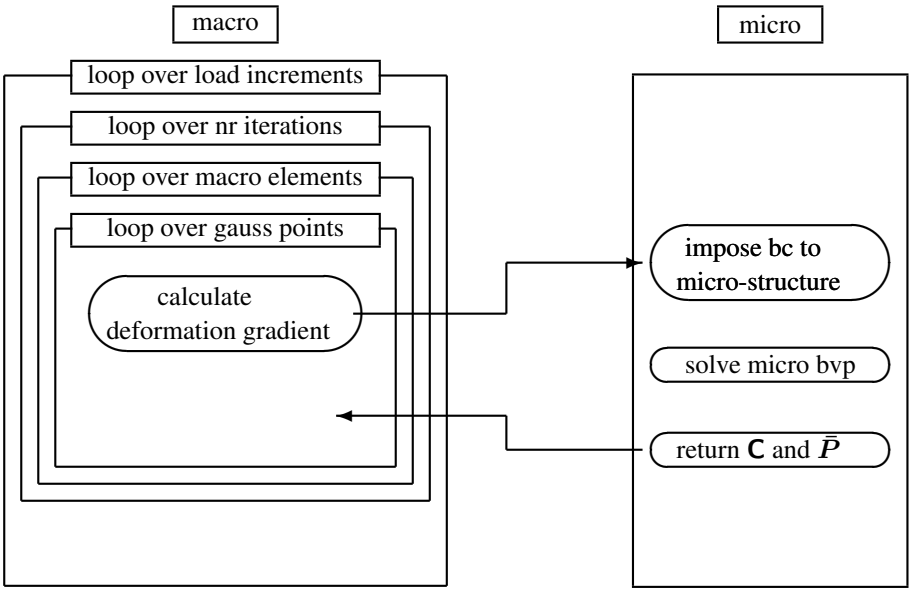


Figure 2: Simplified flow chart of computational micro–macro transition.

3.1 Application of the boundary conditions on the RVE

According to Section 2.3.3, different types of boundary conditions can be applied onto the RVE within the stepwise and iterative numerical evaluation of the particular RVE at a macroscopic integration point. In case of linear displacement on the boundary, the displacements of all boundary nodes are prescribed as shown in Figure 1(b). For the periodic case, only the displacement of three independent corner nodes are prescribed, see Figure 1(c).

3.2 Solution of the nonlinear system of equations of the RVE

With the respective boundary conditions, the micro system of equations is solved. In the case of fully prescribed boundary displacement, this is straightforward. With the displacement boundary condition stemming from the current increment of the macro deformation gradient, an iteration scheme (e. g. Newton-Raphson) is used to find the solution at (micro) equilibrium.

The linearised system of equations with the algorithmic tangent tensor $\mathbf{K}_{IK}$

$$\mathbf{K}_{IK} \cdot \Delta\boldsymbol{\varphi}_K = \boldsymbol{f}_I^{\text{ext}} - \boldsymbol{f}_I^{\text{int}}\,, \qquad \mathbf{K}_{IK} = \frac{\partial \mathbf{R}_I}{\partial \boldsymbol{\varphi}_{\text{K}}} \tag{14}$$

is solved such that the residual vector becomes zero:

$$\mathbf{R}_I = \mathop{\mathbf{A}}_{e=1}^{n_{\text{el}}} \int_{\mathcal{B}_0} \boldsymbol{P} \cdot \nabla_{\boldsymbol{X}} N_I - \boldsymbol{f}_I^{\text{ext}} \stackrel{!}{=} \mathbf{0}\,. \tag{15}$$

Note that for periodic displacement and anti-periodic traction on the boundary, the periodicity constraints must be imposed before the solution of the system is performed. We transform the system (14) towards a new system only formulated in terms of independent degrees of freedom

$$\mathbf{K}^\star \cdot \Delta\boldsymbol{\varphi}_i = \Delta\boldsymbol{f}^\star \tag{16}$$

as proposed by [Kou02]. Thereby the transformed stiffness matrix and the transformed incremental load vector read:

$$\mathbf{K}^\star = \mathbf{K}_{\text{ii}} + \mathbf{C}_{\text{di}}^{\text{t}} \cdot \mathbf{K}_{\text{id}} + \mathbf{K}_{\text{id}} \cdot \mathbf{C}_{\text{di}} + \mathbf{C}_{\text{di}}^{\text{t}} \cdot \mathbf{K}_{\text{dd}} \cdot \mathbf{C}_{\text{di}} \tag{17}$$

$$\Delta\boldsymbol{f}^\star = \Delta\boldsymbol{f}_{\text{i}} + \mathbf{C}_{\text{di}}^{\text{t}} \cdot \Delta\boldsymbol{f}_{\text{d}}\,, \tag{18}$$

which is based on the relation $\Delta\boldsymbol{\varphi}_{\text{d}} = \mathbf{C}_{\text{di}} \cdot \Delta\boldsymbol{\varphi}_{\text{i}}$ between the dependent and the independent degrees of freedom with the so-called dependency matrix $\mathbf{C}_{\text{di}}$.

3.3 Obtaining the macro stress and tangent operator

Once the solution has been obtained iteratively, the macroscopic stress and tangent operator are computed. With the equivalence $(10)_2$ the macro Piola stress $\bar{\boldsymbol{P}}$ is discretely evaluated as a summation over boundary nodes:

$$\Delta\bar{\boldsymbol{P}} = \frac{1}{V}\sum_I \Delta\boldsymbol{f}_I \otimes \boldsymbol{X}_I \,. \tag{19}$$

With (14), the tangent operator in the relation $\Delta\bar{\boldsymbol{P}} = \partial_{\bar{\boldsymbol{F}}}\bar{\boldsymbol{P}} : \Delta\bar{\boldsymbol{F}}$ is extracted as[3]

$$\bar{\mathsf{A}} := \partial_{\bar{\boldsymbol{F}}}\bar{\boldsymbol{P}} = \frac{1}{V}\sum_I\sum_K \mathsf{K}_{IK}\,\overline{\otimes}\,[\boldsymbol{X}_K \otimes \boldsymbol{X}_I] \,. \tag{20}$$

In the case of linear displacement boundary conditions, the summations over nodes (I) and (K) run over all boundary nodes, while for periodic boundary conditions, only the three independent boundary nodes are taken into account. In each of these cases a condensation is applied to the stiffness matrix at the solved state.

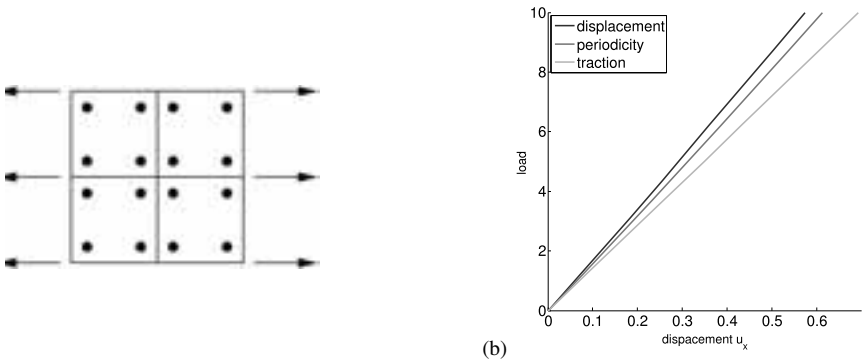
4 Numerical examples and visualisation

Based on the presented framework, some numerical examples are shown.

Figure 3 shows a benchmark-type problem. Hereby, the macro specimen is discretised with four finite elements, each of which possesses four integration points, which are depicted by the bullets. At each integration point an underlying RVE with a circular centered hole shown in Figure $1(a)$ is installed. Uniaxial loading is applied on the macro structure. The influence of different choices for the RVE boundary conditions (Section 3.1) on the macro response are examined. Thereby, the macroscopic load displacement curves reflect the fact that the RVE under displacement boundary conditions exhibits a stiffer behaviour than the RVE under periodic boundary conditions.

As a more realistic example, Figure 4 shows a rectangular specimen under uniaxial tension, the response of which is determined with the same underlying RVE, in particular a quadratic specimen which has a circular centered hole in the undeformed configuration. On the deformed macro mesh the Cauchy stress component $\bar{\sigma}_{22}$ is plotted. Furthermore, at four several placements of the macro specimen the deformed underlying micro structures are scaled into. Both the deformed shape and the distribution of the longitudinal normal component of the Cauchy stress, σ_{22}, strongly vary depending on the position at which the RVE is evaluated. Contrary to our last contribution [HKS06], here the Cauchy stress is a symmetric quantity due to the used standard continuum .

[3] The modified dyadic product of two second-order tensors is defined as $[\boldsymbol{A}\,\overline{\otimes}\,\boldsymbol{B}] : \boldsymbol{C} = \boldsymbol{A}\cdot\boldsymbol{C}\cdot\boldsymbol{B}^{\mathrm{t}}$.



(a) (b)

Figure 3: (a) Macro boundary value problem discretised with four elements with four integration points per element. (b) Comparison of macro load displacement curves for different RVE boundary conditions (reproduced in colorplate 193).

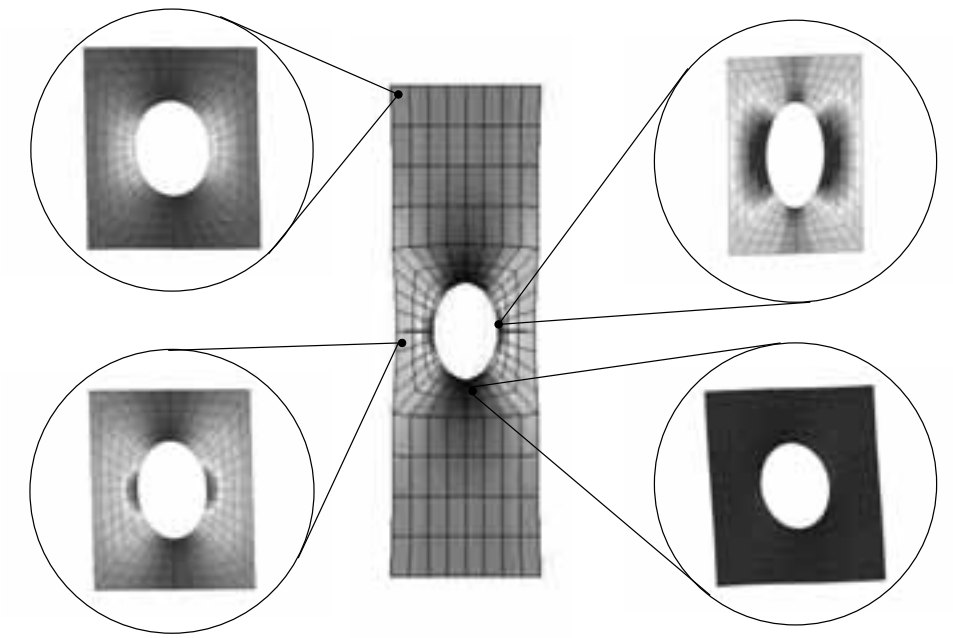


Figure 4: Deformed specimen with circular hole under longitudinal tension and deformed RVEs with logitudinal normal stress component $\bar{\sigma}_{22}$ and σ_{22}, respectively (reproduced in colorplate 193).

5 Conclusion

In this contribution, we have given a briew review on a finite-element-based multi scale method coined as computational homogenisation or rather FE^2. This approach is characterised by the solution of separate boundary value problems on both the macro and the

micro scale, which are linked at each integration point of each macro finite element, while the material behaviour is exclusively determined by the constitution and response of the micro structure.

While here we have chosen a continuous RVE with a phenomenological constitutive behaviour, one could also employ other material laws or other compositions. For instance inelastic formulations of standard continua [RMS07] or micromorphic continua [HS07] represent a significant challenge. A further extension of the methodology to obtain material properties of cohesive layers has recently been developed by the authors [HRSS07, HRSS08, HSS08]. In the contribution of [HSS08] a micromorphic continuum [HKS06, HKS07b], was employed at the micro level to account for a large intrinsic substructure underlying to the material layer. Thereby nonsymmetric stress measures as well as higher order tensors evolve, which from our perspective represent further challenges in the visualisation. Futhermore we would like to refer to the discrete element method which accounts for the interaction of single particles [MKS06, MKS07, MSK07], or atomistic models [SS03, SES07] and, if employed at the microstructural scale, can lead to realistic simulation of structures at the nano level.

Acknowledgement

We gratefully acknowledge financial support by the German Science Foundation (DFG) within the International Research Training Group 1131 "Visualization of Large and Unstructured Data Sets – Applications in Geospatial Planning, Modeling, and Engineering" as well as through the Research Training Group 814 "Engineering Materials on Different Scales – Experiment, Modelling, and Simulation".

References

[FC00] F. Feyel and J.-L. Chaboche. FE^2 multiscale approach for modelling the elastoviscoplastic behaviour of long fibre SiC/Ti composites materials. *Comput. Methods Appl. Mech. Engrg.*, 183:309–330, 2000.

[Has64] Z. Hashin. Theory of mechanical behaviour of heterogeneous media. *Appl. Mech. Rev.*, 17:1–9, 1964.

[Has83] Z. Hashin. Analysis of composite materials – A survey. *J. Appl. Mech.*, 50:481–505, 1983.

[Hil63] R. Hill. Elastic properties of reinforced solids: Some theoretical principles. *J. Mech. Phys. Solid.*, 11:357–372, 1963.

[Hil72] R. Hill. On constitutive macro-variables for heterogeneous solids at finite strain. *Proc. Roy. Soc. Lond. A*, 326:131–147, 1972.

[HKS06] C. B. Hirschberger, E. Kuhl, and P. Steinmann. Computational modelling of micromorphic continua – theory, numerics, and visualisation challenges. In H. Hagen, A. Kerren,

and P. Dannenmann, editors, *Visualization of large and unstructured data sets*, volume S-4 of *GI-Edition Lecture Notes in Informatics (LNI)*, pages 155–164, 2006.

[HKS07a] C. B. Hirschberger, E. Kuhl, and P. Steinmann. Computational material forces in micromorphic continua. *Proc. Appl. Math. Mech.*, 6:379–380, 2007.

[HKS07b] C. B. Hirschberger, E. Kuhl, and P. Steinmann. On deformational and configurational mechanics of micromorphic hyperelasticity – theory and computation. *Comput. Methods Appl. Mech. Engrg.*, 196:4027–4044, 2007.

[HKS07c] C. B. Hirschberger, E. Kuhl, and P. Steinmann. Recent aspects in the mechanics of micromorphic continua. In *Proceedings of the International Workshop on Nonlocal Modeling of Materials Failure, Wuppertal*, pages 3–14. Aedificatio Publishers, 2007.

[HRSS07] C. B. Hirschberger, S. Ricker, P. Steinmann, and N. Sukumar. A computational homogenisation approach for cohesive interfaces. In D. R. J. Owen and E. O nate, editors, *Computational Plasticity IX: Fundamentals and Applications*, pages 442–445. CIMNE, 2007.

[HRSS08] C. B. Hirschberger, S. Ricker, P. Steinmann, and N. Sukumar. On the computational homogenization of a heterogeneous material layer. *Eng. Fract. Mech.*, submitted, 2008.

[HS62] Z. Hashin and S. Shtrikman. On some variational principles in anisotropic and nonhomogeneous elasticity. *J. Mech. Phys. Solid.*, 10:335–342, 1962.

[HS07] C. B. Hirschberger and P. Steinmann. Classification of concepts in thermodynamically consistent generalized plasticity. *J. Engineering Mech.*, submitted, 2007.

[HSS08] C. B. Hirschberger, N. Sukumar, and P. Steinmann. Computational homogenization of material layers with micromorphic mesostructure. *Phil. Mag.*, submitted, 2008.

[KBB01] V. G. Kouznetsova, W. A. M. Brekelmans, and F. P. T. Baaijens. An approach to micro-macro modeling of heterogeneous materials. *Comput. Mech.*, 27:37–48, 2001.

[KGB02] V. G. Kouznetsova, M. G. D. Geers, and W. A. M. Brekelmans. Multi-scale constitutive modelling of heterogeneous materials with a gradient-enhanced computational homogenization scheme. *Int. J. Numer. Meth. Engng*, 54:1235–1260, 2002.

[Kou02] V. G. Kouznetsova. *Computational Homogenization for the Multiscale Analysis of Multi-Phase Materials*. PhD thesis, Technische Universiteit Eindhoven, 2002.

[MD04] C. Miehe and J. Dettmar. A framework for micro-to-macro transitions in periodic particle aggregates of granular materials. *Comput. Methods Appl. Mech. Engrg.*, 193:225–256, 2004.

[Mie03] C. Miehe. Computational micro-to-macro transitions discretized micro-structures of heterogeneous materials at finite strains based on the minimization of averaged incremental energy. *Comput. Methods Appl. Mech. Engrg.*, 192:559–591, 2003.

[MK02] C. Miehe and A. Koch. Computational micro-to-macro transitions of discretized microstructures undergoing small strains. *Arch. Appl. Mech.*, 72:300–317, 2002.

[MKS06] H. A. Meier, E. Kuhl, and P. Steinmann. On discrete modeling and visualization of granular media. In H. Hagen, A. Kerren, and P. Dannenmann, editors, *Visualization of Large and Unstructured Data Sets*, volume S-4, pages 165–175. Lecture Notes in Informatics, 2006.

[MKS07] H. A. Meier, E. Kuhl, and P. Steinmann. A note on the generation of periodic granular microstructures based on grain size distributions. *Int. J. Numer. Anal. Meth. Geomech.*, 2007. in press.

[MMS99] J. C. Michel, H. Moulinec, and P. Suquet. Effective properties of composite materials with periodic microstructure: a computational approach. *Comput. Methods Appl. Mech. Engrg.*, 172:109–143, 1999.

[MMS01] J. C. Michel, H. Moulinec, and P. Suquet. A computational scheme for linear and non-linear composites with arbitrary phase contrast. *Int. J. Numer. Meth. Engng*, 52:139–160, 2001.

[MSK07] H. A. Meier, P. Steinmann, and E. Kuhl. Towards multiscale computation of confined granular media – contact forces, stresses and tangent operators. *Technische Mechanik*, 2007. submitted.

[MSS99] C. Miehe, J. Schotte, and J. Schröder. Computational micro-macro transitions and overall moduli in the analysis of polycrystals at large strains. *Comput. Mater. Sci.*, 16:372–382, 1999.

[NNH99] A. Nemat-Nasser and M. Hori. *Micromechanics: Overall Properties of Heterogeneous Materials*. North-Holland Elsevier, 2nd revised edition, 1999.

[RMS07] S. Ricker, A. Menzel, and P. Steinmann. Computational homogenization for discrete micro-structures including damage. In M. Jirásek, Z. Bittnar, and H. Mang, editors, *Modelling of Heterogeneous Materials*, pages 182–183, Prague, 2007.

[SBM98] R. J. M. Smit, W. A. M. Brekelmans, and H. E. H. Meijer. Prediction of the mechanical behaviour of nonlinear heterogeneous systems by multi-level finite element modeling. *Comput. Methods Appl. Mech. Engrg.*, 155:181–192, 1998.

[SES07] P. Steinmann, A. Elizondo, and R. Sunyk. Studies of validity of the Cauchy-Born rule by direct comparison of continuum and atomistic modelling. *Modelling Simul. Mater. Sci. Eng.*, 15:S271–S281, 2007.

[SS03] R. Sunyk and P. Steinmann. On higher gradients in continuum-atomistic modelling. *Int. J. Solid Struct.*, 40:6877–6896, 2003.

[THKK00] K. Terada, M. Hori, T. Kyoya, and N. Kikuchi. Simulation of the multi-scale convergence in computational homogenzation approaches. *Int. J. Solid Struct.*, 37:2285–2311, 2000.

[TIK98] K. Terada, T. Ito, and N. Kikuchi. Characterization of the mechanical behaviors of solid-fluid mixture by the homogenization method. *Comput. Methods Appl. Mech. Engrg.*, 153:233–257, 1998.

[TK01] K. Terada and N. Kikuchi. A class of general algorithms for multi-scale analyses of heterogeneous media. *Comput. Methods Appl. Mech. Engrg.*, 190:5427–5464, 2001.

[Wal66] L. J. Walpole. On bounds for the overall elastic moduli of inhomogeneous systems – I. *J. Mech. Phys. Solid.*, 14:151–162, 1966.

[Wil77] J. R. Willis. Bounds and self-consistent estimates for the overall properties of composites. *J. Mech. Phys. Solid.*, 25:185–202, 1977.

[Wil81] J. R. Willis. Variational and related methods for the overall properties of anisotropic composites. *Adv. Appl. Mech.*, 21:1–78, 1981.

Finite Elasto-Plasto-Dynamics – Challenges & Solutions

Rouven Mohr[1], Andreas Menzel[2,3], and Paul Steinmann[4]

[1] University of Kaiserslautern
Chair of Applied Mechanics
Department of Mechanical and Process Engineering
D-67653 Kaiserslautern, Germany
rmohr@rhrk.uni-kl.de

[2] Dortmund University of Technology
Group of Mechanics and Machine Dynamics
Faculty of Mechanical Engineering
D-44227 Dortmund, Germany
andreas.menzel@udo.edu

[3] Lund University
Division of Solid Mechanics
SE-22100 Lund, Sweden
andreas.menzel@solid.lth.se

[4] University of Erlangen-Nuremberg
Institute of Applied Mechanics
Department of Mechanical Engineering
D-91058 Erlangen, Germany
paul.steinmann@ltm.uni-erlangen.de

Abstract: In this contribution, we deal with time-stepping schemes for geometrically nonlinear multiplicative elasto-plasto-dynamics. Thereby, the approximation in space as well as in time rely both on a Finite Element approach, providing a general framework which conceptually includes also higher-order schemes. In this context, the algorithmic conservation properties of the related integrators strongly depend on the numerical computation of time integrals, particularly, if plastic deformations are involved. However, the application of adequate quadrature rules enables a fulfilment of physically motivated balance laws and, consequently, the consistent integration of finite elasto-plasto-dynamics. Using exemplarily linear Finite Elements in time, the resulting integration schemes are analysed regarding the obtained conservation properties and assessed in comparison to classical time-stepping schemes which commonly adopt a time-discretisation procedure based on Finite Differences.

1 Introduction

On the one hand, computational modelling of materials and structures often demands the incorporation of inelastic and dynamic effects. On the other hand, the performance of classical time integration schemes for structural dynamics, as for instance developed in [HHT77, New59], is strongly restricted when dealing with highly nonlinear systems. In a nonlinear setting, advanced numerical techniques are required to satisfy the classical balance laws as for instance balance of linear and angular momentum or the classical laws of thermodynamics. Nowadays, energy and momentum conserving time integrators for dynamical systems, like multibody systems or elasto-dynamics, are well-established in the computational dynamics community, compare e.g. [BB99, BBT01a, BBT01b, Gonz00, KC99, ST92]. In contrast to the commonly used time discretisation based on Finite Differences, one-step implicit integration algorithms relying on Finite Elements in space and time were developed, for instance, in Betsch and Steinmann [BS00a, BS00b, BS01]. Therein, conservation of energy and angular momentum have been shown to be closely related to quadrature formulas required for numerical integration in time. Furthermore, specific algorithmic energy conserving schemes for hyperelastic materials can be based on the introduction of an enhanced stress tensor for time shape functions of arbitrary order, compare Gross *et al.* [GBS05]. However, most of the proposed approaches are restricted to conservative dynamical systems. Nevertheless, the consideration of plastic deformations in a dynamical framework, involving dissipation effects, is of cardinal importance for various applications in engineering. In the last years, notable contributions dealing with finite elasto-plasto-dynamics have been published by Meng and Laursen [ML02a, ML02b], Noels *et al.* [NSP06] and Armero [Arm05, Arm06, AZ06]. In this contribution, we follow the concepts which have been proposed for hyperelasticity in [BS01, GBS05] and pick-up the general framework of Galerkin methods in space and time, developing integrators for finite multiplicative elasto-plasto-dynamics with pre-defined conservation properties, compare Mohr *et al*. [MMS06a, MMS07c, MMS07a, MMS07b]. By means of a representative numerical example, the excellent performance of the resulting schemes, which base on linear Finite Elements in time combined with different quadrature rules, will be demonstrated and compared with the performance of well-accepted standard integrators.

2 Semi-Discrete Dynamics

To set the stage, we start with some basic notation of geometrically nonlinear continuum mechanics. First, the nonlinear deformation map $\boldsymbol{\varphi}(\boldsymbol{X}, t) : \mathcal{B}_0 \times [0, T] \rightarrow \mathcal{B}_t$ shall be introduced as a mapping from the reference to the spatial configuration. As proposed by Lee [Lee69], the resulting deformation gradient $\boldsymbol{F} := \nabla_{\boldsymbol{X}} \boldsymbol{\varphi}(\boldsymbol{X}, t)$ is assumed to be multiplicatively decomposed into an elastic and a plastic part

$$\boldsymbol{F} \doteq \boldsymbol{F}_e \cdot \boldsymbol{F}_p \,, \tag{1}$$

implying an additional intermediate configuration related to purely plastic deformations, as illustrated in Fig. 1. In contrast to the modelling of elasticity, additional internal variables

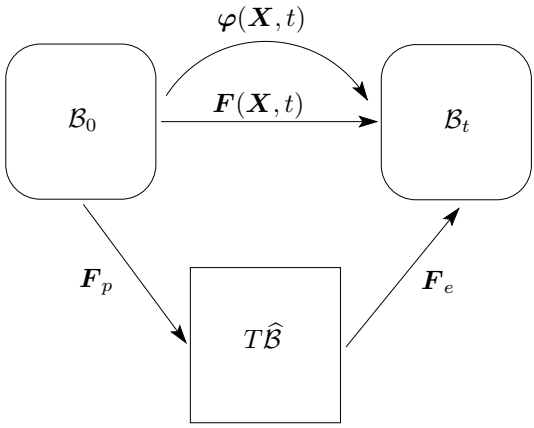


Figure 1: Resulting configurations within the framework of finite multiplicative plasticity

are included in the Helmholtz energy density ψ for the plastic case to model the loading history. By assuming an additive structure and elastic isotropy, the Helmholtz energy density ψ can be written in terms of the eigenvalues ${}^{i}\lambda_{\widehat{\boldsymbol{C}_e}}$ of the elastic right Cauchy-Green strain tensor

$$\widehat{\boldsymbol{C}_e} := \boldsymbol{F}_e^{\mathrm{t}} \cdot \boldsymbol{F}_e \tag{2}$$

of the intermediate configuration and the internal variables $\boldsymbol{\kappa} = [\boldsymbol{\kappa}^{mac}, \boldsymbol{\kappa}^{mic}]$

$$\psi(\boldsymbol{F}, \boldsymbol{\kappa}) = \psi^{mac}({}^{1}\lambda_{\widehat{\boldsymbol{C}_e}}, {}^{2}\lambda_{\widehat{\boldsymbol{C}_e}}, {}^{3}\lambda_{\widehat{\boldsymbol{C}_e}}) + \psi^{mic}(\boldsymbol{\kappa}^{mic})\,. \tag{3}$$

For thermodynamical aspects, it is accepted to introduce the so-called conjugated thermodynamical forces $\boldsymbol{\beta} := -\nabla_{\boldsymbol{\kappa}}\psi$. To differ between elastic and plastic deformation states, the yield function Φ is introduced which defines the elastic range $\mathbb{E}_{\boldsymbol{\beta}} := \{\boldsymbol{\beta} \,|\, \Phi(\boldsymbol{\beta}; \boldsymbol{\kappa}) < 0\}$. In view of thermodynamically consistent modelling, the dissipation inequality – corresponding to the second law of thermodynamics – should be fulfilled, namely

$$\mathcal{D} = \left\langle \boldsymbol{\beta}, \dot{\boldsymbol{\kappa}} \right\rangle \quad \geq 0\,. \tag{4}$$

For more detailed information, we refer to Simo [Sim98]. Subsequently, we apply a standard Finite Element discretisation in space for the reference configuration of a solid continuum body. Using the spatial approximations for the nonlinear deformation map, the semi-discrete map $\boldsymbol{\varphi} : \mathcal{B}_0 \times [0, T] \rightarrow \mathbb{R}^{n_{dim}}$ can be written by means of the spatial shape functions $N_A(\boldsymbol{X})$ in the form:

$$\boldsymbol{\varphi}(\boldsymbol{X}, t) \approx \sum_{A=1}^{n_{node}} \boldsymbol{q}_A(t)\, N_A(\boldsymbol{X}) \tag{5}$$

Consequently, the approximations in space of the spatial velocity $\boldsymbol{v} := \sum_{A=1}^{n_{node}} \dot{\boldsymbol{q}}_A \, N_A$, the deformation gradient $\boldsymbol{F} = \sum_{A=1}^{n_{node}} \boldsymbol{q}_A \otimes \nabla_{\boldsymbol{X}} N_A$, and the right Cauchy-Green strain tensor

$$\boldsymbol{C} := \boldsymbol{F}^{\mathrm{t}} \cdot \boldsymbol{F} = \sum_{A,B=1}^{n_{node}} \boldsymbol{q}_A \cdot \boldsymbol{q}_B \, \nabla_{\boldsymbol{X}} N_A \otimes \nabla_{\boldsymbol{X}} N_B \tag{6}$$

can be computed straightforwardly. To obtain a semi-discrete system of equations of motion, we combine the placements of the spatial nodes $\boldsymbol{q} = [\boldsymbol{q}_1, ..., \boldsymbol{q}_{n_{node}}]^{\mathrm{t}}$ and the nodal generalised momenta $\boldsymbol{p} := \mathbb{M} \cdot \dot{\boldsymbol{q}} = [\boldsymbol{p}_1, ..., \boldsymbol{p}_{n_{node}}]^{\mathrm{t}}$ to a vector of the canonical phase-space variables $\boldsymbol{z} := [\boldsymbol{q}, \boldsymbol{p}]^{\mathrm{t}}$. Furthermore, the kinetic and free energy

$$T(\boldsymbol{p}) = \frac{1}{2} \, \boldsymbol{p} \cdot \mathbb{M}^{-1} \cdot \boldsymbol{p} \qquad \text{respectively} \qquad \Psi = \int_{\mathcal{B}_0} \psi \, \mathrm{d}V \tag{7}$$

are introduced for the semi-discrete case. Motivated by the well-known Hamiltonian H for conservative systems, the sum consisting of kinetic energy, free energy, and possibly an external potential V^{ext} will be denoted by

$$H(\boldsymbol{q}, \boldsymbol{p}; \boldsymbol{\kappa}) = T + \Psi + V^{ext} \, . \tag{8}$$

Dealing with dissipative systems, the global accumulated dissipation

$$D := \int_{\mathcal{B}_0} d \, \mathrm{d}V \qquad \text{based on} \qquad d := \int_0^t \mathcal{D} \, \mathrm{d}t \tag{9}$$

has to be taken into account. Analogously to the purely elastic case, the resulting (canonical) equations of motion can still be written in a compact format of the Hamiltonian-type

$$\dot{\boldsymbol{z}}(t) = \mathbb{J} \cdot \nabla_{\boldsymbol{z}} H(\boldsymbol{z}; \boldsymbol{\kappa}) \qquad \text{with} \qquad \mathbb{J} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{I} \\ -\boldsymbol{I} & \boldsymbol{0} \end{bmatrix} , \tag{10}$$

representing a system of ordinary differential equations of first order. Therein, the gradient with respect to the global variables $\boldsymbol{z}$ can be specified by

$$\nabla_{\boldsymbol{z}} H = \begin{bmatrix} \boldsymbol{F}^{int} - \boldsymbol{F}^{ext} \\ \\ \mathbb{M}^{-1} \cdot \boldsymbol{p} \end{bmatrix} , \tag{11}$$

involving the definition of the internal load vector

$$\boldsymbol{F}^{int}(\boldsymbol{S}) := \nabla_{\boldsymbol{q}} \int_{\mathcal{B}_0} \psi \, \mathrm{d}V \tag{12}$$

based on the Piola Kirchhoff stresses $\boldsymbol{S} = 2 \, \nabla_{\boldsymbol{C}} \, \psi$. In the following, especially this highly nonlinear internal load vector plays a crucial role with regard to the discretisation in time and, consequently, concerning the resulting conservation properties of the Galerkin-based time-stepping schemes.

3 Discretisation in Time

As mentioned above, in the proposed concept not only the approximation in space but also the approximation in time relies on a Finite Element approach, see also [BS00b, Bot97, EEHJ96]. We start with a decomposition of the time interval $[0,T] = \bigcup_{n=0}^{N} [t_n, t_{n+1}]$ and a map of each sub-interval to the reference time interval $[0,1]$ via the function $\alpha(t) := [t - t_n]/h_n$, involving the time-step size $h_n = t_{n+1} - t_n$. For the time approximation, a continuous Galerkin method (or short: a 'cG(k) method') shall be applied. Therefore, the approximations in time for the unknown and the test function

$$\boldsymbol{z}^h = \sum_{j=1}^{k+1} M_j(\alpha)\, \boldsymbol{z}_j \qquad\qquad \delta\boldsymbol{z}^h = \sum_{i=1}^{k} \widetilde{M}_i(\alpha)\, \delta\boldsymbol{z}_i \tag{13}$$

are introduced [1]. In a compact notation the resulting weak form in time is given by the integral

$$\int_0^1 \left[\mathbb{J} \cdot \delta\boldsymbol{z}^h\right] \cdot \left[\mathrm{D}_\alpha \boldsymbol{z}^h - h_n\, \mathbb{J} \cdot \nabla_{\boldsymbol{z}} H(\boldsymbol{z};\boldsymbol{\kappa})\right] \mathrm{d}\alpha = 0\,. \tag{14}$$

Inserting the approximations in time (13) in Eq. (14) renders the discrete system of equations of motion

$$\sum_{j=1}^{k+1} \int_0^1 \widetilde{M}_i M_j^{'}\, \mathrm{d}\alpha\; \boldsymbol{z}_j - h_n \mathbb{J} \cdot \int_0^1 \widetilde{M}_i\, \nabla_{\boldsymbol{z}} H(\boldsymbol{z};\boldsymbol{\kappa})\, \mathrm{d}\alpha = \boldsymbol{0} \qquad \forall\ i = 1, ..., k \tag{15}$$

involving a time-integrated internal load vector. Thereby, higher-order integration schemes are included within this general framework. Further details can be found in Betsch and Steinmann [BS01]. Dealing with time-stepping schemes, special emphasis should be always placed on resulting algorithmic conservation properties which strongly influence the numerical performance. In this context, it is often desirable to transfer as many as possible of the conservation properties from the continuous to the completely discrete system, especially, regarding the underlying physics and the robustness of the related integration scheme. For the proposed concept, the approximation of the time-integrated internal load vector

$$\bar{\boldsymbol{F}}_i^{int} := \int_0^1 \widetilde{M}_i(\alpha)\, \boldsymbol{F}^{int}(\boldsymbol{S})\, \mathrm{d}\alpha \qquad \text{with} \qquad i = 1, ..., k \tag{16}$$

is the crux concerning the offered conservation properties of Galerkin-based time-stepping schemes. In fact, integrators with pre-defined conservation properties can be designed, adjusting the applied quadrature rule.

[1] It is important to emphasise that the time shape functions $M_j \in \mathcal{P}^k$ are polynomials of degree k, whereas the reduced shape functions $\widetilde{M}_i \in \mathcal{P}^{k-1}$ are only of degree $k-1$.

3.1 Standard quadrature

Involving $\boldsymbol{I} = \sum_{A=1}^{n_{node}} \boldsymbol{p}_A$ and $\boldsymbol{L} = \sum_{A=1}^{n_{node}} \boldsymbol{q}_A \times \boldsymbol{p}_A$, the application of a standard Gauss quadrature rule for the approximation of the time integral given by Eq. (16) already enables the conservation of both momentum maps (for vanishing external loads)

$$\boldsymbol{I}_{\alpha=1} - \boldsymbol{I}_{\alpha=0} = \boldsymbol{0} \qquad\qquad \boldsymbol{L}_{\alpha=1} - \boldsymbol{L}_{\alpha=0} = \boldsymbol{0}\,, \tag{17}$$

if an adequate number of integration points in time is incorporated, compare Betsch and Steinmann [BS01]. Nevertheless, an additional conservation of the total energy for elastic deformations, for instance, cannot be generally offered by means of such a standard quadrature rule. Moreover, the incorporation of physical dissipation effects related to plastic deformations poses further challenges for the applied quadrature rule. To tackle potential problems concerning algorithmic conservation properties, nonstandard quadrature rules must be taken into account.

3.2 Nonstandard quadrature

First, a nonstandard quadrature rule is introduced to guarantee the thermodynamical consistency of the integrator, in addition to the mechanical consistency which includes the conservation of both momentum maps corresponding to Eq. (17). Thermodynamically consistent integrators for elasto-plasto-dynamics are characterised by the conservation of the total energy in the elastic case

$$H_{\alpha=1} - H_{\alpha=0} = 0 \qquad \text{with} \qquad \Delta D = 0 \tag{18}$$

and by the (strictly) positive dissipation for plastic deformations represented by

$$H_{\alpha=1} - H_{\alpha=0} < 0 \qquad \text{combined with} \qquad \Delta D > 0\,, \tag{19}$$

compare Mohr *et al.* [MMS06a]. Finally, the application of an appropriate nonstandard quadrature rule renders energy-consistent Galerkin-based time-stepping schemes [2]. Thereby, energy-consistency includes the conservation of the sum consisting of the total energy and the dissipation represented by

$$H_{\alpha=1} - H_{\alpha=0} = -\Delta D \tag{20}$$

for elastic ($\Delta D = 0$) as well as for plastic ($\Delta D > 0$) deformations, whereby the fulfilment of Eq. (20) can be offered within the calculation accuracy. Detailed background informations concerning the consistency of a Galerkin-based integration of finite elasto-plasto-dynamics and corresponding technical details, like for instance the local integration of the plastic evolution equations, are presented in Mohr *et al.* [MMS07c].

[2] In the following, the thermodynamically consistent and mechanically consistent cG method will be referred to as 'TCMC-cG method' and the energy-consistent, mechanically consistent scheme as 'ECMC-cG method' to abbreviate the notation.

4 Numerical Examples

In this section, the performance of the outlined Galerkin-based concepts is analysed by means of a representative numerical example, evaluating the featured algorithmic conservation properties for the elastic as well as for the plastic case. As an appropriate example, the free motion of a 'Flying Frame' with the mass density ρ, consisting of 48 isoparametric 4-node elements in space, will be investigated. To start the free flight, the frame is equipped with a given initial velocity $||\boldsymbol{v}_0|| = 85$ and, furthermore, some external loads $\boldsymbol{F}^{ext}$ are applied during the loading period T_{load}. Thereby, the norm of the external load vector $||\boldsymbol{F}_A^{ext}|| = f(t)$ (at the spatial node A) is prescribed by the piecewise linear function $f(t)$ with the maximum value f_{max}, as illustrated in Fig. 2 a). For the following computations, linear Finite Elements in time have been applied as a fundamental example. The constitutive law relies on a Helmholtz energy density ψ of the Hencky-type with the material parameters λ, μ and on a v. Mises-type yield function Φ, involving the yield stress Y_0 and the modulus of linear isotropic hardening H.

4.1 Standard vs. nonstandard quadrature

First, the differences between the abovementioned standard and nonstandard quadrature rule for the approximation of the time-integrated internal load vector (16) will be studied in detail, regarding the purely elastic as well as the plastic case. In this context, a standard cG(1) method is compared with the 'TCMC-cG(1) method' respectively the 'ECMC-cG(1) method'. For the present computations, we have incorporated the parameters $\lambda = 10\,000$, $\mu = 5000$, $Y_0 = \infty|500$, $H = 500$, $\rho = 5.0$, $f_{max} = 100$, $T_{load} = 1.0$, and $h_n = 0.1$. Some snapshots of the motion are pictured in Fig. 2 b), whereby the plastic motion is displayed below the elastic case. The global accumulated dissipation D is, as

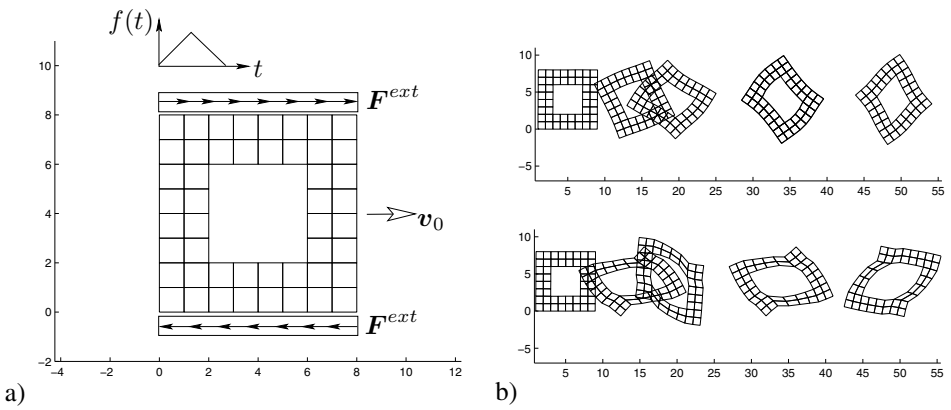


Figure 2: Configurations: a) initial mesh, b) sequence of the motion (elastic & plastic case) for $t \in \{0.1, 10, 15, 30, 45\}$

expected, equal to zero for each of the considered Galerkin-based integrators if the deformations are purely elastic, and it is (strictly) positive if plastic deformations are involved due to the application of an adequate local update algorithm for the plastic variables, compare Fig. 4. Furthermore, it can be clearly seen in Fig. 6 that mechanical consistency, related to a conservation of both momentum maps, can be guaranteed not only for time-stepping schemes which base on nonstandard quadrature rules, but also for the classical cG method which adopts a standard Gauss quadrature rule. Nevertheless, the influence of the applied quadrature rule becomes obvious when a plot of the total energy H is considered, as displayed in Fig. 3. In the elastic case, the total energy calculated by means of the standard cG method is characterised by strong oscillations, whereas the 'TCMC-cG method' as well as the 'ECMC-cG method' guarantee both the conservation of the total energy. Moreover, the cG method features an unphysical increase of the total energy in the plastic case. Contrariwise, both schemes based on nonstandard quadrature rules guarantee a monotonic decrease of the total energy caused by the (strictly) non-negative plastic dissipation. However, a physically correct decrease of the total energy H respectively increase of the dissipation D – related to a conservation of the augmented Hamiltonian $H + D$ – is guaranteed exclusively by the 'ECMC-cG method', as illustrated in Fig. 5.

4.2 Standard vs. nonstandard integrator

Finally, the performance of the proposed Galerkin-based integration schemes, represented by the 'ECMC-cG method', will be compared with the results calculated by means of standard time integration schemes. In this context, we have exemplarily chosen two of the most-established integrators at all: the classical Newmark scheme with $\gamma = 0.5$, $\beta = 0.25$ and the Hilber-Hughes-Taylor method (abbreviated by 'HHT method') with the parameters $\gamma = 0.8$, $\beta = 0.422$, and $\alpha = -0.30$, as proposed in [Hug87]. Both standard integrators have been developed originally for linear dynamical systems, whereby

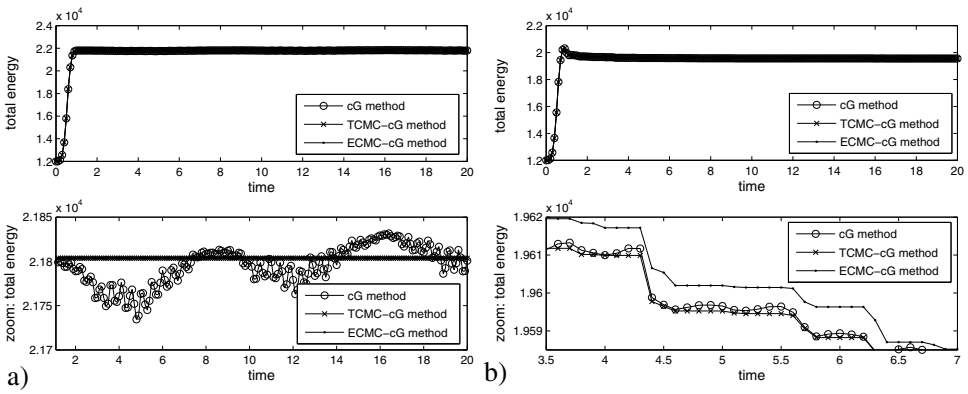


Figure 3: Total energy H: a) purely elastic case, b) plastic case

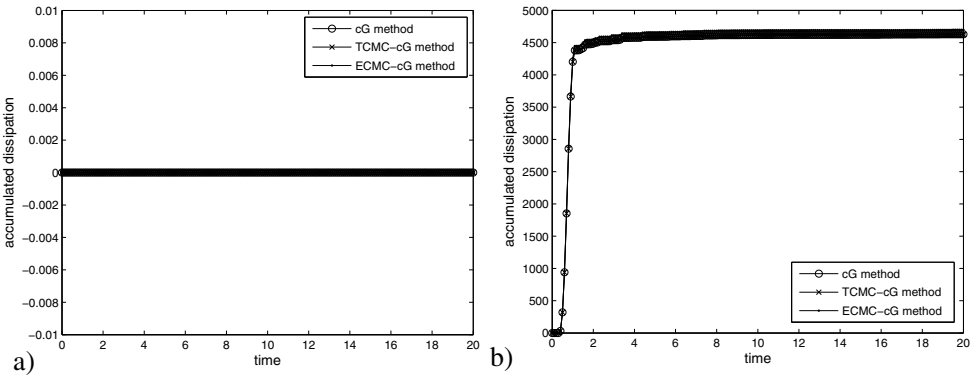


Figure 4: Accumulated dissipation D: a) purely elastic case, b) plastic case

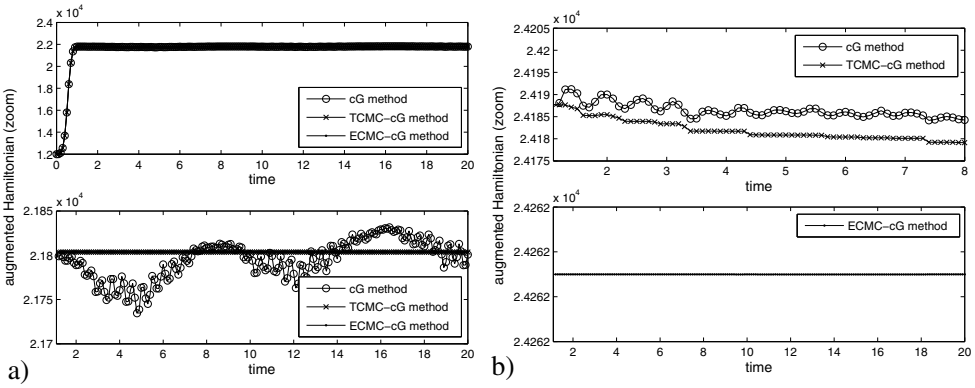


Figure 5: Augmented Hamiltonian $H + D$: a) purely elastic case, b) plastic case

for the chosen parameters the Newmark scheme is related to the (undamped) trapezoidal rule and the 'HHT method' has been specifically designed to provide a numerically dissipative behaviour. Once more, the 'Flying Frame' has been used for the computations, whereby the initial setup remains unchanged, only the applied parameters have been altered: $\lambda = 500$, $\mu = 250$, $Y_0 = 40$, $H = 100$, $\rho = 3.0$, $f_{max} = 20$, and $T_{load} = 1.0$. Moreover, small as well as large time-step sizes have been applied to assess fairly the offered performance of the different time-stepping schemes and, additionally, the time-step sizes have been changed during the calculation, namely from $h_n = 0.02$ to $h_n = 0.06$ respectively from $h_n = 0.1$ to $h_n = 0.3$ after $t = 2.2$, to check the robustness of the integrators. A sequence of the motion, including contour plots of the hardening variable, can be regarded in Fig. 11 and the local evolution of the hardening parameter is pictured exemplarily for two points of the frame in Fig. 12. However, in the following, special emphasis is placed on the resulting consistency properties. Considering Fig. 8, the accumulated dissipation D is, once more, (strictly) non-negative due to the chosen local update and the results of the different integrators are qualitatively similar, especially if small time-step

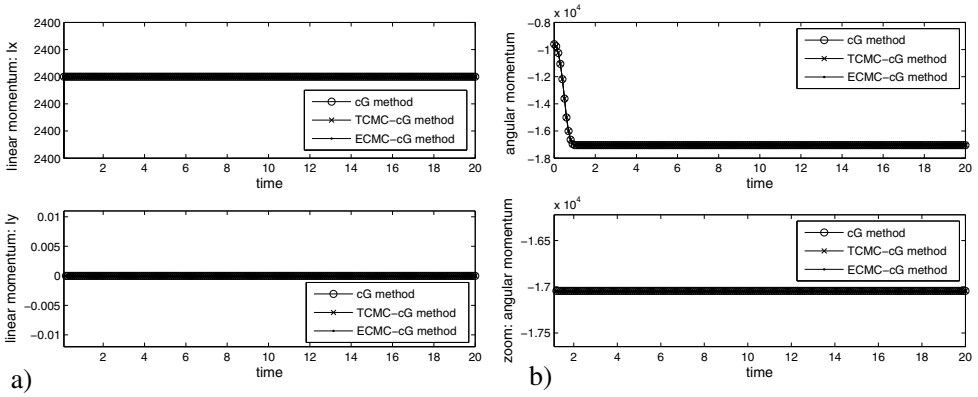


Figure 6: Momentum maps: a) linear momentum $\boldsymbol{I}$ (elastic case), b) component of the angular momentum $\boldsymbol{L}$ (plastic case)

sizes are applied. Nevertheless, the results differ quantitatively, whereby the application of the 'HHT method' results in the lowest physical dissipation, which seems to be significantly underestimated when dealing with large time-step sizes. Moreover, the differences in the total energy H between the classical integrators and the Galerkin-based 'ECMC-cG method' are quite impressive, compare Fig. 7. It can be clearly seen that the standard integrators are not able at all to feature a physically correct monotonic decrease of the total energy. Rather, both classical integration schemes suffer from oscillations in the total energy which increase drastically for large time-step sizes, especially if the widespread Newmark method is applied. One might think that such aspects are only of theoretical interest, but in fact, on the one hand, a physically correct time integration is essential for a qualitatively correct simulation of the dynamical behaviour and, on the other hand, the evolution of the total energy is directly related to the robustness of the integrators. This accepted fact is also confirmed by the present example, since the application of the Newmark method results in a critical energy blow-up when large time-step sizes are taken into account [3], compare Fig. 7 b). Contrariwise, the 'HHT method' offers an unphysical decrease of the total energy caused by its numerically dissipative character. Please note furthermore, that this strong decrease of the total energy is not accompanied by a high physical dissipation D, as discussed above. To investigate the relation between the decrease of the total energy and the increase of the accumulated (physical) dissipation in detail, the augmented Hamiltonian $H + D$ is plotted in Fig. 9. Once more, the results of both standard integrators show an unphysical behaviour which is characterised by oscillations and an increase respectively decrease of the augmented Hamiltonian. Solely, the 'ECMC-cG method' captures the conservation of the augmented Hamiltonian, respecting the energy balance (20). Furthermore, the standard integrators violate the conservation of angular momentum, whereas the Galerkin-based scheme features the mechanical consistency, as pictured in Fig. 10. It is important to emphasise that the abovementioned consistency properties of the 'ECMC-cG method' are guaranteed for small as well as for large time-step sizes,

[3] A program abort within the calculations is displayed by means of a vertical dashed line in the plots.

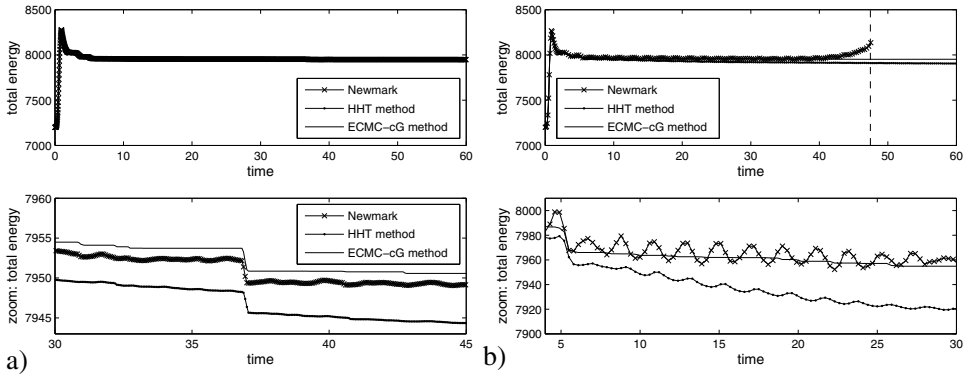


Figure 7: Total energy H: a) small time-step size, b) large time-step size

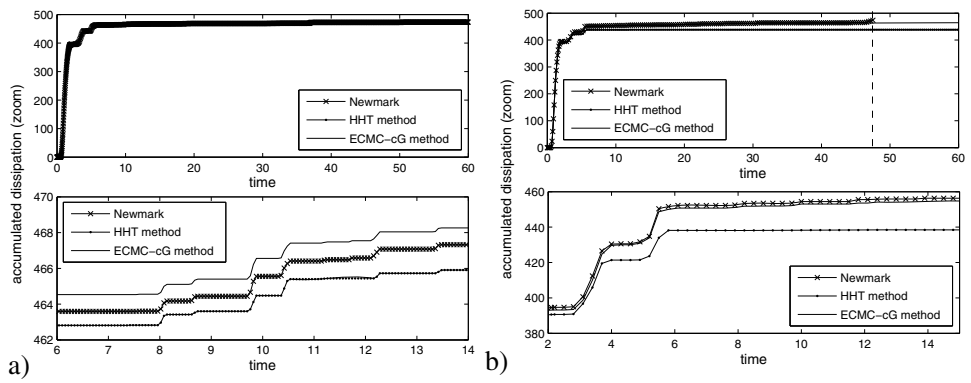


Figure 8: Accumulated dissipation D: a) small time-step size, b) large time-step size

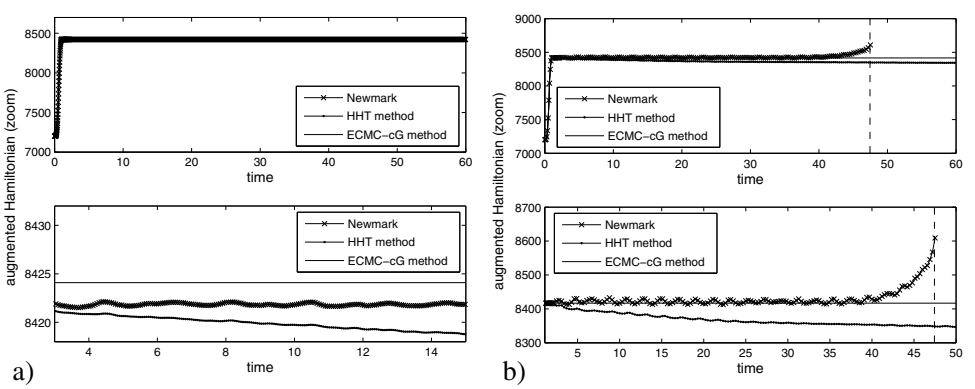


Figure 9: Augmented Hamiltonian $H + D$: a) small time-step size, b) large time-step size

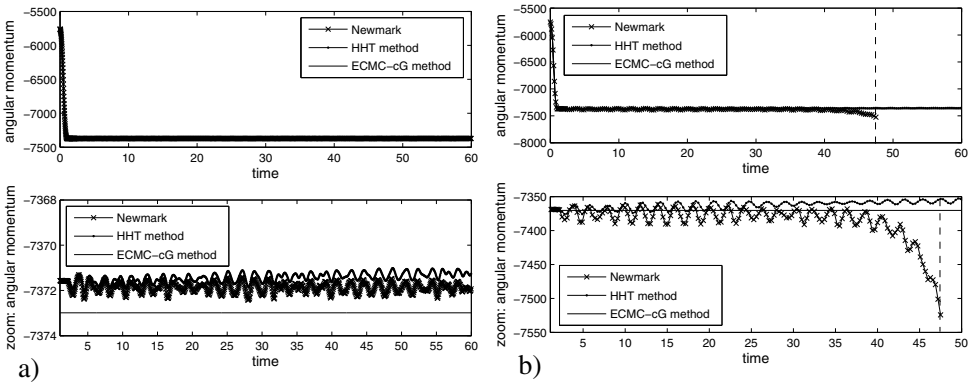


Figure 10: Component of the angular momentum $\boldsymbol{L}$: a) small time-step size, b) large time-step size

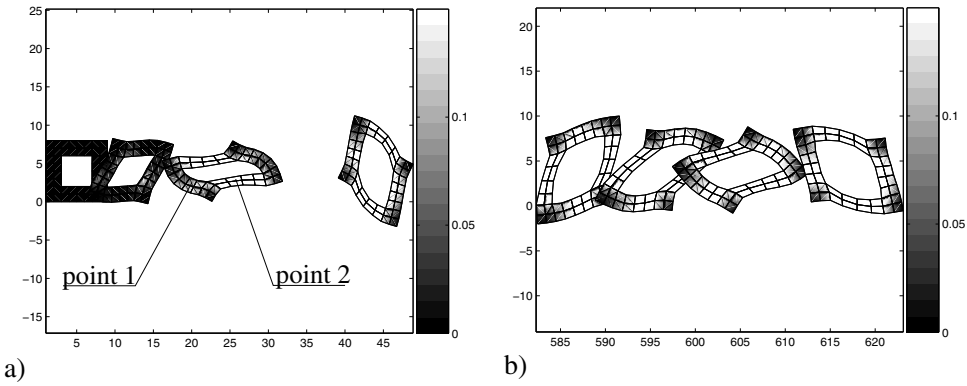


Figure 11: Contour plot of the hardening parameter (large time-step size): a) $t \in \{0.1, 0.8, 2, 4\}$, b) $t \in \{58.3, 59.2, 60.1, 61.3\}$

unaffected by changes of the step size during the calculation. Consequently, the proposed consistent Galerkin-based integration schemes enable an exceedingly robust integration of finite elasto-plasto-dynamics that is of particular importance for long-time simulations.

5 Conclusions

In the present paper, we have proposed time integration algorithms based on Finite Elements in time for nonlinear dynamics including plastic deformations, whereby the kinematic description of the applied plasticity model adopts a multiplicative decomposition of the deformation gradient into an elastic and a plastic part. With regard to a physically correct integration, nonstandard quadrature rules are required to feature thermodynamical

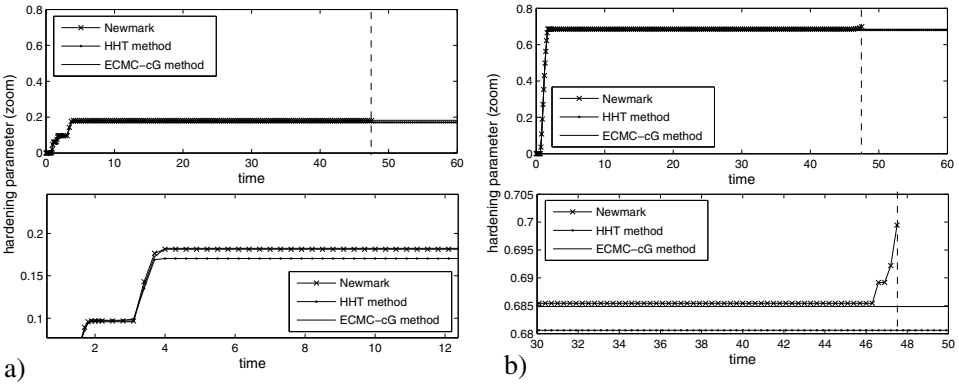


Figure 12: Evolution of the hardening parameter (large time-step size) in: a) point 1, b) point 2

consistency respectively energy-consistency. In this context, special emphasis has been placed on the assessment of the resulting schemes, especially, in comparison to classical integrators which are in particular well-established for linear (elastic) dynamical systems. Thereby, the superior performance of the proposed methods has been clearly confirmed, whereby particularly the 'ECMC-cG method' covers essential conservation properties of the continuum formulation for elastic as well as for plastic deformations. Recapitulating, Galerkin-based time-stepping schemes combined with adequate quadrature rules are also preeminently appropriate for geometrically nonlinear elasto-plasto-dynamics, providing an excellent numerical performance with pre-defined conservation properties.

Acknowledgements Financial support by the German Research Foundation DFG within the International Research Training Group 1131 'Visualization of Large and Unstructured Data Sets. Applications in Geospatial Planning, Modeling, and Engineering' is gratefully acknowledged by the authors.

References

[Arm05] F. Armero. Energy-dissipative momentum-conserving time-stepping algorithms for dynamic finite strain plasticity. In Proceedings of *COMPLAS VIII – International Conference on Computational Plasticity*, Barcelona, 2005.

[Arm06] F. Armero. Energy-dissipative momentum-conserving time-stepping algorithms for finite strain multiplicative plasticity. *Computer Methods in Applied Mechanics and Engineering*, 195:4862–4889, 2006.

[AZ06] F. Armero, C. Zambrana. Numerical integration of the nonlinear dynamics of elasto-plastic solids. In *Computational Mechanics – Solids, Structures and Coupled Problems*, 127–146, Springer, 2006.

[BB99] O. Bauchau, C. Bottasso. On the design of energy preserving and decaying schemes for flexible, non-linear multi-body systems. *Computer Methods in Applied Mechanics and Engineering*, 169:61–79, 1999.

[BS00a] P. Betsch, P. Steinmann. Inherently energy conserving time finite elements for classical mechanics. *Journal of Computational Physics*, 160:88–116, 2000.

[BS00b] P. Betsch, P. Steinmann. Conservation properties of a time FE method. Part I: Time-stepping schemes for N-body problems. *International Journal for Numerical Methods in Engineering*, 49:599–638, 2000.

[BS01] P. Betsch, P. Steinmann. Conservation properties of a time FE method. Part II: Time-stepping schemes for nonlinear elastodynamics. *International Journal for Numerical Methods in Engineering*, 50:1931–1955, 2001.

[Bot97] C. Bottasso. A new look at finite elements in time: a variational interpretation of Runge-Kutta methods. *Applied Numerical Mathematics*, 25:355–368, 1997.

[BBT01a] M. Borri, C. Bottasso, and L. Trainelli. Integration of elastic multibody systems by invariant conserving/dissipating algorithms. i. Formulation. *Computer Methods in Applied Mechanics and Engineering*, 190:3669–3699, 2001.

[BBT01b] C. Bottasso, M. Borri, and L. Trainelli. Integration of elastic multibody systems by invariant conserving/dissipating algorithms. ii. Numerical schemes and applications. *Computer Methods in Applied Mechanics and Engineering*, 190:3701–3733, 2001.

[EEHJ96] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. *Computational Differential Equations*. Cambridge University Press, 1996.

[Gonz00] O. Gonzalez. Exact energy and momentum conserving algorithms for general models in nonlinear elasticity. *Computer Methods in Applied Mechanics and Engineering*, 190:1763–1783, 2000.

[GBS05] M. Gross, P. Betsch, and P. Steinmann. Conservation properties of a time FE method. Part IV: Higher order energy and momentum conserving schemes. *International Journal for Numerical Methods in Engineering*, 63:1849–1897, 2005.

[HHT77] H. Hilber, T. Hughes, R. Taylor. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering & Structural Dynamics*, 5:283–292, 1977.

[Hug87] T. Hughes. *The Finite Element Method*. Prentice-Hall, New Jersey, 1987.

[KC99] D. Kuhl, M. Crisfield. Energy-conserving and decaying algorithms in non-linear structural mechanics. *International Journal for Numerical Methods in Engineering*, 45:569–599, 1999.

[Lee69] E. Lee. Elastic-plastic deformation at finite strains. *Journal of Applied Mechanics*, 36:1–6, 1969.

[ML02a] X. Meng, T. Laursen. Energy consistent algorithms for dynamic finite deformation plasticity. *Computer Methods in Applied Mechanics and Engineering*, 191:1639–1675, 2002.

[ML02b] X. Meng, T. Laursen. On energy consistency of large deformation plasticity models, with application to the design of unconditionally stable time integrators. *Finite Elements in Analysis and Design*, 38:949–963, 2002.

[MMS06a] R. Mohr, A. Menzel, and P. Steinmann. Galerkin-based time integrators for geometrically nonlinear elasto-plastodynamics – Challenges in modeling and visualization. In *Visualization of Large and Unstructured Data Sets, GI-Edition Lecture Notes in Informatics (LNI)*, S-4:185–194, 2006.

[MMS07a] R. Mohr, A. Menzel, and P. Steinmann. Conservation properties of Galerkin-based time integrators for geometrically nonlinear elasto-plasto-dynamics. In Proceedings of *IMPLAST – Symposium on Plasticity and Impact Mechanics*, Bochum, 2007.

[MMS07b] R. Mohr, A. Menzel, and P. Steinmann. Conservation properties of Galerkin-based time-stepping schemes for finite elasto-plasto-dynamics. In Proceedings of *COMPLAS IX – International Conference on Computational Plasticity*, Barcelona, 2007.

[MMS07c] R. Mohr, A. Menzel, and P. Steinmann. A consistent time-FE method for large strain elasto-plasto-dynamics. Submitted for publication, 2007.

[New59] N. Newmark. A method of computation for structural dynamics. *ASCE Journal of the Engineering Mechanics Division*, 85:67–94, 1959.

[NSP06] L. Noels, L. Stainier, and J. Ponthot. An energy momentum conserving algorithm using the variational formulation of visco-plastic updates. *International Journal for Numerical Methods in Engineering*, 65:904–942, 2006.

[ST92] J. Simo, N. Tarnow. The discrete energy-momentum method. Conserving algorithms for nonlinear elastodynamics. *Zeitschrift fuer Angewandte Mathematik und Physik (ZAMP)*, 43:757–792, 1992.

[Sim98] J. Simo. Numerical analysis and simulation of plasticity. In *Handbook of Numerical Analysis*, 6:183–499, 1998.

Towards completeness, a multiscale approach of confined particulate systems.

H.A. Meier[a], P. Steinmann[b] and E. Kuhl[c]

[a]Department of Mechanical Engineering,
University of Kaiserslautern, D-67653 Kaiserslautern, Germany.

[b]Department of Mechanical Engineering,
University of Erlangen-Nuremberg, D-91058 Erlangen, Germany.

[c]Department of Mechanical Engineering,
Stanford University, Stanford, CA 95305-4040, USA.

homei@rhrk.uni-kl.de, ps@rhrk.uni-kl.de, ekuhl@stanford.edu

Abstract: Numerical simulation and computational visualization of the failure characteristics of confined granular assemblies, e.g., sand, gravel or other types of loose aggregates, is the focal point of this publication. In general, standard continuum descriptions are exhausted if applied to loose granular materials, while discrete formulations fail to describe huge overall particulate structures. We propose a complete two scale homogenization procedure, including both a continuous and a discontinuous scale. Thus, we combine the capability of discrete methods to describe the behavior of the single grains and the possibility of a continuum approach to discretize the overall structure. Connections between the two scales are based on the concept of introducing a representative volume element on the discrete microscale. Driven by quantities from the macroscale, the representative volume element acts like a material law, returning the needed quantities to the continuous macro level. The particular challenge of this work lies in defining the connection between these two scales in terms of physical quantities and equations on the one hand and in terms of introducing appropriate visual tools which ultimately yield an improved understanding of these complex coupling mechanism on the other hand.

1 Introduction

Simulation of the distinguishing and complicated behavior of granular media calls for the mobilization of multiscale simulation techniques. While standard continuum methods strand by reproducing distinguishing behaviors, e.g., the breaking and forming of contacts between the grains, discrete methods are limited due to their computational costs. A multiscale combination of both methods leads to a powerful and efficient simulation tool. Thereby, a finite element method (fem) discretizes the overall structure on the ma-

croscale level. A discrete element method (dem) is used on the Gauss point level to simulate the behavior of the single grains. Connection between the two scales is based on the concept of introducing a representative volume element (rve) on the microscale level. This rve contains the discrete granular structure and acts as a material law for the macroscale level. The dem, introduced by [CS78, CS79], is suitable to capture the behavior of granular aggregates. In such a multiscale combination, the number of particles (nop) is limited inside the rve and the drawback of the dem disappears naturally. Studies pertaining the macroscopic stress and strain formulation are found in the publication of [KR96, KDHR00, DKHR01, EDM01, ERDD03].
This publication focuses on the introduction of a consistent tangent operator, allowing the use of an implicit solution scheme on the macroscale level. We restrict this contribution to the assumption of Taylor and Voigt, see [Voi89]. Thus, we do not consider any kind of micro fluctuations. All particles are mapped by the macroscopic deformation gradient tensor, leading to a homogeneous microscale deformation. The Piola stress as well as the consistent tangent operator are derived from the overall macroscopic energy, ab initio guaranteeing a major symmetry of the tangent operator.
The publication is segmented as follows: The calculation of the inter particle contact forces is presented in Section 2. Therefore, a force potential function is introduced which depends on the particle overlap. Section 3 concerns the homogenization process, based on the averaged macroscopic energy, resulting in the definition of the averaged macroscopic stress and tangent operator. To ensure the representativeness of the used volume element, a basic deformation rve study is outlined in Section 4. Uniformity of the initial contact network is compared by a contact normal density function and by characteristic error bar analyzes. A final example based on the well-known slope stability benchmark problem is illustrated in Section 5. Lastly, Section 6 closes with a final discussion.

2 Microscale - contact force

On the microscale level, the spatial position of each particle, $\boldsymbol{x}_i$ is described by a linear mapping of the initial particle position $\boldsymbol{X}_i$ by the macroscopic deformation gradient $\overline{\boldsymbol{F}}$.

$$\boldsymbol{x}_i = \overline{\boldsymbol{F}} \bullet \boldsymbol{X}_i \tag{1}$$

Individual particle fluctuations are ignored. This mapping, see (1), illustrates the assumption of Taylor and Voigt [Voi89], i.e., restricting the microscopic deformation to be homogeneous over the entire granular assembly. A comparable assumption was stated by Cauchy and Born in the context of continuum atomistics, see [Cau28a, Cau28b, Bor15]. The branch vector in the deformed configuration, $\boldsymbol{l}_{ij}$, connecting the center of particle i with the center of particle j, is received by subtracting the position vector of particle i from the position vector of particle j, see Fig. 1.

$$\boldsymbol{l}_{ij} = \boldsymbol{x}_j - \boldsymbol{x}_i = \overline{\boldsymbol{F}} \bullet \left[\boldsymbol{X}_j - \boldsymbol{X}_i\right] = \overline{\boldsymbol{F}} \bullet \boldsymbol{L}_{ij} \tag{2}$$

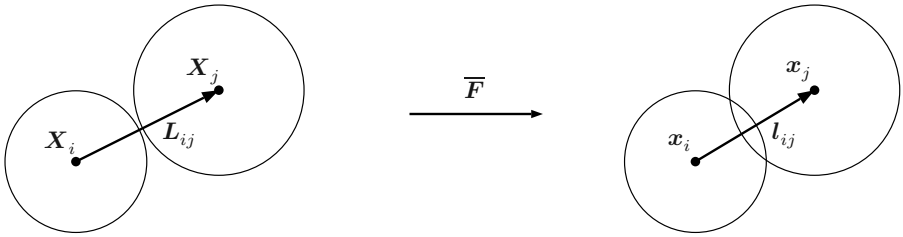


Figure 1: Initial and deformed configuration of the particles i and j. Left: Initial configuration with ε_{ij} less than zero. The branch vector $\boldsymbol{L}_{ij}$ connects the centers of the particles i and j. Right: Current configuration with an overlap ε_{ij} greater than zero. The particles i and j are in contact. The branch vector $\boldsymbol{l}_{ij}$ relates the two particle centers.

By subtracting the length of the branch vector $\left|\left|\boldsymbol{l}_{ij}\right|\right|$ from the sum of the particle radii, denoted by r_i and r_j, we obtain a description of the particle overlap.

$$\varepsilon_{ij} = r_i + r_j - \left|\left|\boldsymbol{l}_{ij}\right|\right| \tag{3}$$

Inspecting the definition of the particle overlap, we find ε_{ij} to have a positive or zero value in the case of contact. In this state, we allow the transfer of contact forces between the contacting grains. Accordingly, a negative value of ε_{ij} signals a gap of the size $|\varepsilon_{ij}|$. Clearly, (3) does not only represent the particle overlap, but also serves as an implicit contact check. Next, we introduce a force-potential function Φ_{ij} which depends on the overlap ε_{ij}, see, e.g., [MD04, Zoh05]. We require the force-potential function to be convex and its derivative monotonously increasing. Additionally, we demand its derivative to be zero for ε_{ij} being zero itself. Thus, we implicitly enforce the constraint $\varepsilon_{ij} < 0$.

$$\Phi_{ij} = \frac{E_{\mathsf{n}ij}}{2} \left[\mathscr{H}\left(\varepsilon_{ij}\right) \varepsilon_{ij}\right]^2 \tag{4}$$

The normal contact stiffness between the particles i and j is denoted by $E_{\mathsf{n}ij}$, whereas $\mathscr{H}$ expresses the Heaviside function. By letting the Heaviside function act on the particle overlap, the previous postulation and requirements are included in (4). Thus, our force-potential function describes a non-harmonic potential, solely delivering a value different from zero in the case of contact. Performing the derivative of (4) with respect to the position of particle i leads to the normal contact force $\boldsymbol{f}_{ij}$, acting on particle i.

$$\boldsymbol{f}_{ij} = -\frac{\mathrm{d}\Phi_{ij}}{\mathrm{d}\boldsymbol{x}_i} = -E_{\mathsf{n}ij}\mathscr{H}\left(\varepsilon_{ij}\right) \varepsilon_{ij} \boldsymbol{n}_{ij} \tag{5}$$

Therein, the magnitude of the contact force is given by $-E_{\mathsf{n}ij}\mathscr{H}\left(\varepsilon_{ij}\right) \varepsilon_{ij}$. The contact normal, pointing from the center of particle i to the center of particle j, is denoted by $\boldsymbol{n}_{ij} = \boldsymbol{l}_{ij} / \left|\left|\boldsymbol{l}_{ij}\right|\right|$. Using Newtons third law of reciprocal forces, the contact force acting on particle j equals $\boldsymbol{f}_{ji} = -\boldsymbol{f}_{ij}$. The complete particle assembly can be compared to

a network of linear springs, see [Hre41]. In contrast to the work of Hrennikoff, however, the contact network between the particles is developed and altered continuously during the simulation.

3 Macroscale - stress and tangent moduli

Our homogenization procedure is based on the well-known energy averaging theorem by Hill ([Hil72]), stating the equivalence between the volume average of the microscopic energy inside the rve and the macroscopic energy.

$$\overline{\Phi}\left(\overline{\boldsymbol{F}}\right) = \left\langle \Phi\left(\varepsilon_{ij}\left(\overline{\boldsymbol{F}}\right)\right)\right\rangle = \frac{1}{2\mathsf{V}_{\mathsf{rve}}} \sum_{i=1}^{\mathsf{nop}} \sum_{\substack{j=1 \\ j\neq i}}^{\mathsf{nop}} \Phi_{ij}\left(\varepsilon_{ij}\right), \tag{6}$$

where $\mathsf{V}_{\mathsf{rve}}$ denotes the volume of the rve in the undeformed reference configuration and

$$\begin{aligned} \left\langle \Phi\left(\overline{\boldsymbol{F}}\right)\right\rangle &= \frac{1}{4\mathsf{V}_{\mathsf{rve}}} \sum_{i=1}^{\mathsf{nop}} \sum_{\substack{j=1 \\ j\neq i}}^{\mathsf{nop}} E_{\mathsf{n}ij} \left[\mathscr{H}\left(\varepsilon_{ij}\right) \varepsilon_{ij}\right]^2 \\ &= \frac{1}{4\mathsf{V}_{\mathsf{rve}}} \sum_{i=1}^{\mathsf{nop}} \sum_{\substack{j=1 \\ j\neq i}}^{\mathsf{nop}} E_{\mathsf{n}ij} \mathscr{H}\left(\varepsilon_{ij}\right) \left[r_i + r_j - \left|\left|\overline{\boldsymbol{F}} \bullet \boldsymbol{L}_{ij}\right|\right|\right]^2. \end{aligned} \tag{7}$$

Based on the selected summation limits, each particle contact is considered twice. A multiplication by a factor of one half corrects the averaged energy output. At this point it is obvious that the volume averaged microscopic energy solely relies on the macroscopic deformation gradient. Insertion of (3) and (4) into (6) leads to the result presented in (7). The macroscopic Piola stress $\overline{\boldsymbol{P}}$ is obtained by taking the derivative of the averaged microscopic energy with respect to the macroscopic deformation gradient tensor.

$$\overline{\boldsymbol{P}}\left(\overline{\boldsymbol{F}}\right) = \frac{\mathrm{d}\left\langle \Phi\left(\overline{\boldsymbol{F}}\right)\right\rangle}{\mathrm{d}\overline{\boldsymbol{F}}} \tag{8}$$

with

$$\begin{aligned} \overline{\boldsymbol{P}}\left(\overline{\boldsymbol{F}}\right) &= \frac{1}{4\mathsf{V}_{\mathsf{rve}}} \sum_{i=1}^{\mathsf{nop}} \sum_{\substack{j=1 \\ j\neq i}}^{\mathsf{nop}} E_{\mathsf{n}ij} \frac{d\mathscr{H}\left(\varepsilon_{ij}\right) \left[r_i + r_j - \left|\left|\overline{\boldsymbol{F}} \bullet \boldsymbol{L}_{ij}\right|\right|\right]^2}{d\overline{\boldsymbol{F}}} \\ &= -\frac{1}{2\mathsf{V}_{\mathsf{rve}}} \sum_{i=1}^{\mathsf{nop}} \sum_{\substack{j=1 \\ j\neq i}}^{\mathsf{nop}} E_{\mathsf{n}ij} \mathscr{H}\left(\varepsilon_{ij}\right) \varepsilon_{ij} \boldsymbol{n}_{ij} \otimes \boldsymbol{L}_{ij}. \end{aligned} \tag{9}$$

In this, $\boldsymbol{n}_{ij} = \overline{\boldsymbol{F}} \bullet \boldsymbol{L}_{ij} / \left|\left| \overline{\boldsymbol{F}} \bullet \boldsymbol{L}_{ij} \right|\right|$ represents the unit contact normal vector in the current configuration. Using (5) we can rewrite (9) in a compact form,

$$\overline{\boldsymbol{P}}\left(\overline{\boldsymbol{F}}\right) = \frac{1}{2\mathsf{V}_{\mathsf{rve}}} \sum_{i=1}^{\mathsf{nop}} \sum_{\substack{j=1 \\ j \neq i}}^{\mathsf{nop}} \boldsymbol{f}_{ij} \otimes \boldsymbol{L}_{ij}, \tag{10}$$

solely consisting of the normal contact forces of the current configuration as well as the branch vector of the initial configuration. Taking the second derivative of the volume averaged microscopic energy with respect to the macroscopic deformation gradient, one obtains the fourth order algorithmic tangent operator $\overline{\mathbb{A}}$. As in classical structural mechanics, a split into a purely geometric and material part is possible. We can observe that non-linearity due to changes inside the contact network are directly reflected onto the material part of the tangent operator.

$$\overline{\mathbb{A}}\left(\overline{\boldsymbol{F}}\right) = \frac{\mathrm{d}^2 \left\langle \Phi\left(\overline{\boldsymbol{F}}\right)\right\rangle}{\mathrm{d}\overline{\boldsymbol{F}} \otimes \mathrm{d}\overline{\boldsymbol{F}}} = \frac{\mathrm{d}\overline{\boldsymbol{P}}\left(\overline{\boldsymbol{F}}\right)}{\mathrm{d}\overline{\boldsymbol{F}}} = \overline{\mathbb{A}}^{\mathsf{geo}}\left(\overline{\boldsymbol{F}}\right) + \overline{\mathbb{A}}^{\mathsf{mat}}\left(\overline{\boldsymbol{F}}\right) \tag{11}$$

with

$$\begin{aligned}
\overline{\mathbb{A}}^{\mathsf{geo}}\left(\overline{\boldsymbol{F}}\right) &= -\frac{1}{2\mathsf{V}_{\mathsf{rve}}} \sum_{i=1}^{\mathsf{nop}} \sum_{\substack{j=1 \\ j \neq i}}^{\mathsf{nop}} \frac{\mathscr{H}\left(\varepsilon_{ij}\right) E_{\mathsf{n}ij}}{\left|\left|\boldsymbol{l}_{ij}\right|\right|} \varepsilon_{ij} \mathbf{1} \overline{\otimes} \left[\boldsymbol{L}_{ij} \otimes \boldsymbol{L}_{ij}\right] \\
\overline{\mathbb{A}}^{\mathsf{mat}}\left(\overline{\boldsymbol{F}}\right) &= \frac{1}{2\mathsf{V}_{\mathsf{rve}}} \sum_{i=1}^{\mathsf{nop}} \sum_{\substack{j=1 \\ j \neq i}}^{\mathsf{nop}} \frac{\mathscr{H}\left(\varepsilon_{ij}\right) E_{\mathsf{n}ij}}{\left|\left|\boldsymbol{l}_{ij}\right|\right|} \left[r_i + r_j\right] \left[\boldsymbol{n}_{ij} \otimes \boldsymbol{n}_{ij}\right] \overline{\otimes} \left[\boldsymbol{L}_{ij} \otimes \boldsymbol{L}_{ij}\right]
\end{aligned} \tag{12}$$

Please note the special dyadic product, $\{\bullet \overline{\otimes} \circ\}_{abcd} = \{\bullet\}_{ac} \otimes \{\circ\}_{bd}$, enforcing the major symmetry of $\overline{\mathbb{A}}$. As with the macroscopic Piola stress, the algorithmic tangent operator solely depends on the macroscopic deformation gradient tensor.

4 Microscale - discrete element method

To select the appropriate rve, we call on a contact normal density function. This density function is used as a measure of uniformity and is outlined in [MSW$^+$07]. Yet, the contact density function promises a deeper insight with respect to the uniformity of the contact network. The rves, generated by the algorithm presented in [MKS07], are based on the grain size distribution shown in Fig. 2. Intrinsically, all generated rves include a geometric periodic boundary, a scaled grains size distribution and an unstructured particle network. We select a grain size distribution for quartz sand as shown in Fig. 2 and generate five rves for a number of 70, 350 and 700 primary particles, depicted in Fig. 3. Corresponding contact normal density functions are located beneath each rve, where we selected angles

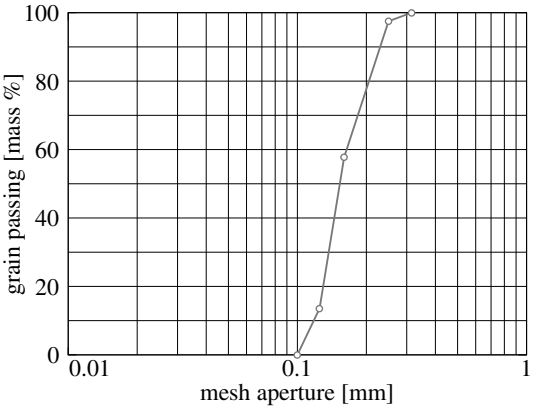


Figure 2: Typical grain size distribution for quartz sand used for rve generation. The grain passing in mass percent over mesh aperture is depicted. The abscissa shows the mesh aperture in [mm], whereas the ordinate reads the grain passing in volume percent.

of influence equal to 10°, 20° and 30°. One can observe that a wider angle of influence smooths the density function, while a smaller angle of influence results into a rather ruff output. Additionally, we find that a larger nop leads to a more uniform contact density function. The first two sets of rves, including each 70 and 350 primary particles, show a strong variation between the produced contact density functions. Instead, the rves including 700 primary particles show a good uniformity. A superior agreement of the contact density functions is found for the third rve of set three. Thus, this rve is considered to behave in an isotropic manner for small deformations. In contrast, the third rve of set one will show an anisotropic behavior. Next, our interest is focused on applying uniaxial compression as well as simple shear on each rve. The produced error bar plots, showing entries of the Cauchy stress versus the corresponding components of the deformation gradient, are depicted in Fig. 4 and 5. While the compression test shows a good overall agreement for all rves, the results of the simple shear test differ significantly. In case of the compression test, a linear stress behavior in the loading direction is observed, while in the orthogonal direction a nonlinear behavior is noticed. It is remarkable that for different nop the stress in the direction orthogonal to the direction of compression seems to converge towards a fixed value. Similar convergence is seen in the plot depicted in Fig. 5. In the case of the simple shear deformation a size reduction of the error bars, correlating to the increase of nop, is noticed. Thus, we can conclude that for an unlimited nop the error bars will tend towards a negligible value, e.g., an almost uniform behavior can be expected. Please note, the coarse scale smoothness of the depicted stress curves is strongly related to the Taylor assumption. Therefore, if fluctuations on the microscale are considered, a non-smooth coarse scale behavior of the stress is observed, see [MD04, Det06]. Nevertheless, fine scale non-smoothness is observed in both approaches. This observation is linked to minor changes inside the contact network.

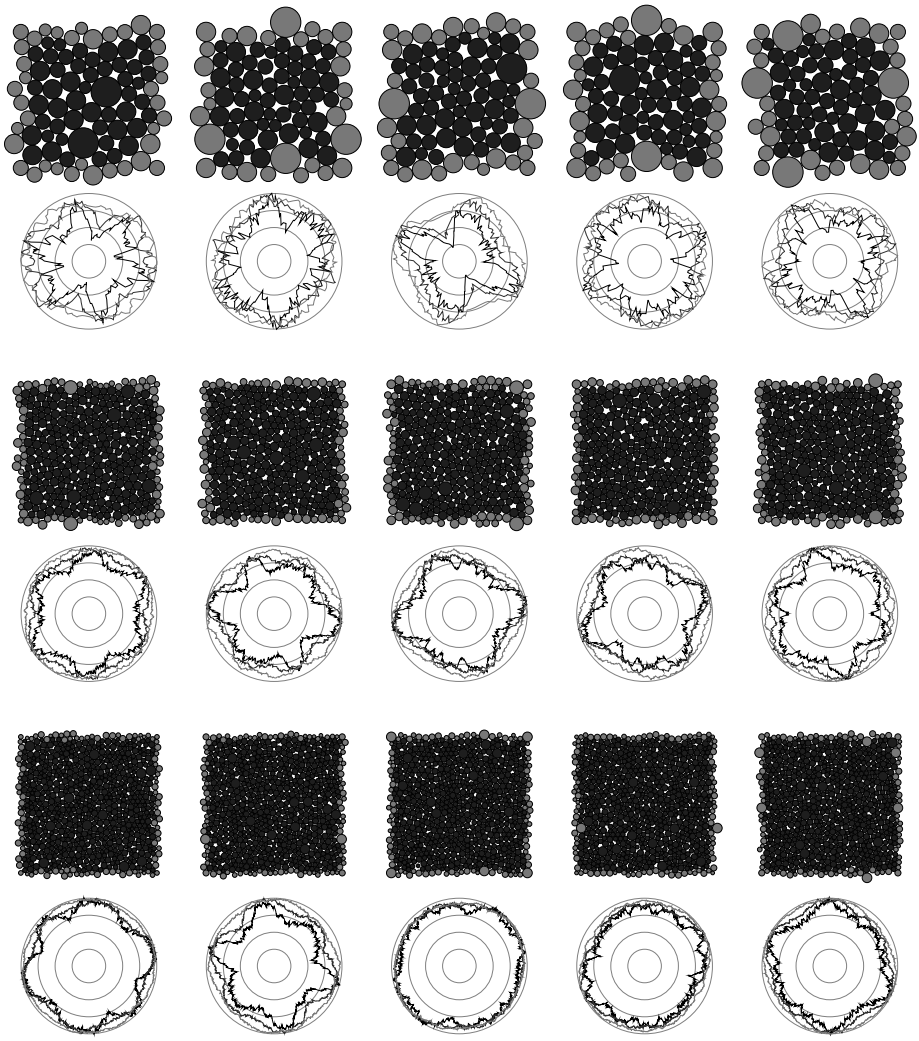


Figure 3: rves and their corresponding contact normal density functions. Black colored functions relate to an angle of influence equal to 10°, while blue and red colored functions correspond to an angle of influence equal to 20° and 30°, respectively. First set: Five rves, containing each 70 primary particles. Second set: Five rves, containing 350 primary particles. Third set: Five rves containing 700 primary particles (reproduced in color on p. 194).

5 Macroscale - finite element method

In the following we will consider a slope stability problem, first discussed in [ZP77]. The slope is subjected to dead load as well as a load originating from an massless strip footing

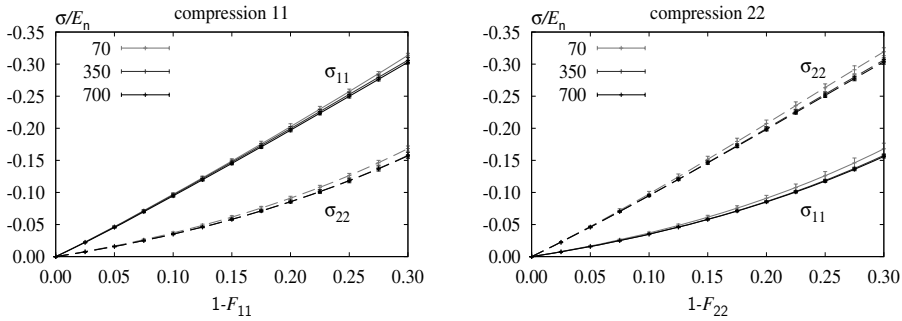


Figure 4: Averaged compression stress curves of the particle assemblies shown in Fig. 3. Error bars show the standard deviation. Left: Compression in 11 direction. Right: Compression in 22 direction. Solid lines relate to the Cauchy stress in 11 direction, while dashed lines are associated with the Cauchy stress in 22 direction.

subjected to an eccentric force. The dimensions of the problem as well as the applied boundary conditions are depicted in Fig. 6. Physical parameters are listed in Table 1. The footing has a width of $23.25\ m$. The dead load of the slope is applied in the first load step.

Table 1: Physical parameters for slope example

mass density	2.5E+03	$[kg/m^3]$
normal contact stiffness	2.8E+07	$[N/m^2]$
load	-4.0E+07	$[N]$

Twenty load steps are used to apply the eccentric force which is subjected to the massless strip footing. Fig. 7 shows a contour plot of the macroscopic von Mises stress. Four rves, connected to Gauss points of interest, are depicted, showing the force chain network in the deformed configuration. The convergence behavior of the relative energy norm for step two, six, ten, fifteen and eighteen is listed in Table 2.

6 Discussion

Inspired by the challenge of defining a complete multiscale calculation cycle, we presented a possibility to derive the macroscopic stress and tangent operator in the context of granular media. Both, the macroscopic stress and the macroscopic tangent operator are derived from the macroscopic strain energy, while the macroscopic strain energy itself depends on the prescribed macroscopic deformation gradient tensor. A complete two scale formulation was outlined, including different methods on the two scales, i.e., we do not apply the common FE^2. An example, showing the successful derivation and implementation was

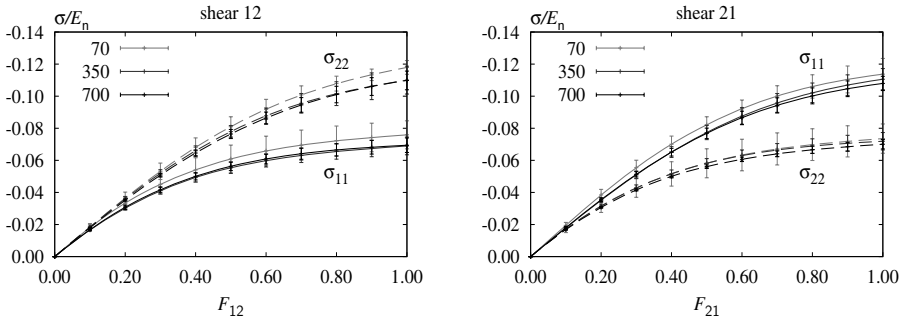


Figure 5: Averaged simple shear stress curves of the particle assemblies shown in Fig. 3. Error bars show the standard deviation. Left: Simple shear in 12 direction. Right: Simple shear in 21 direction. Solid lines relate to the Cauchy stress in 11 direction, while dashed lines are associated with the Cauchy stress in 22 direction.

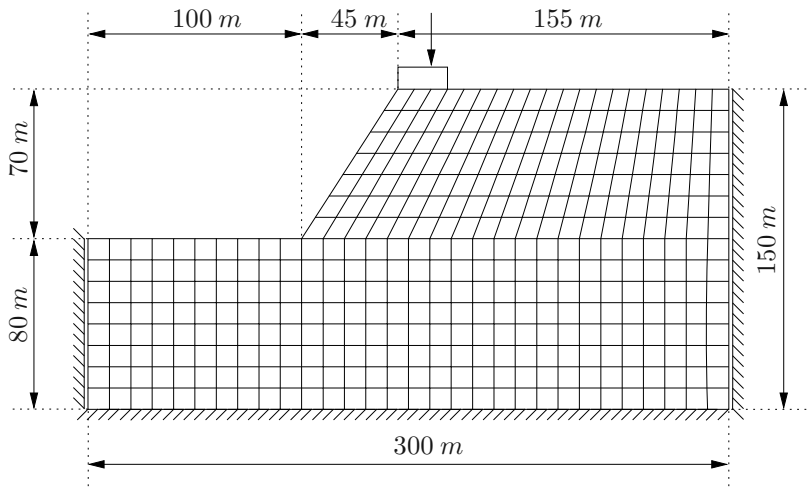


Figure 6: Undeformed mesh of the slope stability problem. Boundary conditions are depicted. 380 Q1 elements, containing each four Gauss points, are used. The massless footing is subjected to an eccentric force. The eccentricity is equal to $3.875\ m$.

granted, using a rve. Prior to the calculation, the representativeness was proofed by the use of a contact normal density function, showing the uniformity of the contact normals of the volume element. Stress curves, resulting from basic rve deformation, reinforce the prediction of the contact normal density functions. In case of the analysis of confined granular media, the results presented show an improvement with regards to the computational cost and the completeness of the algorithm. Thus, large scale computations of confined granular material are possible in the context of multiscale calculations. Finally, we would

Table 2: Convergence behavior of the relative energy norm

iter	step 2	step 6	step 10	step 15	step 18
1	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
2	1.318E-02	3.341E-04	3.940E-04	1.322E-03	4.817E-03
3	1.947E-04	2.308E-08	1.503E-08	1.999E-07	3.816E-06
4	1.492E-07	7.110E-16	7.415E-16	6.114E-14	7.276E-12
5	5.846E-13	3.441E-26	5.014E-26	4.023E-26	5.932E-24
6	3.057E-26				

like to point out that although the presented algorithm has been proven highly effective and algorithmically efficient, it is far from being general enough to capture all characteristic effects in granular media.

Acknowledgment

The authors thank the German Research Foundation (DFG) for 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. Furthermore, we kindly acknowledge the productive and inspiring collaboration with H. Hagen, A. Kerren, M. Schlemmer as well as C. Wagner from the AG Graphische Datenverarbeitung, Kaiserslautern, Germany.

References

[Bor15] M. Born. *Dynamik der Kristallgitter*. Teubner, 1915.

[Cau28a] A. L. Cauchy. De la pression ou tension dans un système de points matériels. *Exercises De Mathématiques*, pages 253–277, 1828.

[Cau28b] A. L. Cauchy. Sur l'équilibre et le mouvement d'un système de points matériels sollicités par des forces d'attraction ou de répulsion mutuelle. *Exercises De Mathématiques*, pages 227–252, 1828.

[CS78] P. A. Cundall and O. D. L. Strack. The Distinct Element Method as a Tool For Research in Granular Media. Technical report, Report to the National Science Foundation Concerning NSF Grant ENG76-20711, PART I, 1978.

[CS79] P. A. Cundall and O. D. L. Strack. The Distinct Element Method as a Tool for Research in Granular Media. Technical report, Report to the National Science Foundation Concerning NSF Grant ENG76-20711, PART II, 1979.

[Det06] J. P. Dettmar. *Static and Dynamic Homogenization Analyses of Discrete Granular and Atomistic Systems on Different Time and Length Scales*. PhD thesis, Institut für Mechanik (Bauswesen) , Universität Stuttgart, 2006.

[DKHR01] G. A. D'Addetta, F. Kun, H. J. Hermann, and E. Ramm. *From solids to granulates - discrete element simulations of fracture and fragmentation processes in geomaterials*, pages 231–258. Continuous and Discontinuous Modelling of Cohesive Frictional Materials, Lecture Notes in Physics 586. Springer-Verlag, Berlin, Germany, 2001.

[EDM01] W. Ehlers, S. Diebels, and T. Michelitsch. *Microscopic modelling of granular materials taking into account particle rotations*, pages 259–274. Continuous and Discontinuous Modelling of Cohesive Frictional Materials, Lecture Notes in Physics 586. Springer-Verlag, Berlin, Germany, 2001.

[ERDD03] W. Ehlers, E. Ramm, S. Diebels, and G. A. D'Addetta. From particle ensembles to Cosserat continua: Homogenization of contact forces towards stresses and couple stresses. *Int. J. Solids and Structures*, 40:6681–6702, 2003.

[Hil72] R. Hill. On constitutive macro-variables for heterogeneous solids at finite strain. *Proc. R. Soc. Lond. A*, 326:131–147, 1972.

[Hre41] A. Hrennikoff. Solution of Problems of Elasticity by the Framework Method. *ASME J. Appl. Mech.*, 8:169–175, 1941.

[KDHR00] E. Kuhl, G. A. D'Addetta, H. J. Herrmann, and E. Ramm. A comparison of discrete granular material models with continuous microplane formulations. *Granular Matter*, 2:123–135, 2000.

[KR96] N. P. Kruyt and L. Rothenburg. Micromechanical Definition of the Strain Tensor for Granular Materials. *ASME Journal of Applied Mechanics*, 118:706–711, 1996.

[MD04] C. Miehe and J. Dettmar. A framework for micro - macro transitions in periodic particle aggregates of granular materials. *Comput. Methods Appl. Mech. Engrg.*, 193:225–256, 2004.

[MKS07] H. A. Meier, E. Kuhl, and P. Steinmann. A note on the generation of periodic granular microstructures based on grain size distributions. *Int. J. Numer. Anal. Meth. Geomech.,(in press)*, 2007.

[MSW+07] H. A. Meier, M. Schlemmer, C. Wagner, A. Kerren, H. Hagen, E. Kuhl, and P. Steinmann. Visualization of Particle Interactions in Granular Media. *Submitted for publication*, 2007.

[Voi89] W. Voigt. Über die Beziehungen zwischen den beiden Elastizitätskonstanten isotroper Körper. *Wied. Ann.*, 38:573–587, 1889.

[Zoh05] T. I. Zohdi. Charge-induced clustering in multifield particulate flows. *Int. J. Numer. Meth. Engn.*, 62:870–898, 2005.

[ZP77] O. C. Zienkiewicz and G. N. Pande. Time-dependent multilaminate model of rocks - a numerical study of deformation and failure rock masses. *Int. J. Anal. Meth. Geomech.*, 1:219–247, 1977.

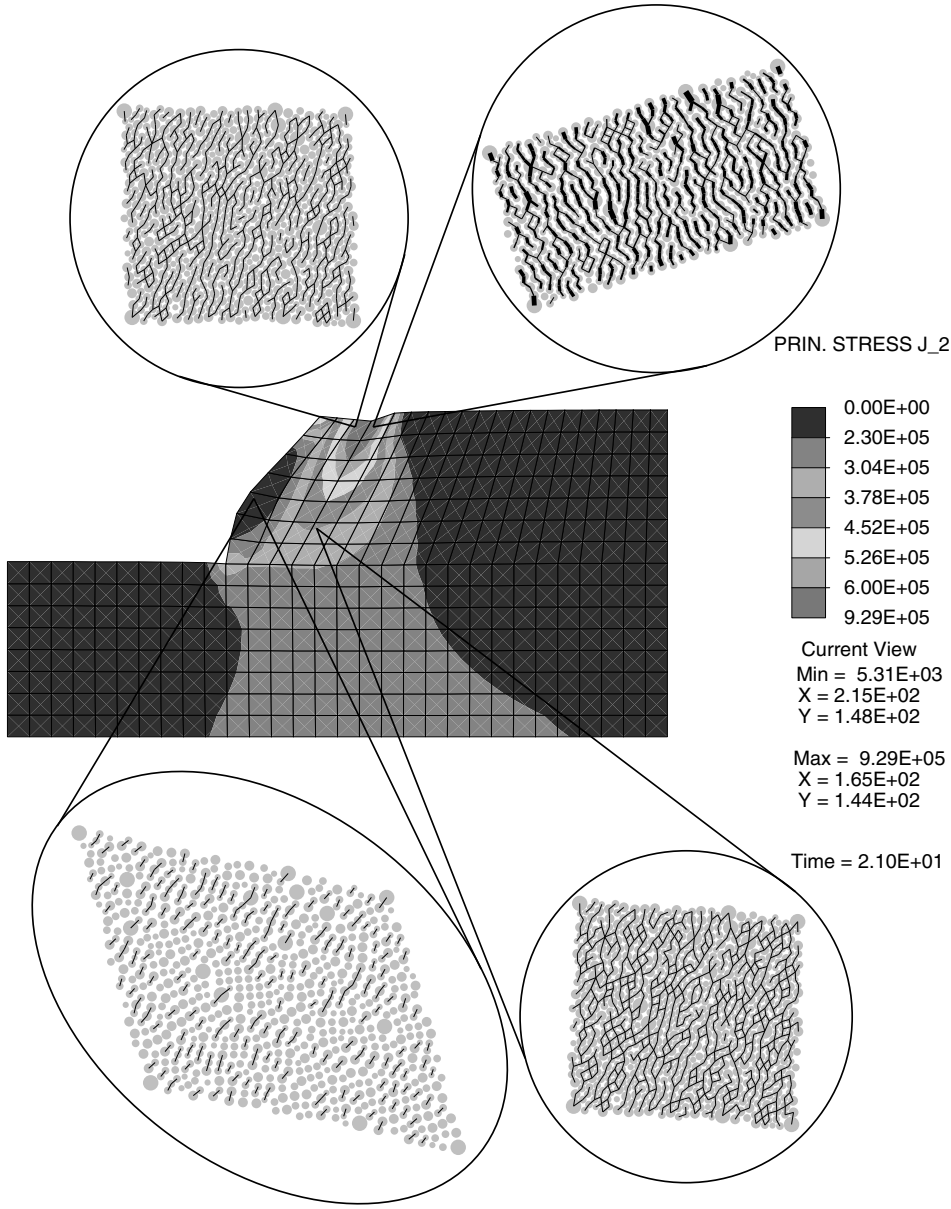


Figure 7: Final, deformed slope. The macroscopic von Mises stress is plotted. Initial localization is visible. Four deformed rves containing their contact networks are depicted. The tickness of the branches corresponds to the magnitude of the contact forces. The scaling factors for the branch thickness as well as for the deformation are equal for all four particle plots (reproduced in color on p. 195).

Geometric Numerical Integration of Simple Dynamical Systems

Patrick R. Schmitt and Paul Steinmann
University of Kaiserslautern
Chair of Applied Mechanics
Department of Mechanical and Process Engineering
D-67653 Kaiserslautern, Germany
prschmit@rhrk.uni-kl.de

Abstract: Understanding the behavior of a dynamical system is usually accomplished by visualization of its phase space portraits. Finite element simulations of dynamical systems yield a very high dimensionality of phase space, i.e. twice the number of nodal degrees of freedom. Therefore insight into phase space structure can only be gained by reduction of the model's dimensionality. The phase space of Hamiltonian systems is of particular interest because of its inherent geometric features namely being the co-tangent bundle of the configuration space of the problem and therefore having a natural symplectic structure. In this contribution a class of geometry preserving integrators based on Lie-groups and -algebras is presented which preserve these geometric features exactly. Examples of calculations for a simple dynamical system are detailed.

1 Introduction

In order to study the behaviour of dynamical systems one usually makes use of the theory of Hamiltonian systems and the phase-space concept. Within Hamilton's framework a dynamical system with n degrees of freedom (DOFs) is described by generalized coordinates $q_k \in \mathbb{R}$ and corresponding canonical momenta $p_k \in \mathbb{R}$ for $k = 1, \dots, n$. The scalar Hamiltonian $H : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$ is sufficient to describe the dynamics of the system completely. The equations of motion are Hamilton's equations:

$$\dot{p}_k = -\frac{\partial H(q,p,t)}{\partial q_k}, \qquad \dot{q}_k = \frac{\partial H(q,p,t)}{\partial p_k}, \quad k = 1, \dots, n$$

Solutions to this system of $2n$ ordinary differential equations are curves $(q(t), p(t)) \in \mathbb{R}^{2n}$ in *phase-space*.

To illustrate the phase-space concept figure 1 depicts one of the most simple dynamical systems, i.e. the 2-dimensional mathematical pendulum. Solution curves are clearly divided by a separatrix into two regions of vastly different behaviour. In the neighorhood of the origin of the coordinate system the pendulum is in normal back-and-forth oscillatory

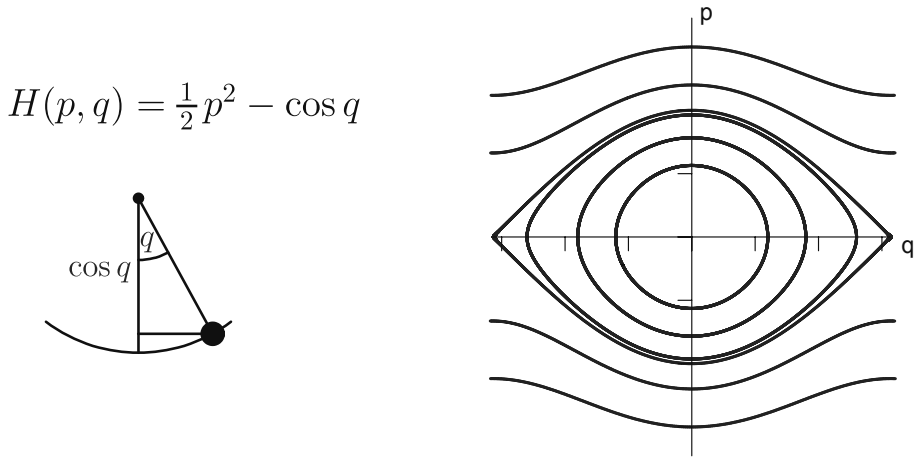


Figure 1: 2-dimensional mathematical pendulum in "natural" units, gravity acting downwards (left) and its phase-space portrait (right)

motion (closed, ellipsoidal solution curves). The outer region describes clockwise and counterclockwise spinning motion.

The dynamics of engineering structures are usually modelled by finite element simulations which leads to high dimensionality of phase spaces, i.e. twice the number of nodal degrees of freedom. Even small academic examples like the rotor blade shown in the upper left quadrant of figure 2 have a huge number of DOFs, in this case 716. The time-evolution of this freely flying rotor blade was determined by solving Hamilton's equations with an enhanced Galerkin method [Gro04]. The three quadrants labeled "phase-space projection" show different views of the $\{q_1, p_1, p_2\}$-projection for an arbitrarily chosen node. Obviously projection to an only 3-dimensional subspace of a vastly higher dimensional phase-space can not yield insight into the overall phase space structure.

Inspired by ideas from [MR94, SK03] we therefore turn to a geometrized version of Hamiltonian dynamics. The (finite dimensional) configuration space of a physical system has the structure of a (smooth) manifold and is denoted by Q. The momentum phase-space of this system is just the cotangent bundle T^*Q which is equipped with a natural symplectic structure ω (a 2-form in the language of differential geometry). The Hamiltonian is a smooth map $H : T^*Q \to \mathbb{R}$ and the corresponding Hamiltonian vector field is denoted X_H and determined by $dH = \omega(X_H, .)$. Hamilton's equations are just the equations of the flow of the vector-field X_H.

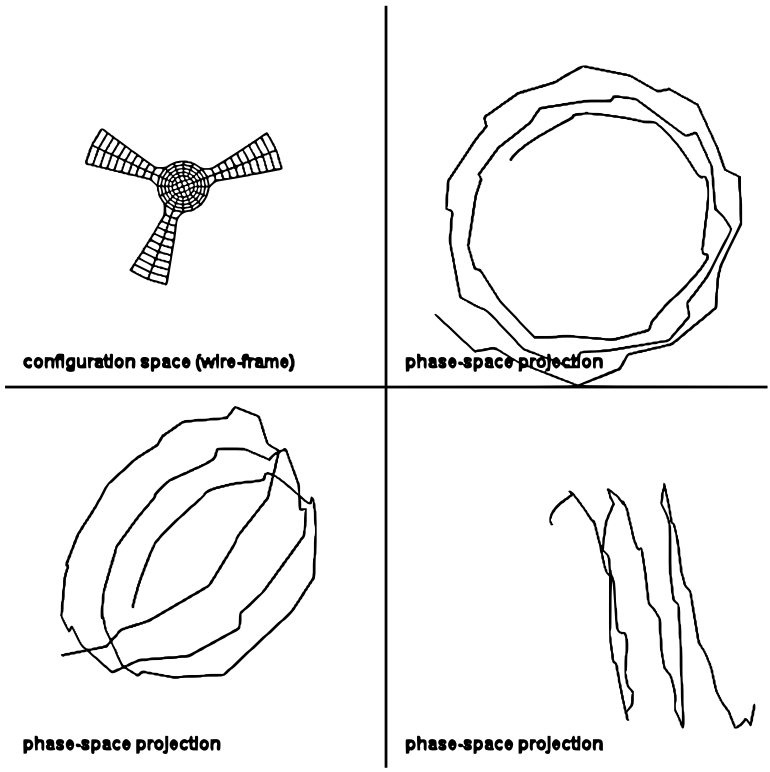


Figure 2: freely flying 2-dimensional hyper-elastic rotor blade, 140 4-node elements $\rightarrow$ 716 DOFs, different views of the $\{q_1, p_1, p_2\}$-projection for a single node

2 Geometric integration (Lie-group methods)

In order to exploit the geometric sructure of Hamiltonian dynamics to the fullest extent, we have to consider integration schemes that generate solutions which remain in the cotangent bundle of the configuration manifold T^*Q. In recent years a powerful machinery of geometric integration (especially Lie-group integrators) has been developed, see for example [FMMK99, IMNZ00, HLW02]. We will present its basic ingredients in the following subsections.

2.1 Lie-groups and Lie-algebras

We will start by looking at Lie-groups. A Lie-group is a group with a smooth (differentiable) group operation and smooth inversion. Therefore it can also be viewed as a

differentiable manifold. Several well-known examples [Olv93] are listed below:

$$\begin{aligned} GL(n,\mathbb{R}) &= \{g \in Mat(n\times n,\mathbb{R}) | \det g \neq 0\} && (1)\\ SL(n,\mathbb{R}) &= \{g \in GL(n,\mathbb{R}) | \det g = 1\} && (2)\\ O(n,\mathbb{R}) &= \left\{g \in GL(n,\mathbb{R}) | g^T g = e\right\} && (3)\\ SO(n,\mathbb{R}) &= \{g \in O(n,\mathbb{R}) | \det g = 1\} && (4) \end{aligned}$$

The Lie-algebra $\mathfrak{g}$ corresponding to a Lie-group G can be defined as the tangent space to the group at the identity element $e \in G$

$$\mathfrak{g} = T_e G.$$

$\mathfrak{g}$ is a vector space equipped with a skew-symmetric, bilinear product called Lie-bracket

$$[\,.\,,\,.\,] : \mathfrak{g}\times\mathfrak{g} \to \mathfrak{g}$$

which satisfies Jacobi's identity: let $a, b, c \in \mathfrak{g}$

$$[a,[b,c]] + [b,[c,a]] + [c,[a,b]] = 0.$$

The Lie-algebras corresponding to the Lie-groups (1)−(4) defined above are the following:

$$\begin{aligned} \mathfrak{gl}(n,\mathbb{R}) &= \{X, Y \in Mat(n\times n,\mathbb{R}) | [X,Y] = XY - YX\} && (5)\\ \mathfrak{sl}(n,\mathbb{R}) &= \{X \in \mathfrak{gl}(n,\mathbb{R}) | \mathrm{tr} X = 0\} && (6)\\ \mathfrak{o}(n,\mathbb{R}) &= \left\{X \in \mathfrak{gl}(n,\mathbb{R}) | X^T + X = 0\right\} && (7)\\ \mathfrak{so}(n,\mathbb{R}) &= \mathfrak{o}(n,\mathbb{R}) && (8) \end{aligned}$$

Because of this tangent space construction there are several choices of coordinate charts that map from the Lie-algebra back to the Lie-group. One such coordinate chart is the exponential map $\exp : \mathfrak{g} \to G$ which maps the whole Lie-algebra to a neighbourhood of $e \in G$. Because of the smooth group operation required above, this map can be extended to the whole group G via left- or right-translation. For matrix groups the exponential map is just the standard matrix exponential.

2.2 Lie-group and Lie-algebra actions

Lie-group and Lie-algebra actions describe the effect of group and algebra elements on points of a manifold, think of rotations $R \in SO(3)$ acting on points of $\mathbb{R}^3$ for example.
A (left) *Lie-group action* is a map

$$\Lambda : G \times \mathcal{M} \to \mathcal{M}.$$

It induces a corresponding *Lie-algebra action*

$$\lambda : \mathfrak{g} \times \mathcal{M} \to \mathcal{M}$$

via

$$\lambda(v,p) = \Lambda(\exp(v),p) \qquad v \in \mathfrak{g},\ p \in \mathcal{M}.$$

Next we define a map $\lambda_* : \mathfrak{g} \to \mathcal{X}(\mathcal{M})$ from the Lie-algebra to the set of vector-fields on $\mathcal{M}$ pointwise as

$$(\lambda_* v)(p) = \left.\frac{d}{dt}\right|_{t=0} \lambda(tv,p)$$

A differential equation on the manifold can now be written as

$$y' = \mathcal{F}_\lambda(t,y) = (\lambda_* f(t,y))\,(y), \qquad y(0) = p \in \mathcal{M} \tag{9}$$

where $f : \mathbb{R} \times \mathcal{M} \to \mathfrak{g}$.

These definitions allow us to state the main paradigm of Lie-group methods as follows: The solution of the differential equation (9) for small $t \in \mathbb{R}^+$ is

$$y(t) = \lambda(u(t),p)$$

where $u(t) \in \mathfrak{g}$ satisfies

$$u' = \tilde{f}(u) = d\exp_u^{-1}\left(f(t,\lambda(u,p))\right), \qquad u(0) = 0 \in \mathfrak{g}. \tag{10}$$

This statement follows by the commutativity of the following diagram

$$\begin{array}{ccc} T\mathfrak{g} & \xrightarrow{\lambda'_p} & T\mathcal{M} \\ {\scriptstyle \tilde{f}}\uparrow & & \uparrow{\scriptstyle \mathcal{F}} \\ \mathfrak{g} & \xrightarrow[\lambda_p]{} & \mathcal{M} \end{array}$$

One problem we have not yet considered is the non-commutativity of general Lie-groups and Lie-algebras. Let $u_1, u_2 \in \mathfrak{g}$, then

$$\begin{array}{rrcl} \Lambda(\exp(u_1),\Lambda(\exp(u_2),p)) & = & \Lambda(\exp(u_1)\cdot\exp(u_2),p) \\ & = & \Lambda(\exp(\mathcal{B}(u_1,u_2),p) \\ \Rightarrow \qquad \lambda(u_1,\lambda(u_2,p)) & = & \lambda(\mathcal{B}(u_1,u_2),p) \end{array}$$

The symbol $\mathcal{B}(.,.)$ is the Baker-Campbell-Hausdorff formula which has the following explicit form in terms of iterated commutators

$$\begin{aligned} \mathcal{B}(u_1,u_2) \;=\; & u_1 + u_2 + \frac{1}{2}[u_1,u_2] \\ & + \frac{1}{12}[u_1,[u_1,u_2]] - \frac{1}{12}[u_2,[u_1,u_2]] + \ldots . \end{aligned}$$

In order to solve differential equations on manifolds it is sufficient to solve corresponding equations in the Lie-algebra which is a linear space. Therefore standard integration

schemes can be applied on the algebra-level and the solution mapped back to the nonlinear manifold via appropriate coordinate mappings. Suitable combinations of these have been developed in recent years, foremost the *Runge-Kutta-Munthe-Kaas* methods which usually require many iterated commutator evaluations and *Crouch-Grossman* methods which evaluate vector field flows via matrix exponentials.

3 A model system on the sphere S^2

This section sets out to illustrate the benefit of the presented geometric integration techniques. We introduce the following ordinary first-order differential equation on $\mathbb{R}^3$

$$\dot{\boldsymbol{x}} = \begin{bmatrix} -y + xz^2 \\ x + yz^2 \\ -z(x^2+y^2) \end{bmatrix}, \quad \boldsymbol{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \in \mathbb{R}^3. \tag{11}$$

A number of analytical observations on the behaviour of solutions to this equation can be made. Calculating the time derivative of the squared solution-norm yields

$$\begin{aligned} \frac{d}{dt}||\boldsymbol{x}(t)||^2 &= \frac{d}{dt}\left[\boldsymbol{x}(t)\cdot\boldsymbol{x}(t)\right] = 2\boldsymbol{x}(t)\cdot\dot{\boldsymbol{x}}(t) \\ &= 2\left[-xy + x^2z^2 + yx + y^2z^2 - z^2x^2 - z^2y^2\right] = 0 \end{aligned}$$

From this it is obvious that equation (11) has spherical symmetry and it suffices to study its behaviour on the unit sphere $S^2 \subset \mathbb{R}^3$. Furthermore the problem restricted to S^2 has two fixed points (the poles $\boldsymbol{x}_\pm = [0,0,\pm 1]^T$) and an attractor (the equator $z(t) = 0\,\forall t$).

In order to use the Lie-group based methods introduced before we have to realize that S^2 can also be interpreted as a Lie-group, namely $S^2 \cong SO(3)$.
The corresponding Lie-algebra $T_eSO(3) = \mathfrak{so}(3)$ is the set of all skew-symmetric matrices and the exponential map $\exp : \mathfrak{so}(3) \to SO(3)$ is the standard matrix exponential.
In order to formulate equation (11) on the Lie-algebra $\mathfrak{so}(3)$ we need another convenient fact namely $\mathfrak{so}(3) \cong \mathbb{R}^3$ via the hat-map

$$\mathbb{R}^3 \ni \boldsymbol{\omega} \mapsto \hat{\omega} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \in \mathfrak{so}(3) \tag{12}$$

Since any vector-field on S^2 can be written as cross product $\boldsymbol{\omega} \times \boldsymbol{x}$ we get the differential equation on the algebra as

$$\dot{\boldsymbol{x}} = \begin{bmatrix} -y + xz^2 \\ x + yz^2 \\ -z(x^2+y^2) \end{bmatrix} = \begin{bmatrix} \omega_2 z - \omega_3 y \\ \omega_3 x - \omega_1 z \\ \omega_1 y - \omega_2 x \end{bmatrix} = \boldsymbol{\omega} \times \boldsymbol{x}$$

where obviously $\boldsymbol{\omega} = \left[-z[x+y], z[x-y], x^2+y^2\right]^T$. Therefore using the definition of

the hat-map (12) a Lie-type integrator with fixed stepsize h takes the following form

$$\boldsymbol{x}_{n+1} = \exp\left(h\hat{\omega}_n\right)\boldsymbol{x}_n = \exp\left(h\begin{bmatrix} 0 & -[x_n^2+y_n^2] & z_n[x_n-y_n] \\ x_n^2+y_n^2 & 0 & z_n[x_n+y_n] \\ -z_n[x_n-y_n] & -z_n[x_n+y_n] & 0 \end{bmatrix}\right)\boldsymbol{x}_n.$$

We compare this geometric integrator with three classic integration schemes that do not take into account the symmetry of the problem: Euler-forward, Euler-backward (both with fixed step-size) and a standard 4th order Runge-Kutte scheme with variable step-size (MATLAB ode45).

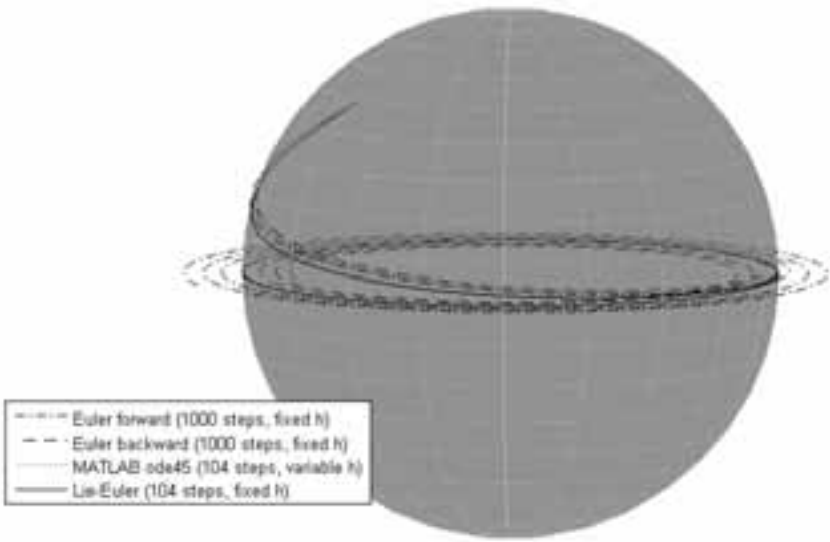


Figure 3: Solution trajectories "on" S^2 for the start value $\left[\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right]^T$. Note that the Euler-forward and Euler-backward solutions both drift away from the solution manifold (outwards and inwards respectively) while the standard 4th order Runge-Kutta (MATLAB ode45) and the Lie-Euler scheme seem to perform much better even with only approximately 10% of the number of steps.

We start the computation at an initial value $\boldsymbol{x}(0) = \left[\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right]^T$ and choose stepsizes for the classic Euler-forward and Euler-backward schemes as 0.02, 0.04, 0.002 and 0.0002 while fixing the number of steps for MATLAB ode45 and the geometric Lie-Euler-scheme at 104. The rather poor performance of the standard Euler-schemes with regard to preserving the spherical symmetry of equation (11) is apparent in figure 3.

Because the 3-dimensional plot employed before is insufficient to determine the amount of deviation from the exact solution manifold we calculate the vector-norm for the different solution schemes and plot them in figure 4 for the standard Euler stepsize of 0.002. While

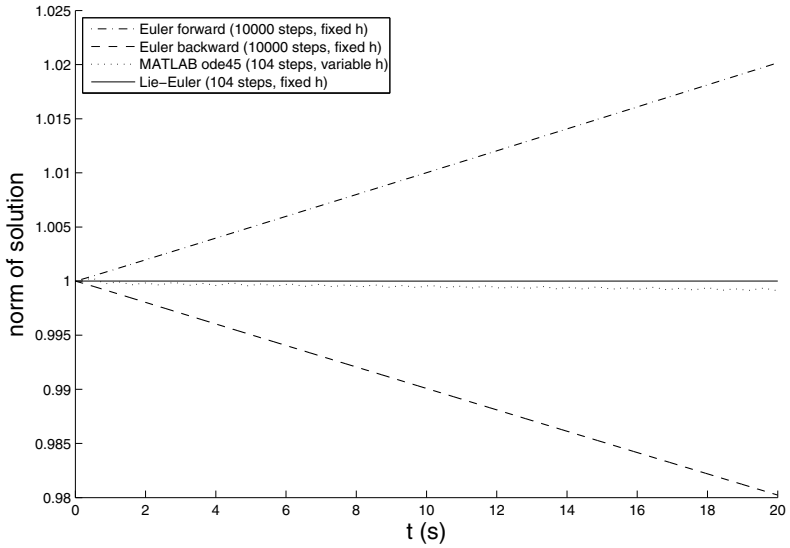


Figure 4: Deviation of trajectories from the configuration manifold S^2 for the different integration schemes illustrated by the fact that $\boldsymbol{x}(t) \in S^2 \Leftrightarrow ||\boldsymbol{x}(t)|| = 1$.

the classic Euler-schemes still deviate strongly from the expected result of $||\boldsymbol{x}(t)|| = 1$ for this step-size, both the standard 4th order Runge-Kutta and the Lie-Euler scheme perform rather well with only 104 steps.

Since the equator is an attractor of the differential equation an exact analytical solution can be stated for start values $z(0) = 0$ as

$$\boldsymbol{x}(t) = \begin{bmatrix} \cos(\varphi_0 + t) \\ \sin(\varphi_0 + t) \\ 0 \end{bmatrix} \qquad \varphi_0 \in \mathbb{R}.$$

Using this result the deviation of the four numerical integration schemes from the exact solution on the equator can be investigated. A start value $\varphi_0 = 0.5\,\pi$ is choosen and the stepsize for the standard Euler-schemes is varied in the same manner as above.

Figure 5 illustrates the fact that in order to reach an approximately equivalent accuracy the number of steps for the two standard Euler-schemes has to be increased to about 1000 times the number of MATLAB ode45 steps. Also note that the Lie-Euler integrator produces the exact solution up to machine precision with only 104 steps.

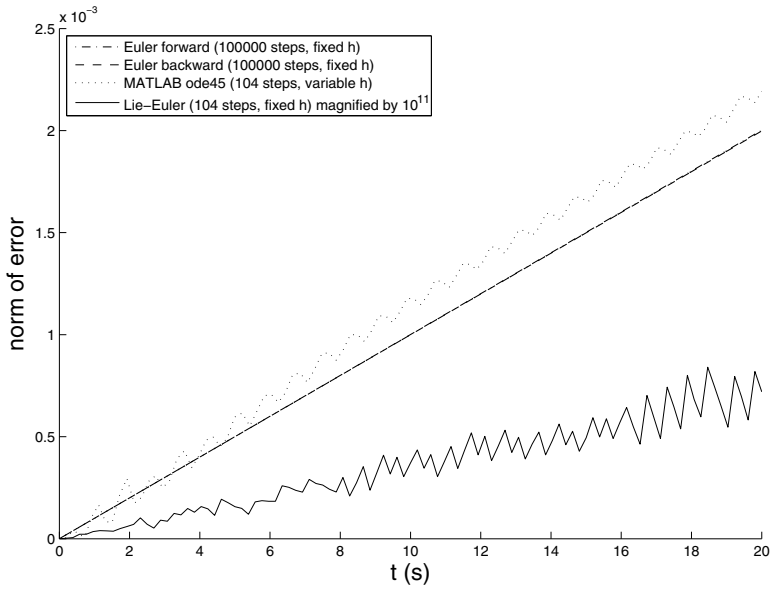


Figure 5: Comparison to the exact analytical solution on the equator of S^2. Note that the error of the Lie-Euler integrator is scaled by a factor of 10^{11}. The error values of the two classic Euler schemes are almost indistinguishable.

4 Conclusions

In this contribution we have demonstrated the common problems in understanding the dynamic behaviour of engineering structures caused by the high dimensionality of their respective phase-spaces. In order to reduce this high dimensionality simpler model systems had to be studied. The intrinsic geometric structure of the Hamiltonian framework of dynamics led to the consideration of geometry preserving integrators of Lie-type. An example system has been cast into this Lie-type framework which made its dynamics approachable by new geometric numerical integration schemes. A comparison with standard integration schemes showed the superiority of this specific geometric integration scheme.

Acknowledgements Financial support by the DFG within the IRTG 1131 is gratefully acknowledged.

References

[FMMK99] Stig Faltinsen, Arne Marthinsen, and Hans Z. Munthe-Kaas. Multistep Methods Integrating Ordinary Differential Equations on Manifolds. Technical Report Numerics No. 3/1999, The Norwegian University of Science and Technology, Trondheim, Norway, 1999.

[Gro04] M. Groß. *Conserving Time Integrators for Nonlinear Elastodynamics*. Dissertation, Lehrstuhl für Technische Mechanik, Technische Universität Kaiserlautern, 03 2004.

[HLW02] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*, volume 31 of *Springer series in computational mathematics*. Springer-Verlag, New York, 2002.

[IMNZ00] A. Iserles, H. Z. Munthe-Kaas, S. P. Nørsett, and A. Zanna. Lie-group methods. *Acta Numerica*, 9:215–365, 2000.

[MR94] Jerrold E. Marsden and Tudor S. Ratiu. *Introduction to Mechanics and Symmetry*. Texts in appliexd mathematics. Springer Verlag, New York, 1994.

[Olv93] Peter J. Olver. *Applications of Lie Groups to Differential Equations*. Graduate Texts in Mathematics. Springer Verlag, New York, 2nd edition, 1993.

[SK03] J. J. Sławianowski and V. Kovalchuk. Invariant Geodetic Problems on the Affine Group and Related Hamiltonian Systems. *Rep. Math. Phys.*, 51(2/3):371–379, 2003.

PointCloudXplore 2: Visual Exploration of 3D Gene Expression

O. Rübel[1,2,3], G.H. Weber[3], M.-Y. Huang[2], E.W. Bethel[3] , S.V.E. Keränen[4], C.C. Fowlkes[5], C.L. Luengo Hendriks[7], Angela H. DePace[8], L. Simirenko[4], M.B. Eisen[4], M.D. Biggin[4], H. Hagen[1], J. Malik[6], D.W. Knowles[7] and B. Hamann[1,2,3]

[1] International Research Training Group, University of Kaiserslautern, Germany
[2] Institute for Data Analysis and Visualization, University of California, Davis
[3] Computational Research Division, Lawrence Berkeley National Laboratory, Berkeley
[4] Genomics Division, Lawrence Berkeley National Laboratory, Berkeley
[5] Computer Science Department, University of California, Irvine
[6] Computer Science Division, University of California, Berkeley
[7] Life Sciences Division, Lawrence Berkeley National Laboratory, Berkeley
[8] Department of Molecular and Cellular Biology and the Center for Integrative Genomics, University of California, Berkeley

Abstract: To better understand how developmental regulatory networks are defined in the genome sequence, the *Berkeley Drosophila Transcription Network Project* (*BDNTP*) has developed a suite of methods to describe 3D gene expression data, i.e., the output of the network at cellular resolution for multiple time points. To allow researchers to explore these novel data sets we have developed PointCloudXplore (PCX). In PCX we have linked physical and information visualization views via the concept of brushing (cell selection). For each view dedicated operations for performing selection of cells are available. In PCX, all cell selections are stored in a central management system. Cells selected in one view can in this way be highlighted in any view allowing further cell subset properties to be determined. Complex cell queries can be defined by combining different cell selections using logical operations such as AND, OR, and NOT. Here we are going to provide an overview of PointCloudXplore 2 (PCX2), the latest publicly available version of PCX. PCX2 has shown to be an effective tool for visual exploration of 3D gene expression data. We discuss (i) all views available in PCX2, (ii) different strategies to perform cell selection, (iii) the basic architecture of PCX2., and (iv) illustrate the usefulness of PCX2 using selected examples.

1 Introduction

Understanding how embryo development is controlled is a fundamental question in biology. Specific combinations of developmental regulatory factors –forming parts of complex genetic regulatory networks– are responsible for determining the cell fates [Law92]. Biologists have typically analyzed gene expression and morphology by visual inspection of photomicrographic images. In order to understand animal development, methods to computationally describe gene expression at cellular resolution are required. To address this challenge, the BDTNP has developed image analysis methods to extract information about gene expression from 3D imaging data, using early *Drosophila melanogaster* embryos as a model [BDT]. Blastoderm stage *Drosophila* embryos are stained, mounted, and imaged using laser microscopy. The image stacks are then converted into matrices specifying the position of blastoderm nuclei and the expression levels of select genes around each nucleus [LKF+06, KFL+06]. These novel, so called PointCloud datasets, promise to be an invaluable resource for studying embryo development. Due to the limited number of different fluorophores we can spectrally distinguish as well as the difficulty in adding multiple labels to embryos it is, however, experimentally not practical to obtain the expression of more than a few genes in a single embryo. To make possible to compare the relationships between regulators and their many target genes in a common coordinate framework, a set of PointClouds are registered into one or more Virtual PointClouds [FLK+05, FLK+07].

To help biologists to compare and analyze 3D gene expression data, we developed PointCloudXplore (PCX) a tool specifically designed for exploration of PointCloud data. In PCX we have linked physical and information visualization views via the concept of brushing (cell selection). Here we are going to introduce PointCloudXplore 2 (PCX2), the latest publicly available version of PCX. We are going to discuss all views available in PCX2 and describe how linking of the views via cell selection is used for effective data exploration. Afterwards, we will explain different dedicated strategies for performing cell selection and describe the basic architecture of PCX2. The following parts of this manuscript are structured as follows. In Section 2 we will provide an overview of related work. The different visualization techniques available in PCX2 are then described in Section 3. In Section 4.1 and 4.2, linking of the views via cell selection and all operations for performing and combining cell selections will be explained. A brief overview of the implementation of PCX2 will be provided in Section 5. In Section 6 we present our conclusions and describe future plans.

2 Previous Work

PCX2 makes use of the established concept of linking multiple views to visualize high-dimensional 3D gene expression data [BWK00]. In the WEAVE system, e.g., a combination of linked physical views and information visualization views was used for exploration of cardiac simulation and measurement data [GRW+00]. Henze [Hen98] proposed a system of multiple views (called Portraits in his paper) for exploring time-varying computational fluid dynamics (CFD) data sets. Both Henze's system for CFD data and the

WEAVE system use linked views to define features in a data set by refining queries based on brushes. In PCX2 brushes are called cell selectors since this term is found to be more intuitive by target users. Doleisch et al. formalized the concept of defining features via queries using information visualization views and utilizing logical operations to combine several brushes [DGH03] (see also Hauser et al. [HLD02]). A variety of extensions to 3D scatter plots were proposed by Piringer et al. [PKH04] and Kosara et al. [KSH04] improving depth perception and perception of the sample distribution in all dimensions.

Parallel coordinates are a standard visualization for high-dimensional data while many extensions to standard parallel coordinates have been developed to make them more useful. Wegman and Luor [WL97] proposed usage of transparency and "over-plotting" translucent data points/lines to reveal inherent data characteristics by highlighting dense areas. Fua et al. [FWR99] proposed hierarchical parallel coordinates, including several techniques for visualization of selected subsets of the data which are in part also available in PCX2.

3 Visualizing 3D Gene Expression

PCX2 provides different visualizations for 3D gene expression data. In physical views a 3D model, or 2D projections of the 3D model, are used to visualize spatial gene expression patterns (Section 3.1). To support analysis of relationships between the expression of genes directly in expression space, PCX2 features scatter-plots (Section 3.2) as well as parallel coordinates (Section 3.3) as abstract information visualization views. To allow comparison of the expression of genes in a single cell, the Cell Magnifier shows the expression of all genes in a user defined cell as bar-graph (Section 3.4).

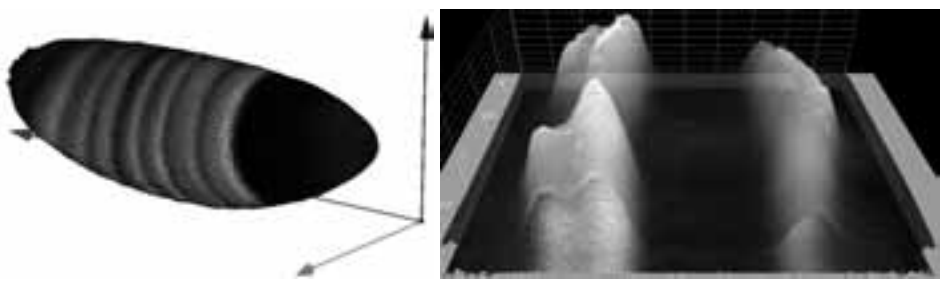


Figure 1: 3D Cell View with expression levels of the genes *eve* and *ftz* shown via color (left). Unrolled View and expression surface showing the expression pattern of the gene *giant* (*gt*) during early embryo development (right).

3.1 Physical Views

Physical views are used in PCX2 to visualize and analyze spatial patterns of gene expression. PCX2 features 3D as well as 2D visualizations of the embryo. In the 3D views

spherical icons (not shown) or polygonal faces are used to represent individual cells (see Figure 1 left). To also take the movement of cells over time into account the user can choose which cell positions from which time-step should be used in the visualization. By using different projections of the 3D embryo model dedicated 2D visualizations of the embryo are created. In Figure 1 (right) the embryo is projected to a 2D plane using a cylindrical projection. PCX2 also allows one to use orthographic projection to create three additional 2D views of the embryo described in more detail in Weber et al. [WRH+07].

In addition to providing an overview of the entire embryo, projecting the embryo to a plane has the advantage of freeing one dimension up for displaying additional information. In PCX2, this "free" dimension can be used to display gene expression values as surface plots. The height of an expression surface is determined by the expression values measured for the gene it represents while the xy-position of surface points are defined by the position of the cells in the underlying 2D physical view. As shown in the right panel of Figure 1, such surface plots enable a more quantitative analysis of gene expression data than possible when just using color. Multiple surfaces can be used to compare quantitative relationships between genes. More details about physical views and their use for the analysis of 3D gene expression are provided in [RWK+06, WRH+07].

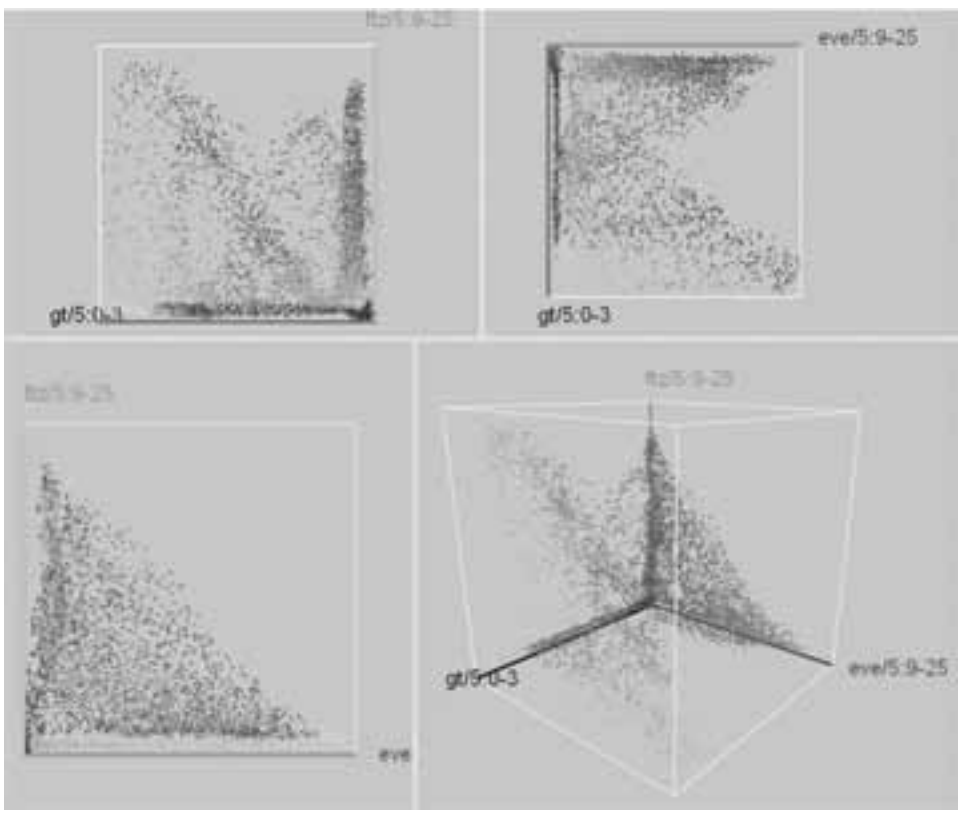


Figure 2: 4-panel layout of three 2D and the according 3D scatter-plot as displayed in PCX2

3.2 Scatter-plots

In contrast to physical views of the embryo, abstract views, such as scatter-plots, discard spatial information and show the quantitative relationships between multiple genes expression in one or all cells. In a scatter-plot, three genes are selected and mapped to the three axes of the Cartesian coordinate system. Each of the axes represents the expression of one gene ranging from no expression at the origin to maximum relative expression. A single point is used to represent each cell of the embryo while the location of the point is specified according to the relative gene expression levels measured for the corresponding cell. To enhance depth perception and to better distinguish separate points we use color, halos, and alpha blending. As shown in Figure 2, in PCX2 we use a 4-panel layout in which a 3D scatter-plot (lower-right panel) is augmented with a set of 2D scatter plots showing the expression relationships between the three possible gene-pairs in the 3D plot. See Weber et al. [WRH+07] for more detailed information.

Scatter-plots are in general found to be an intuitive visualization and are used to reveal information about relationships between genes. In the scatter-plot of *eve* and *ftz* (Figure 2 lower-left panel), e.g., a clear anti-correlation between the two genes can be seen.

3.3 Parallel Coordinates

To make visualization of the expression of many genes in parallel possible PCX2 features parallel coordinates. Parallel coordinates were proposed independently by Inselberg [Ins84] and Wegman [Weg90] and are a common information visualization technique for high-dimensional data sets. Each data dimension (gene) is represented in parallel coordinates by one parallel axis. Each data sample (cell) has a set of associated expression values defining a point on each parallel axis. By connecting the corresponding points of neighboring axes each cell can be represented by a data line, i.e., a "zigzag" line connecting adjacent parallel axes. The intersection point of the data line with each vertical axis corresponds to the value of the sample for the corresponding dimension (i.e., the relative expression level for the corresponding gene in that cell). In Figure 3, an example parallel coordinate view is shown visualizing the expression of six selected genes in all cells of the embryo.

3D-parallel coordinates are a direct extension of 2D-parallel coordinates in which the parallel axes are extruded into the z dimension. As illustrated in Figure 4, lines are sorted along the z axis with respect to a selected data dimension so that each parallel axis is *de facto* a 2D-scatter-plot. As in 2D-parallel coordinates corresponding points of neighboring axes are connected via line segments. In PCX2 the same information is displayed along the z dimension for all axes, in this way a 3D-plot of non-intersecting parallel data lines is created each representing one cell. Data lines can be sorted in parallel coordinates according to the position of cells along the anterior-posterior axis (AP)(back to front) or dorsal-ventral axis (DV)(around the embryo) which can be derived from the Unrolled View. Beside this, lines can also be sorted according to the expression of a selected gene.

PCX2 supports many important extension to standard parallel coordinates such as use of transparent data lines, semi-transparent color bands to visualize cell selections, or dimensional zooming. Besides this also highlighting and animation of line traces is supported by PCX2. For more detailed information see [RWK+06].

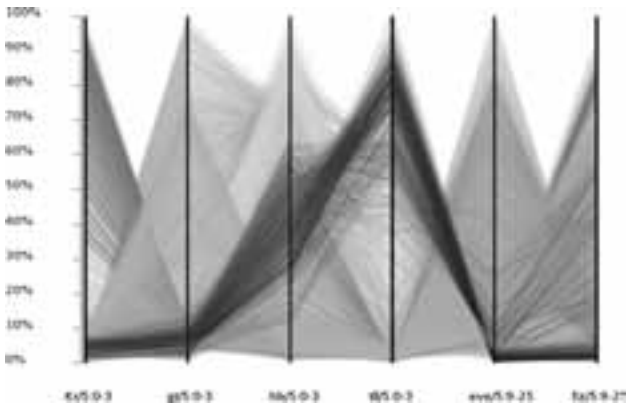


Figure 3: 2D Parallel coordinate view showing the expression of six selected genes in all blastoderm cells of the embryo (reproduced in color on p. 196).

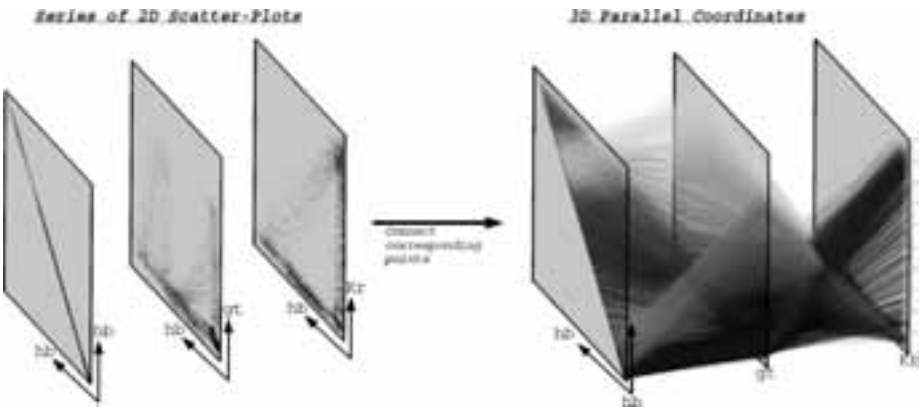


Figure 4: 3D Parallel coordinates are a direct extension to regular parallel coordinates and combine the principle of 2D-scatter-plots and 2D-parallel coordinates (reproduced in color on p. 196).

3.4 Cell Magnifier

Unlike all other views currently available in PCX2, the Cell Magnifier concentrates not on comparing gene expression values in different cells but on comparing expression values in just one cell. A bar graph is used to visualize gene expression values. Each gene is represented by one bar while the exact measured expression value is displayed in addition

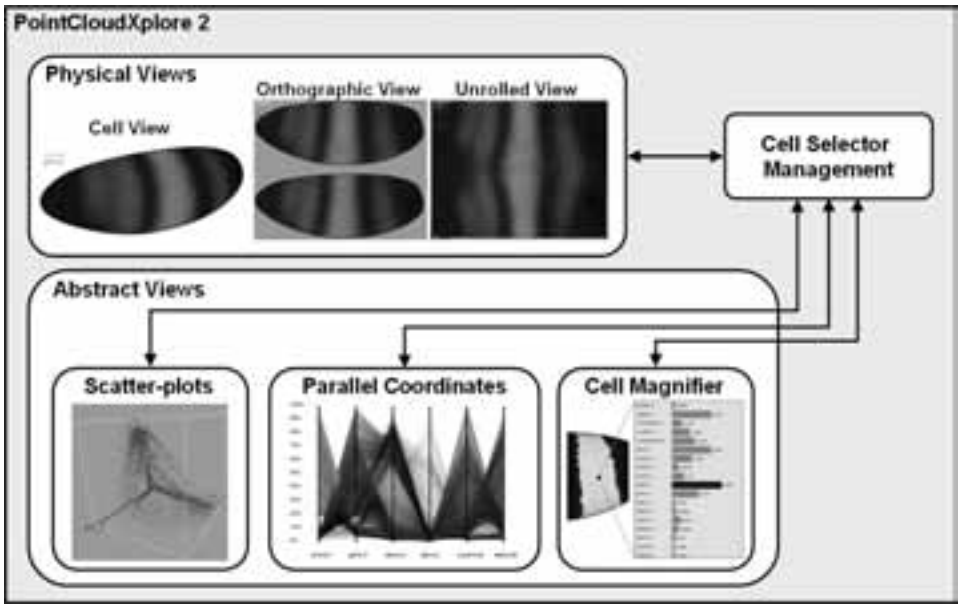


Figure 5: PointClouxXplore 2 system design. All views are linked via a central cell selector management system.

beside each bar. The cell to be displayed in the Cell Magnifier can be selected in any physical view and is highlighted by graying it out. An example illustrating use of the Cell Magnifier to analyze the expression in a cell located in the posterior *giant* expression domain will be provided later in Figure 8. For more detailed information about the Cell Magnifier see Weber et al. [WRH+07].

4 Linking the Views via Cell Selection

The different views described in the previous section are useful in their own right and can be used individually to mine data sets for new information. It is, however, often useful to compare and correlate the information of different views. The principle of cell selection (also referred to as brushing) provides effective means for linking different views. By selecting cells of interest in one view and highlighting the same cell subset in all other views interactive exploration of the data becomes possible. Selection of cells can be performed in each view of PCX2 with respect to different data properties, e.g. cell position or gene expression (see Section 4.1). To allow for definition of more complex cell queries, cell selections can be combined using logical operations (see Section 4.2).

In PCX2 all cell selections are stored in dedicated so called cell selector objects. All cell selectors are stored and managed in a central cell selector management system. All views

have access to the same set of cell selectors via this central management system. This simple but effective system design allows one to easily integrate new views in PCX2 and link them to the already existing views. Figure 5 illustrates the basic system design of PCX2. The different views are each independent of one another and are interconnected only indirectly via the central cell selector management. Independent of the criteria employed to perform cell selection, each cell selector defines for each cell whether it is the selected or not. In addition, each cell selector is required to have a name, color, and a flag defining whether the cell selector should be displayed or not. Therefore, a common base class can be used to define properties common for all types of cell selectors. By using such a common interface all types of cell selectors can be efficiently displayed in all views. Which types of cell selectors can be edited and created in the different views then depends on according to which data properties cell selection can be performed in a view.

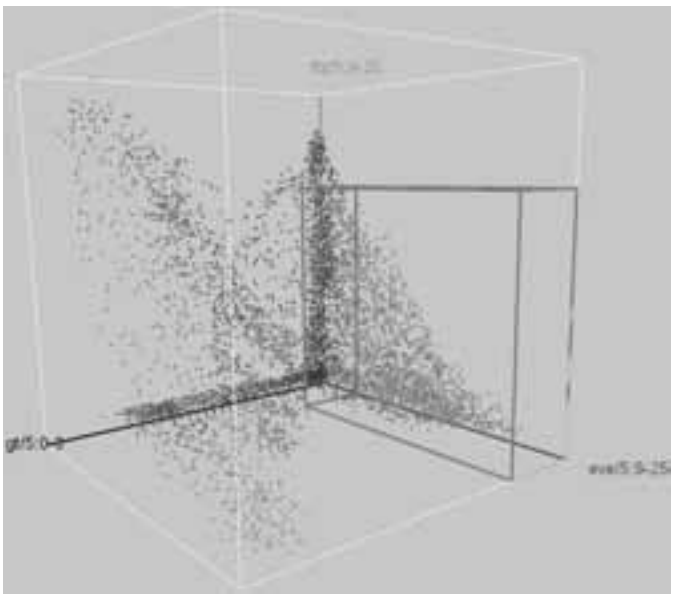


Figure 6: By drawing an axes aligned box in a scatter-plot one can select cells with respect to the expression of up to three genes (reproduced in color on p. 196).

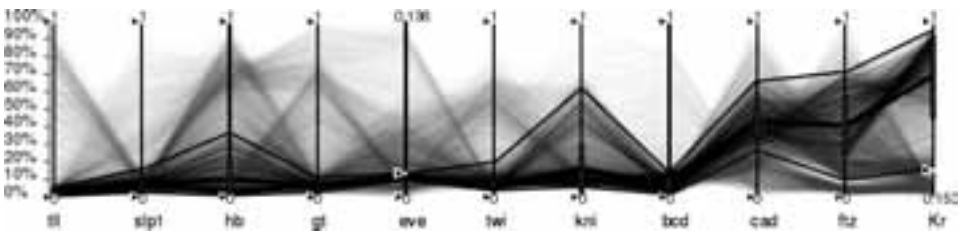


Figure 7: By defining ranges in expression using handles attached to each parallel axis it is possible to select cells with respect to the expression of all genes displayed in the parallel coordinate view.

4.1 Basic Cell Selection Operations

Cell selection can be performed with respect to different data properties in different views. The basic cell selection operations performed directly by the user can be subdivided into three different categories:

- **Cell selection based on cell positions** can be performed in any physical view. By drawing on the surface of the embryo the researcher can define which cells in which region of the embryo are of interest.
- **Cell selection based on gene expression** can be performed in scatter-plots as well as parallel coordinates. Dedicated cell selectors allow the user to select cells with respect to ranges in gene expression. All cells contained in a so defined multi-dimensional box in gene expression space are marked as selected. In a scatter-plot cell selection can be performed with respect to the expression of up to three genes by defining an axis-aligned box in the plot (see Figure 6). In parallel coordinates cell selection can be performed with respect to all n displayed data dimensions by defining ranges in gene expression with sliders attached to each parallel axis (see Figure 7).
- **Cell selection based on spatial information and gene expression** can be performed using seed cell selection and the Cell Magnifier. Seed cell selection requires the index of a cell as well a set of expression range(s) for selected genes as input. Using the cell currently displayed in the Cell Magnifier as seed point to initiate the selection process a flood fill method is used to identify all cells in a contiguous region whose expression levels lie within the specified expression range(s). In the example shown in Figure 8, seed cell selection is used to select the posterior expression domain of the *giant* expression pattern. Besides seed cell selection one can also use 3D parallel coordinates to select cells with respect to spatial information as well as gene expression. If data lines are sorted in 3D parallel coordinates with respect to AP- or DV-cell positions then ranges in AP/DV-space can be used in addition to range(s) in gene expression to perform cell selection.

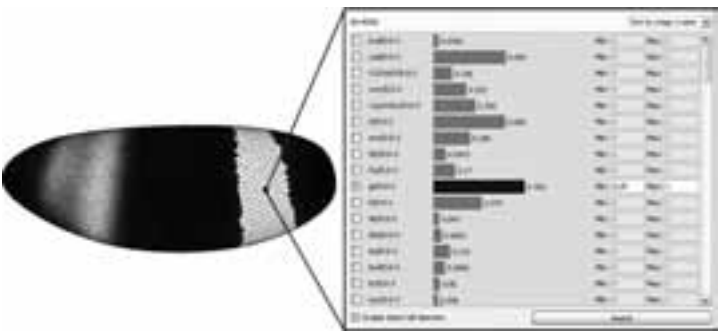


Figure 8: Using seed cell selection it is possible to easily and accurately select contiguous regions on the embryo that show specific expression properties.

4.2 Combining Cell Selections

To make definition of more complex cell queries possible, basic cell selections can be combined in PCX2 using logical operations such as AND, OR, and NOT. The AND-operator defines the intersection of two cell selections and the OR-operator their union. The NOT-operator then inverts a cell selection and selects all cells not selected by the input cell selection. Logical operators are implemented in PCX2 as special types of cell selectors that define a new selection of cells based on the cell selection defined by one or two other cell selectors. A logical operator can again function as input to another logical operator allowing complex cell queries to be defined. Such cell queries consisting of both basic and logical cell selectors can be represented as a tree structure in which the basic cell selectors are always leaves of the tree. In Figure 9 cell selectors are used to define the patterns of the genes *giant* (*gt*), *hunchback* (*hb*), and *Krüppel* (*Kr*) via thresholding. The thresholded patterns are then combined using logical operators in order to validate a model for regulation of stripe two of the *eve* expression pattern. A comparison of the defined selection with the thresholded pattern of the gene *eve* shows a close match in the domain of *eve* stripe two indicating that *gt*, *hb*, and *Kr* may be able to regulate this part of the *eve* pattern.

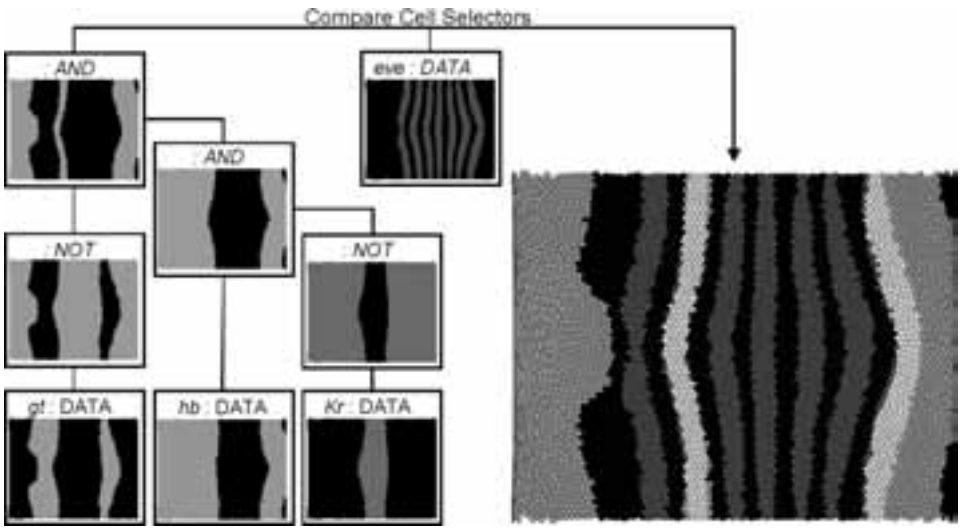


Figure 9: Combining cell selectors defining the patterns of *gt*, *hb*, and *Kr* using logical operators. The selection result is then compared with a cell selector that selects all cells expressing *eve* at medium to high levels by defining the overlay of both selections (reproduced in color on p. 196).

5 Implementation

PCX2 is an interactive exploration tool. All here described views are rendered interactively and all described interactions take, at most, a fraction of a second to complete. PCX2

is implemented as C++ stand-alone application using Trolltechs Qt 4.2 library (http://www.trolltech.com/) and OpenGL (http://www.opengl.org/) as cross-platform widget and graphics libraries. PCX2 is available for Linux, MacOS and Windows. An earlier version of PCX has already been released to the public and can be downloaded free of charge from the BDTNPs web page http://bdtnp.lbl.gov/Fly-Net/bioimaging.jsp?w=pcx. The full version of PCX2 as described here will also be made available for free soon via the BDTNPs web page. PCX2 is used as a means to view the BDTNP's release data sets [BDT].

6 Conclusions and Future Work

PCX has already shown to be a valuable tool to members of the BDTNP as well as users of the publicly available 3D gene expression database of the BDTNP. Along with the next release of the database we will also make PCX2 available for free public download. PCX2 provides all features of PCX and includes additional extension such as 3D parallel coordinates and supports visualization of the latest 3D gene expression data created by the BDTNP. Scatter-plots and parallel coordinates should allow researchers to examine relationships between 20-30 genes.

Here we have provided an overview of all views available in PCX2 and shown how linking of the views via cell selection can be used for effective data exploration. Using selected examples, we have shown how one can analyze the spatial expression pattern of genes using physical views and expression surfaces. Information visualization views such as scatter-plots and parallel coordinates then enable analysis of gene inter-relationships directly in gene expression space. We have illustrated how dedicated cell selection operations such as seed cell selection can be used to effectively define cells of interest, such as an expression domain of a selected gene. We have also shown how binary logical models of genetic interactions can be analyzed by combining thresholded patterns of selected genes using logical operations.

As the BDTNP continues to collect data for many more genes, this imposes several new challenges for future work. For example, with our current approach it is possible to effectively visualize expression levels of five to six genes simultaneously in a physical view. (The exact number depends on the spatial distribution of the patterns.) Mapping an even larger number of genes to color will be one of the challenges arising from the increasing number of genes. By integrating automated data analysis methods, such as clustering and dedicated dimension reduction techniques into PCX, we hope to address the arising need to examine the expression of several hundred genes.

Acknowledgments This work was supported by the National Institutes of Health through grant GM70444, by the National Science Foundation through award ACI 9624034 (CAREER Award) as part of the Large Scientific and Software Data Set Visualization (LSSDSV) program under contract ACI 9982251, by the LBNL Laboratory Directed Research Development (LDRD) program; and by a large Information Technology Research (ITR)

grant. Work at Lawrence Berkeley National Laboratory is conducted under Department of Energy contract DE-AC02-05CH11231 We thank the members of the Visualization and Computer Graphics Research Group at the Institute for Data Analysis and Visualization (IDAV) at the University of California, Davis; the members of the BDTNP at the Lawrence Berkeley National Laboratory (LBNL), the members of the Visualization Group at LBNL, and the International Research Training Group 1131 at the University of Kaiserslautern.

References

[BDT] BDTNP. http://bdtnp.lbl.gov/Fly-Net/bioimaging.jsp.

[BWK00] M. Q. Wang Baldonado, A. Woodruff, and A. Kuchinsky. Guidelines for using multiple views in information visualization. In *AVI '00: Proceedings of the working conference on Advanced visual interfaces*, pages 110–119, New York, NY, USA, 2000. ACM Press.

[DGH03] H. Doleisch, M. Gasser, and H. Hauser. Interactive Feature Specification for Focus+Context Visualization of ComplexSimulation Data. In G.-P. Bonneau, S. Hahmann, and C. D. Hansen, editors, *Data Visualization 2003 (Proceedings of the Eurographics/IEEE TCVG Symposimon Visualization)*, 2003.

[FLK+05] C. C. Fowlkes, C. L. Luengo Hendriks, S. V. E. Keränen, M. D. Biggin, D. W. Knowles, D. Sudar, and J. Malik. Registering *Drosophila* embryos at cellular resolution to build a quantitative3d map of gene expression patterns and morphology. In *CSB 2005 Workshop on BioImage Data Minning and Informatics*, August 2005.

[FLK+07] C. C. Fowlkes, C. L. Luengo Hendriks, S. V. E. Keränen, G. H. Weber, O. Rübel, M.-Y. Huang, S. Chatoor, L. Simirenko, M. B. Eisen, B. Hamann, D. W. Knowles, M. D. Biggin, and J.Malik. Constructing a quantitative spatio-temporal atlas of gene expression in Drosophila blastoderm. to appear in: Cell (Accepted January 2008).

[FWR99] Y.-H. Fua, M. O. Ward, and E. A. Rundensteiner. Hierarchical parallel coordinates for exploration of large datasets. In *IEEE Visualization 1999*, pages 43–50. IEEE Computer Society Press, 1999.

[GRW+00] D. L. Gresh, B. E. Rogowitz, R. L. Winslow, D. F. Scollan, and C. K. Yung. WEAVE: A system for visually linking 3-D and statistical visualizations,applied to cardiac simulation and measurement data. In Thomas Ertl, Bernd Hamann, and Amitabh Varshney, editors, *Proceedings IEEE Visualization 2000*, pages 489–492, Los Alamitos, CA, USA, 2000. IEEE Computer Society Press.

[Hen98] Chris Henze. Feature detection in linked derived spaces. In David Ebert, Holly Rushmeier, and Hans Hagen, editors, *Proceedings IEEE Visualization '98*, pages 87–94, Los Alamitos, CA, USA, 1998. IEEE Computer Society Press.

[HLD02] Helwig Hauser, Florian Ledermann, and Helmut Doleisch. Angular Brushing of Extended Parallel Coordinates. In *IEEE Symposium on Information Visualization (InfoVis'02)*, pages 127–130. IEEE Computer Society Press, 2002.

[Ins84] Alfred Inselberg. Parallel Coordinates for Multidimensional Displays. In *Spatial Information Technologies for Remote Sensing Today and Tomorrow, The Ninth William T. Pecora Memorial Remote Sensing Symposium*, pages 312–324. IEEE Computer Society Press, 1984.

[KFL+06] S. V. E. Keränen, C. C. Fowlkes, C. L. Luengo Hendriks, D. Sudar, D. W. Knowles, J. Malik, and M. D. Biggin. Three-Dimensional Morphology and Gene Expression in the Drosophila blastoderm at Cellular Resolution I: Dynamics. *Genome Biology*, 7(12):R124, 2006.

[KSH04] R. Kosara, G. N. Sahling, and H. Hauser. Linking Scientific and Information Visualization with Interactive 3D Scatterplots. In *Short Communication Papers Proceedings of the 12th International Conferencein Central Europe on Computer Graphics, Visualization, and Computer Vision (WSCG)*, pages 133–140, 2004.

[Law92] P. A. Lawrence. *The Making of a Fly: The Genetics of Animal Design*. Blackwell Publishing, Inc., 1992.

[LKF+06] C. L. Luengo Hendriks, S. V. E. Keränen, C. C. Fowlkes, L. Simirenko, G. H. Weber, A. H. DePace, C. Henriquez, D. W. Kaszuba, B. Hamann, M.l B. Eisen, J. Malik, D. Sudar, M. D. Biggin, and D. W. Knowles. Three-Dimensional Morphology and Gene Expression in the Drosophila blastoderm at Cellular Resolution I: Data Acquisition Pipeline. *Genome Biology*, 7(12):R123, 2006.

[PKH04] H. Piringer, R. Kosara, and H. Hauser. Interactive Focus+Context Visualization with Linked 2D/3D Scatterplots. In *Proceedings of the Second International Conference on Coordinated & Multiple Views in Exploratory Visualization (CMV'04)*, pages 49–60, Washington, DC, USA, 2004. IEEE Computer Society.

[RWK+06] O. Rübel, G. H. Weber, S. V. E. Keränen, C. C. Fowlkes, C. L. Luengo Hendriks, L. Simirenko, N. Y. Shah, M. B. Eisen, M. D. Biggin, H. Hagen, J. D. Sudar, J. Malik, D. W. Knowles, and B. Hamann. PointCloudXplore: Visual analysis of 3D gene expression data using physical views and parallel coordinates. In B. Sousa Santos, T. Ertl, and K.I Joy, editors, *Data Visualization 2006 (Proceedings of EuroVis 2006)*, pages 203–210, Aire-la-Ville, Switzerland, 2006.

[Weg90] Edward J. Wegman. Hyperdimensional Data Analysis Using Parallel Coordinates. *Journal of the American Statistical Association*, 85(411):664–675, September 1990.

[WL97] Edward J. Wegman and Qiang Luo. High dimensional clustering using parallel coordinates and the grand tour. *Computing Science and Statistics*, 28:361–368, 1997.

[WRH+07] G. H. Weber, O. Rübel, M.-Y. Huang, A. H. DePace, C. C. Fowlkes, S. V. E. Keränen, C. L. Luengo Hendriks, H. Hagen, D. W. Knowles, J. Malik, M. D. Biggin, and B. Hamann. Visual Exploration of Three-dimensional Gene Expression Using Physical Views and Linked Abstract Views. *Accepted for Publication in IEEE Transactions on Computational Biology and Bioinformatics*, 2007.

Geomodeling and Geovisualizations in Urban Planning und Real Estate Industry:

The Example of Office Market Research

Christian von Malottki

International Research Training Group
"Visualization of Large and Unstructured Data Sets"
University of Kaiserslautern
D-67653 Kaiserslautern, Germany
malottki@rhrk.uni-kl.de

Abstract: Modeling, quantitative analysis, and forecasting in urban planning have a tradition since the sixties when very complex models for the whole "system of the city" were developed. After a phase of criticism about these complex black box programs in the eighties the topic got in the research focus again because of the easier possibilities for visualizing the results by the means of GIS. Subsequently, geomodeling is also interesting for more specific questions. The example shown in the paper is office market modeling – with a case study in Stuttgart. Due to higher vacancy rates and the degradation of buildings especially from the sixties and the seventies the subject is relevant for investors and real estate brokers but also for city administrations who try to avoid the degradation of whole areas. The classical time-series based office market models from urban economics describe the movement of the entire market but they do not consider local heterogeneity. Cross- sectional models like hedonic price modeling and an adaptation of the hedonic model for vacancy rates shown in the paper are difficult to couple with forecasting results. The microsimulation approach is the best way to integrate forecasting and a detailed spatial resolution. It consists in simulating movements, location choices, and vacancy at the building level. The paper presents the equations and exemplary results of the different simulation steps.

1 Geomodeling and Geovisualization in Urban Planning

Modeling in urban planning â and simulation as the most exact and methodologically complex way of modeling different future development paths â has its roots in the sixties when âlarge scale urban modelsâ were developed [For69]. Caused by the critics on lacking usability for the practice [Lee73], a general trend against quantitative methodology in urban planning became the mainstream. In the nineties modeling got again a discussed topic with the occurrence of easy to use Geographic Information Systems (GIS) [Bat92]. The easy calculation of variables describing location extended the modeling tradition to a real geomodeling. Beside the standard GIS tools a bundle of simulation tools exists

today [WU04, MSW02]. However, further research has to be done: how to calculate spatially more detailed information and how to expand the modeling approach to urban themes apart from the well discussed land-use subject (e.g. for the real estate market; for the noise problem see [SR05, Rum07]) are the principal questions.

Urban planning is in the focus of politicians, the media, and the public. The lesson learned from the criticized âblack boxâ models of the sixties is the finding that visualization plays a major role in all kind of analysis and forecasting in urban planning. The typical way of visualizing zone or building related output of all kinds of urban models consists in the application of one of the standard GIS software packages. New challenges arise with integration of user generated vector data, e.g. from Google Earth, the availability of 3D-city models with mapped facades [HMM05], interactive WebGIS especially for planning process participation purposes, and the use of new augmented reality techniques for the visualization of urban future alternatives [Wie07].

2 Basics of Office Market Modeling

Office market modeling is the example chosen for the demonstration that the geomodeling and geosimulation approach can also be used for spatial questions that are not in the focus of the existing urban simulation tools.

Reference studies in office market research mainly have an economic background dealing with the cyclical movements (âhog cycleâ) of the markets. The phenomenon occurs because multiple developers start constructing without knowing from each other at a time of high space absorption. They accomplish the construction two to four years after when the market conditions often have changed fundamentally. The consequence is a high rate of vacant buildings. During a decline phase of the market the vacancy shifts from new buildings to old ones by movements. In the recent years the vacancy problem became also a discussed urban planning topic. The first reason is the existence of an increasing stock of old buildings which is left over and can not be let any more. The second reason is the technological and demographic change that will lower the total space required for office uses in most European countries. Office markets are markets with heterogeneous goods. The subsequent question for developers, investors, and urban planning authorities is where respectively in which buildings the problems of vacancy risks or low prices will occur. This requires a micro-level approach analyzing single buildings or geographically detailed zones.

The case study city for the subsequent analysis is Stuttgart in Southwestern Germany. The city with 590,000 inhabitants is mainly dominated by the headquarters of companies in the production sector (Daimler, Porsche, Bosch). It was chosen because of an actual lack of office market studies and a very stable market avoiding wrong conclusions because of short-run market movements. The data for the analysis consists in a Shapefile with all 185,000 buildings of the city, an extended attribute database for the 32,000 buildings with more than 200 mÂ of floor space, an office building inventory of the field study of the consulting company Baasner / Langwald / Mller including vacancies, and finally price

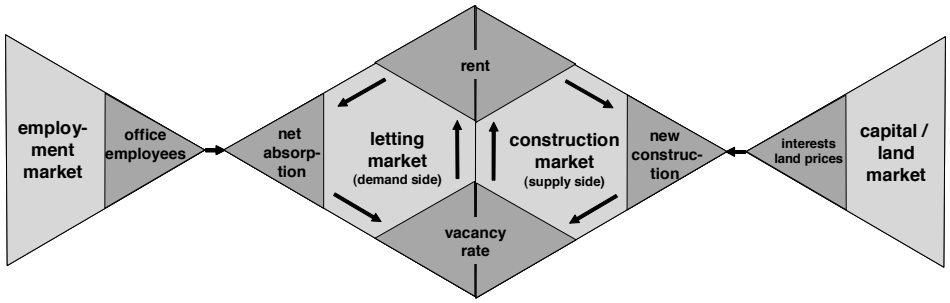


Figure 1: The interaction of the different office markets.

data of the cadastral authority and the Investment Property Databank (IPD).

3 Urban economic approaches

The classical space requirement calculation of urban planners ignores traditionally the existence of vacancy and the heterogeneity of the office space. For this reason an in- depth analysis of the local differences in office markets has to fall back on economic methods. As these are ignoring traditionally the spatial aspect, it has to be added where possible.

3.1 Longitudinal models: Simultaneous equations

The following office market model is based on works of e.g. [Ros84, DiW96, Für06]. It requires time-series data of an entire office market (in general: one city or metropolitan area). Temporal lags in the equations give the model its dynamics. The parameters of the interacting markets (the labor market on the demand side and the capital market on the supply side) are given exogenously. Figure 1 illustrates the relations between the most important variables. In the Stuttgart case the following equations were estimated as linear regression models with significant parameters (signs of the coefficients given in brackets):

(1) Rent = f(Vacancy rate of the previous period (-))

(2) Net absorption = f (number of office employees of the year before (+), rent (-))

The new vacancy rate can be calculated easily by the occupied stock and the total stock. The latter one has to be actualized periodically by the supply side development:

(3) New construction = f (Net absorption of two years before (+))

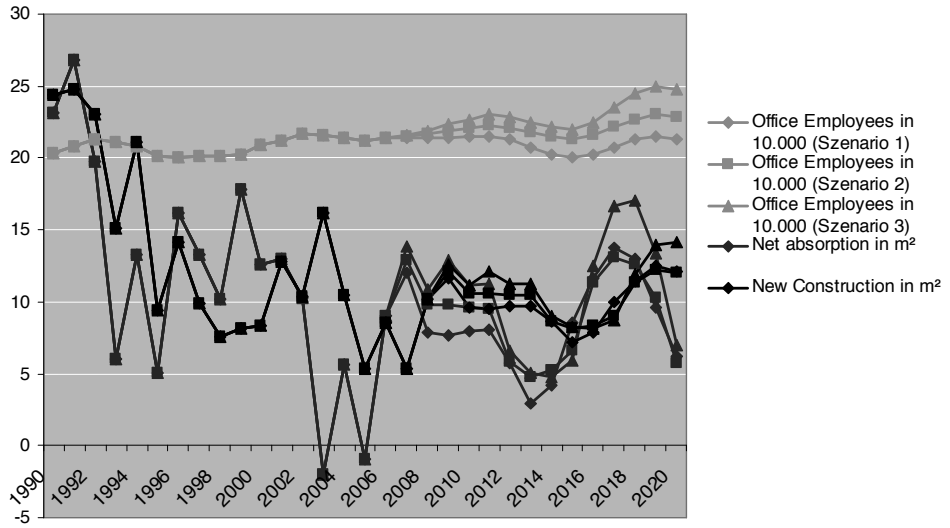


Figure 2: Forecasting changes in space demand (net absorption) and supply (new construction) for exogenously given office employment scenarios ranging from 215,000 to 250,000 employees in 2020.

The equations can then be used for the calculation of market scenarios with different values for the development of the number of employees. Figure 2 shows some results for two scenarios with a constant cycle length and a target value of 230,000 and 250,000 employees in the service sector.

The problem of the longitudinal models consists in the incapability to adapt to detailed locations. Despite the theoretical possibility of integrating flows and price elasticities between parts of a city, the approach fails due to missing long-run time-series for submarkets.

3.2 Cross-sectional models: Hedonic price modeling and the estimation of vacancy risks

One solution for analyzing market outcomes at a spatially detailed level consists in applying the hedonic price modeling approach (see [Des00, NM06]). Its general idea is the repartition of the price of a heterogeneous good on its different characteristics. Office buildings have physical characteristics (size, building quality, parking facilities etc.) and spatial characteristics (accessibility, cluster effects etc.). The spatial parameters are measured by GIS. To measure the effect of neighbouring buildings and zone characteristics each building is equipped with three buffer zones in the distance of 200, 500 and 1,000 meters (figure 4). All together, there are ca. 80 building and location variables, which are tested in all the following models. This overview paper reports only the most significant ones contributing strongly to the adjusted R^2.

An exemplary hedonic price model for office buildings in Stuttgart results in a linear regression equation with the following significant variables:

(4) Rent = f (construction year of the building (+), number of m^2 of retail space in the 200-m-buffer (+), change of the number of office occupiers in the 200-m-buffer within the last 5 years in % (+))

The function can then be used for an estimation and visualization of the rent level throughout the city. Figure 5 illustrates the outcome with a high price zone in the city center. The hedonic price modeling is a quite common approach. Despite its main use for residential appraisal in the US it can easily be adapted for office markets (with a reduction of the goodness-of-fit of the model because of the lower standardization of office buildings compared to residential ones). However, the vacancy rate is more interesting for urban planning and risk management in investment than prices. As a consequence it is useful to substitute the explained variable âRentâ by âVacancyâ. With the standard linear regression model it works only if vacancy rates for zones are explained (see [NI07]). At the building level the proposal consists in working with a binary logistic model where y=1 means the existence of vacancy in the building regardless of how many percent. In Stuttgart 15% of the buildings fit into this category of (partly) vacant buildings. The binary logistic transformation allows the calculation of probabilities for staying vacant:

$$(5) \quad p(y_i = 1) = \left(1 + e^{-(\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_n x_n)}\right)^{-1}$$

The influence factors x_1, x_2, x_3 etc. in the Stuttgart case are:

(6) Probability of vacancy = f (number of storeys (+), vacancy rate in the buffer zones (+), percentage of different construction periods in the 200-m-buffer zone, construction year (+), distance to the city center (-), distance to the nearest commuter rail station (-))

Some of the estimated coefficients β_1, β_2, β_3 etc. carry an unexpected sign (e.g. the city center and the rail variable). A possible explanation is the fact that speculatively erected buildings (with higher probability of staying vacant) cluster mainly in the city center and around rail stations.

3.3 Evaluation of the approach

The combined application of longitudinal and cross-sectional models allows a forecast at the macro level and a detailed local analysis for today. Given the influence parameters of todayâs cross-sectional models the function could also be applied for future situations (e.g. new office clusters change the input parameters). In some case it might be sufficient. In others it can be quite problematic because the significance and the coefficients of the parameters can change. For this reason an alternative approach is discussed: the microsimulation.

4 Microsimulation approach

The microsimulation in urban planning reference studies is usually carried out for grid cells and for employees in different economic sectors regardless of their real estate typology they occupy. The fact that Stuttgart has a lot less office buildings than potential grid cells lets the author omit the grid cell concept and return to a building-based approach (where additional attributes from cadastral data are available). Consequently the urban microsimulation literature [MSW02, WU03] can be used as a source of ideas but the model equations have to be adapted and estimated for the specific building-based office market problems.

The first requirement for simulations is a starting stock (which is given by the cadastral data) and a starting population. The latter one comes from a business inventory of the chamber of commerce completed by an administration list of the yellow pages. The first problem consists in locating the users in the building. A simple alphanumerical join with the address field reaches a success ratio of 93.4 neighbourhood-matching in GIS (different approaches and their effectivity are discussed in [vMSH]).

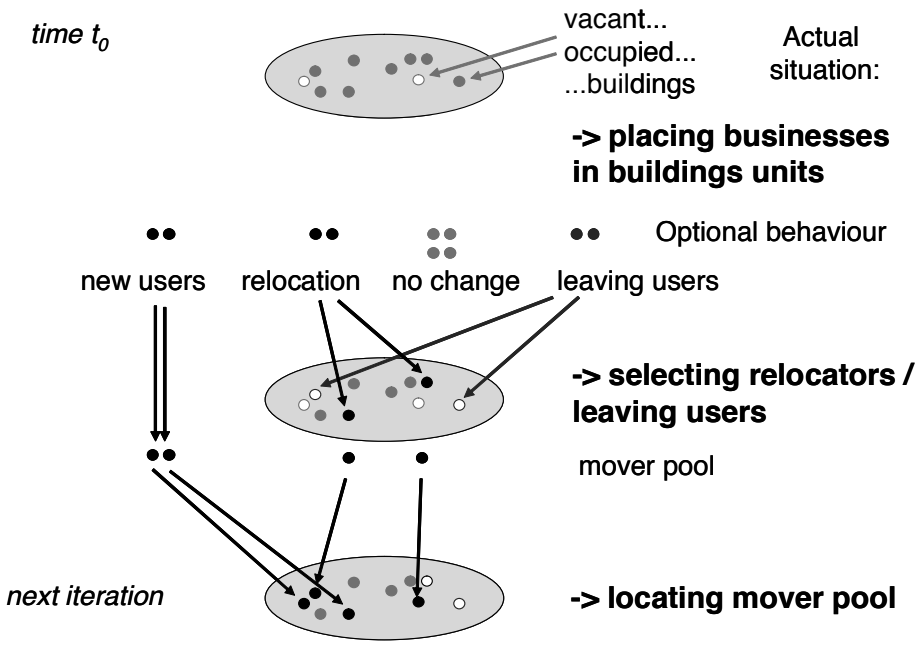


Figure 3: Microsimulation of office markets

Once the starting set of buildings and users is ready, the first iteration consists in the following steps:

1. Which user moves or leaves? To answer this question a binary logit model is estimated based on a dataset of moves collected by the chamber of commerce. The explaining variables are given by the characteristics of the user and the characteristics of the building.

   (7) p(move / leave) = f(herding behaviour: users from the same sector leaving (+), floor space ratio (+), different dummy variables for sectors and company sizes)

2. All moving users and the manually added new users are then put to a mover pool. The subsequent question is: which user chooses which location? The assignment process to the right location will be carried out in two-step-process: first, a building size restriction filters out non-suitable buildings (and the ones without vacant space). Second, different options are drawn randomly. The probability for each option is calculated by different location-choice equations which are estimated within a sample of chosen locations (drawn from the chamber of commerce mover sample) and additionally integrated non-chosen locations. As a location-choice equation can not contain user-specific and building-specific variables at the same time (due to dependence) a set of equations has to be estimated for different user types (clustered by sector and size).

The supply side will be inserted manually in a first step. For the next years the locations of assigned space for new office uses is quite determined by urban planning.

The simulation tool is actually realized in *Access* and will be migrated to *ArcView*. Figure 6 shows exemplary results with the target parameter "vacancy". It is important to remember that one simulation describes only one of millions of possible scenarios. If we are interested in forecasting e.g. the probability of vacancies multiple scenarios have to be processed and evaluated statistically. The simulation tool offers this option.

5 Outlook

The paper showed that different approaches can be used for the spatially detailed analysis of supply and demand in office markets. The microsimulation approach is not yet established in office market research. Its complexity brings with it the fact the further research has to done especially in the following fields:

- How can new construction be modeled (given the strongly regulated urban planning in Europe) and how can the refurbishment topic be integrated? How are tearing down and changing use tackled?
- The spatial autocorrelation (e.g. of the vacancy rates) and the herding behaviour have been integrated yet in a simple way. The methods of spatial econometrics could improve the models.

- How can the simulation results be visualized and communicated through an easily accessible (internet) user interface?

References

[For69] Jay Forrester. *Urban Dynamics*. Cambridge, 1969.

[Lee73] Douglass B. Lee, Jr. Requiem for large-scale models. In *Simuletter VI / 3*, 1973, pp. 16–29

[Bat92] Michael Batty. Urban Modeling in Computer-Graphic and Geographic Information System Environments. In Environment and planning / B, 19/1992, pp. 689 – 708

[WU04] Paul Waddell and Gudmundur Freyr Ulfarsson. Introduction to Urban Simulation. Design and Development of Operational Urban Models. In Stopher, Button, Kingsley, Hensher (publisher): *Handbook in Transport, Volume 5: Transport Geography and Spatial Systems*. 2004, pp. 203–236. Download at www.urbansim.org (last access September 1, 2007).

[MSW02] Ralf Moeckel, Carsten Schrmann and Michael Wegener. Microsimulation of Urban Land Use. In Proceedings of the 42nd European Conference of the ERSA, Dortmund, 2002.

[SR05] Gerhard Steinebach and Martin Rumberg. Die Umgebungslärmrichtlinie der Europäischen Union und ihre Umsetzung in deutsches Recht. In *Zeitschrift fr deutsches und internationales Bau- und Vergaberecht (ZfBR)*, 4/2005, pp. 344–348.

[Rum07] Martin Rumberg. *Raum- und akzeptorbezogene Modellierung der Umgebungslrmbelastung auf der Basis strategischer Lrmkarten â ein Ansatz zur Risiko- und Umweltqualittsbeurteilung in der Stadtplanung*. Kaiserslautern, 2007.

[HMM05] H. Hagen, M. Mnchhofen and F. Michel. Interactive Reconstruction of 3D Models for Virtual Walkthroughs. In *Proceedings of CEMVRC05*, pp. 101-105, Prag, Czech Republic, 2005

[Wie07] Ingo Wietzel. *Methodische Anforderungen an die Augmented-Reality-Technik sowie an immersive Szenarien zur Qualifizierung der Stadtplanung am Beispiel des innerstdtischen Wohnens*. Kaiserslautern, 2007.

[Ros84] Kenneth T. Rosen. Toward a Model of the Office Building Sector. In *AREUEA Journal*, 12 / 3, 1984.

[DiW96] Denise DiPasquale and William Wheaton. *Urban Economics of Real Estate Markets*. Englewood Cliffs, 1996.

[Für06] Franz Fürst. Predictable or not? Forecasting office markets with a simultaneous equation approach. In *Proceedings of the Annual Meeting of the European Real Estate Society (ERES)*. Weimar, Germany, 9 June, 2006.

[Des00] Jake Desyllas. *The relationship between urban street configuration and office rent patterns in Berlin*. London, 2000. Download at: www.intelligentspace.com (last access September 1, 2007).

[NM06] Ingrid Nappi-Chaulet and Tristan-Pierre Maury. A spatiotemporal Autoregressive Price Index for the Paris Office Property Market. In *Proceedings of the Annual Meeting of the European Real Estate Society (ERES)* Weimar, Germany, June 9, 2006.

[NI07] Vanessa Nadalin Gapriotti and Danilo Igliori. The determinants of Vacancy Rates in the Sao Paulo Metropolitan Area: A spatial Approach. In *Proceedings of the European Regional Science Association (ERSA) Conference*. Cergy-Pontoise, 2007.

[WU03] Paul Waddell and Gudmundur Freyr Ulfarsson. *Accessibility and Agglomeration: Discrete-Choice Models of Employment Location by Industry Sector*. Annual Meeting of the Transportation Research Board. Washington, D.C., Jan. 2003. Download at www.urbansim.org (last access September 1, 2007).

[vMSH] C. v. Malottki, G. Steinebach and H. Hagen. *The Visualization of Urban Planning Data – Connecting Alphanumerical Address Data and Building Geometries*. Forthcoming.

[vMal] C. v. Malottki. *Geomodellierung in Stadtplanung und Immobilienwirtschaft – am Beispiel des Broflchenmarktes Stuttgart.* Kaiserslautern, forthcoming.

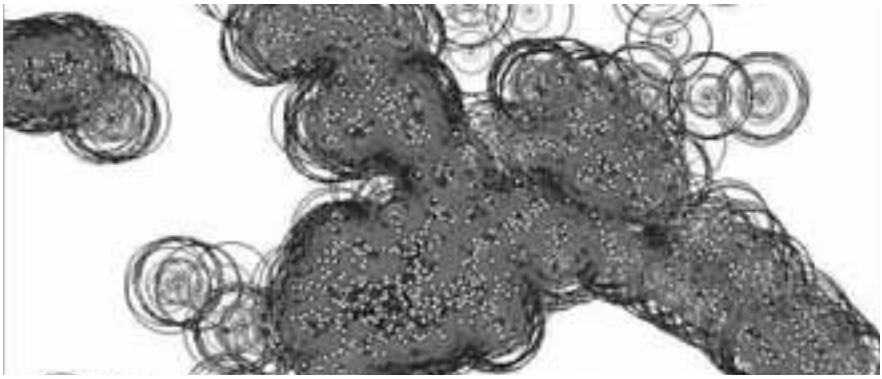


Figure 4: Methodology: The geographic influence factors are raised by influence zones around each building in the sample.

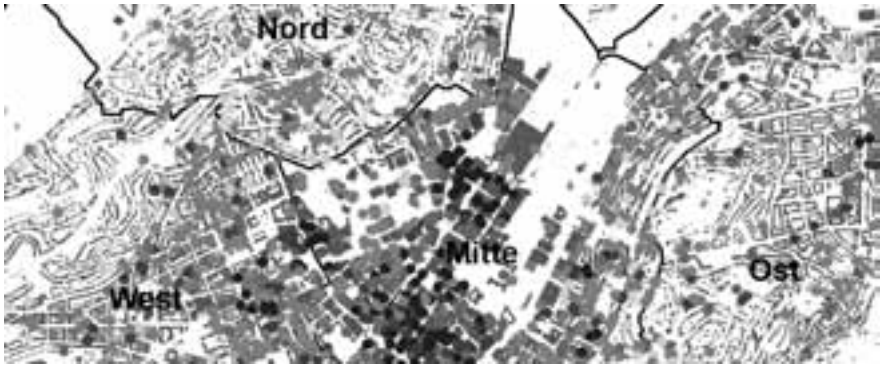


Figure 5: Forecasting price levels for all office buildings in Stuttgart (green = low price level, blue = high price level)

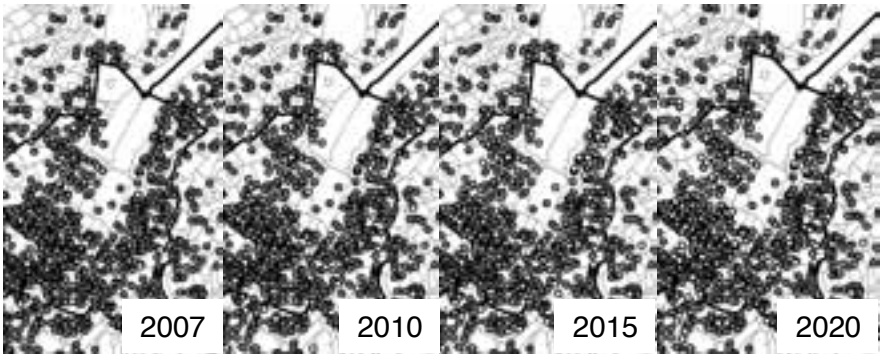


Figure 6: Exemplary simulation results for a leaving scenario. The colors indicate the vacancy rates of the buildings.

Why interval arithmetic is so useful

Younis Hijazi[1], Hans Hagen[1], Charles Hansen[2], and Kenneth I. Joy[3]

[1] University of Kaiserslautern & IRTG 1131
Department of Computer Sciences
D-67653 Kaiserslautern, Germany
http://www-hagen.informatik.uni-kl.de

[2] SCI Institute
University of Utah
http://www.cs.utah.edu/ hansen/

[3] Institute for Data Analysis and Visualization
Computer Science Department
University of California, Davis
http://graphics.cs.ucdavis.edu/ joy/

Abstract: *Interval arithmetic* was introduced by Ramon Moore [Moo66] in the 1960s as an approach to bound rounding errors in mathematical computation. The theory of interval analysis emerged considering the computation of both the exact solution and the error term as a single entity, i.e. the interval. Though a simple idea, it is a very powerful technique with numerous applications in mathematics, computer science, and engineering. In this survey we discuss the basic concepts of interval arithmetic and some of its extensions, and review successful applications of this theory in particular in computer science.

1 Introduction

In this paper we survey interval arithmetic—a very powerful technique for controlling errors in computations—and some of its extensions in the context of self-validated arithmetics. An immediate first-order extension of interval arithmetic (IA) is known as affine arithmetic (AA) and many algorithms using interval techniques compare these two approaches; in practice there is a trade-off between accuracy and speed. Moreover, there are a lot more interval-based alternatives to IA or AA such as Taylor-based arithmetics [Neu03] or extensions of AA that are numerically more stable (e.g. Messine [Mes02]) that we will briefly review in this paper. Our central interest is to show how these interval techniques can be used in practical applications in computer science to provide robust algorithms without necessarily sacrifying speed, as interactivity or real-time are often desired especially in computer graphics and visualization.

2 Interval arithmetic

2.1 A brief history

This section is largely inspired by G. W. Walster's article *Introduction to Interval Arithmetic* [Wal97], as it perfectly introduces how IA emerged.

Ramon E. Moore conceived interval arithmetic in 1957, while an employee of Lockheed Missiles and Space Co. Inc., as an approach to bound rounding errors in mathematical computation. Forty years later, at April 19, 1997 kick-off meeting of Sun Microsystems' interval arithmetic university R & D program, he explained his thinking as follows: in 1957 he was considering how scientists and engineers represent measurements and computed results as $\tilde{x} \pm \epsilon$, where $\tilde{x}$ is the measurement (or result) and ϵ is the error tolerance.

While representing fallible values using the $\tilde{x} \pm \epsilon$ notation is convenient, computing with them is not, even in a case as simple as calculating the area of a room. If the errors due to finite precision arithmetic are simultaneously taken into account, complexity increases further. Error analyses of large scientific, engineering and commercial algorithms are sufficiently complex and labor intensive that they are often not conducted. The result is that machine computing with floating-point arithmetic is not tightly linked to mathematics, science, commerce or engineering.

Moore had a better idea. He reasoned that since $\tilde{x} \pm \epsilon$ consists of two numbers, $\tilde{x}$ and ϵ, why not use two different numbers to represent exactly the same information? That is, instead of $\tilde{x} \pm \epsilon$, use $\tilde{x} - \epsilon$ and $\tilde{x} + \epsilon$, which define the endpoints of an interval containing the exact quantity in question, i.e. x. It was this simple, yet profound, idea that started interval arithmetic and interval analysis, the branch of applied mathematics developed to numerically analyze interval algorithms.

One of the most famous references on IA is probably Moore's *Interval Analysis* book [Moo66] but there are also several more recent surveys introducing IA, e.g. [CMN02, Wal97].

2.2 What is interval arithmetic?

In the same way classical arithmetic operates on real numbers, interval arithmetic defines a set of operations on intervals. We denote an interval as $\overline{\underline{x}} = [\underline{x}, \overline{x}]$, and the base arithmetic operations are as follows:

$$\begin{aligned}
&\overline{\underline{x}} + \overline{\underline{y}} = [\underline{x} + \underline{y}, \overline{x} + \overline{y}], \\
&\overline{\underline{x}} - \overline{\underline{y}} = [\underline{x} - \overline{y}, \overline{x} - \underline{y}], \\
&\overline{\underline{x}} \times \overline{\underline{y}} = [min(\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y}), max(\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y})].
\end{aligned}$$

Other operations such as division can be devised in a similar fashion; in the case of division by an interval containing zero, special care is needed, the same way as for real-number floating point arithmetic.

Three properties of intervals and interval arithmetic make it possible to precisely link the

fallible observations of science and engineering to mathematics and floating-point arithmetic ([Wal97]):

1. Any contiguous set of real numbers (a continuum) can be represented by a containing interval.
2. Intervals provide a convenient and mechanical way to represent and compute guaranteed error bounds using fallible data.
3. All the important properties of infinite precision interval arithmetic can be preserved using finite precision numbers and directed rounding, which is commonly available on most computers (indeed, any machines supporting the IEEE 754 floating-point standard).

2.3 Why interval arithmetic?

There are usually three sources of error while performing numerical computations: rounding, truncation and input errors. In the following examples (taken from [CMN02]) we show how IA is meant to keep track of them.

- *Rounding errors*:
  Consider the expression $f(x) = 1 - x + \frac{x^2}{2}$ with $x = 0.531$, i.e. with 10^{-3} precision. Computing this expression with classical arithmetic gives the result $f(x) = 0.610$. Now, if we perform the computations using IA, we get $f(x) = 0.469 + \frac{0.531^2}{2} \in 0.469 + \frac{[0.281, 0.282]^2}{2}$ and so $f(x) \in 0.469 + [0.140, 0.141] = [0.609, 0.610]$. This guarantees that the exact result is within the interval $[0.609, 0.610]$.
- *Truncation errors*:
  We are now interested in a Taylor series of the exponential function: $e^x = 1 + x + \frac{x^2}{2!}e^t$ where $t \in [0, x]$. For $x < 0$, $e^x \in 1 + x + \frac{x^2}{2!}[0, 1]$. In particular, with $x = -0.531$, we get $e^{-0.531} \in 1 - 0.531 + \frac{(-0.531)^2}{2!}[0, 1] = 0.469 + [0.140, 0.141][0, 1] = [0.469, 0.610]$. This example illustrates how IA keeps track of both, the rounding and the truncation errors.
- *Input errors*:
  Suppose that due to data uncertainty our input is $x \in [-0.532, -0.531]$. If we evaluate the previous expression we obtain $e^x \in 1 + [-0.532, -0.531] + \frac{[-0.532, -0.531]^2}{2}[0, 1] = [0.468, 0.470] + \frac{[0.280, 0.284]^2}{2}[0, 1] = [0.468, 0.470] + [0, 0.142] = [0.468, 0.612]$. This final example illustrates how IA can keep track of all error types simultaneously.

Moreover IA does not suffer from any restriction to a particular class of functions that it can be applied to. Indeed Moore's fundamental theorem of interval arithmetic [Moo66] states that for any function f defined by an arithmetical expression, the corresponding interval evaluation function F is an *inclusion function* of f:

$$F(\underline{x}) \supseteq f(\underline{x}) \;=\; \{f(x) \;\mid\; x \in \underline{x}\}.$$

Given an implicit function f and a n-dimensional bounding box B defined as a product of n intervals, we have a very simple and reliable rejection test for the box B not intersecting the image of the function f (e.g. surface or volume),

$$0 \notin F(B) \Rightarrow 0 \notin f(B).$$

As shown in Figure 1(a), "point sampling" fails as a robust rejection test on non-monotonic intervals. While many methods exist for isolating monotonic regions or approximating the solution, inclusion methods using interval or affine arithmetic are among the most robust and general.

The inclusion property can be used in ray tracing or mesh extraction for identifying and skipping empty regions of space. Note that although $0 \notin F(B)$ guarantees the absence of a root on an interval B, the converse does not necessarily hold: one can have $0 \in F(B)$ without B intersecting f. When $F(B)$ loosely bounds the convex hull, as in Figure 1(b), IA makes for a poor (though still reliable) rejection test. This overestimation problem is a well-known disadvantage, and is fatal for algorithms relying on iterative evaluation of non-diminishing intervals.

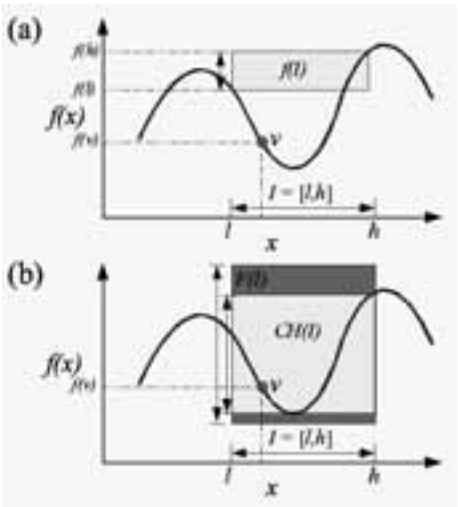


Figure 1: *Inclusion property of interval arithmetic.* (a) Floating point arithmetic is insufficient to guarantee a convex hull over the range. (b) IA is much more robust by encompassing all minima and maxima of the function within that interval. Ideally, $F(I)$ is equal or close to the bounds of the convex hull, $CH(I)$. Here the box B is simply the interval I.

Fortunately, the overestimation error is proportional to domain interval width; therefore IA guarantees (linear) convergence to the correct solution when interval domains diminish. This is the case in algorithms such as sweeping computation of hierarchically subdivided domains [Duf92, HB07], and ray tracing algorithms involving recursive interval bisection [Mit90, CHMS00, KHW$^+$07] as we will see in Section 5. Though the overestimation problem affects the efficiency of these algorithms, recursive IA methods robustly detect

the zeros of a function, given an adequate termination criterion such as a sufficiently small precision ϵ over the domain, or tolerance δ over the range.

Effectively, it suffices to implement a library of these IA operators, and substitute them for the real operators, producing an *interval extension* F. If each component operator preserves the inclusion property, then arbitrary compositions of these operators will as well. As a result, literally any computable function may be expressed as interval arithmetic. Some operations are ill-defined, such as empty-set or infinite intervals. However, these are easily handled in a similar fashion to real-number floating point arithmetic.

In literature we often read "inclusion algebra" or "self-validated arithmetic" when referring to IA and the IA extension is often referred to as the *natural inclusion function*, but it is neither the only mechanism for defining an inclusion algebra, nor always the best. Particularly in the case of multiplication, it greatly overestimates the actual bounds of the range. To overcome this, it is necessary to represent intervals with higher-order approximations.

3 Extensions of interval arithmetic

In this section we discuss extensions of IA. We here focus on first-order arithmetics that can be alternatives to IA for example when higher accuracy in the computations is needed.

3.1 Affine arithmetic

Affine arithmetic was developed by Comba & Stolfi [CS93] to address the bound overestimation problem of IA. Intuitively, if IA approximates the convex hull of f with a bounding box, AA employs a piecewise first-order bounding polygon, such as a parallelogram (see Figure 2).

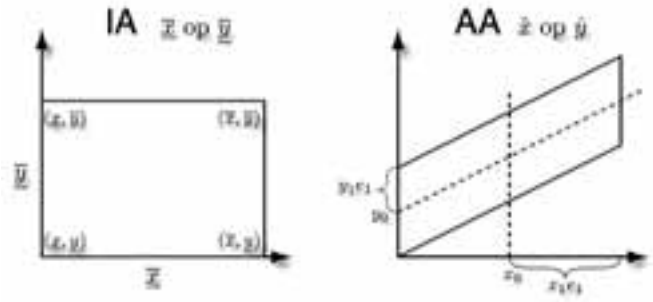


Figure 2: *Bounding box of IA versus AA.*

An affine quantity $\hat{x}$ takes the form

$$\hat{x} = x_0 + \sum_{i=1}^{n} x_i e_i$$

where the $x_i, \forall i \geq 1$, are the *partial deviations* of $\hat{x}$, and $e_i \in [-1, 1]$ are the *error symbols*.

Base affine operations in AA are as follows (where c is a real constant):

$$\mathbf{c} \times \hat{x} = \mathbf{c}x_0 + \mathbf{c}\sum_{i=1}^{n} x_i e_i,$$
$$\mathbf{c} \pm \hat{x} = (\mathbf{c} \pm x_0) + \sum_{i=1}^{n} x_i e_i,$$
$$\hat{x} \pm \hat{y} = (x_0 \pm y_0) \pm \sum_{i=1}^{n} (x_i \pm y_i) e_i.$$

However, *non-affine* operations in AA cause an additional error symbol e_z to be introduced. This is the case in multiplication between two affine forms,

$$\hat{x} \times \hat{y} = x_0 y_0 + \sum_{i=1}^{n} (x_i y_0 + y_i x_0) e_i + rad(\hat{x}) rad(\hat{y}).$$

Other operations in AA, such as square root and cosine, approximate the range of the IA operation using a first-derivative regression curve. These also compute a new error symbol.

The chief improvement in AA comes from maintaining correlated error symbols as orthogonal entities. This effectively allows error among correlated symbols to diminish, as opposed to always increasing monotonically in IA. Figure 3(b) shows how AA be much more accurate than IA (by providing tighter bounds) e.g. in curve approximation.

Conversion between IA and AA An affine form is created from an interval as follows:

$$x_0 = (\overline{x} + \underline{x})/2,$$
$$x_1 = (\overline{x} - \underline{x})/2,$$
$$x_i = 0, \;\; i > 1.$$

and can equally be converted into an interval $\overline{\underline{x}} = [x_0 - rad(\hat{x}), x_0 + rad(\hat{x})]$ where the *radius* of the affine form is given as

$$rad(\hat{x}) = \sum_{i=1}^{n} |x_i|.$$

Affine arithmetic is only one example of first-order interval-based approximation and this topic is still an active research area that concentrates on finding a trade-off between accuracy and computational cost. Other popular first-order arithmetics are E. R. Hansen's generalized interval arithmetic [Han75] and its centered form variant [Neu03], and first-order Taylor arithmetic [Neu03]. All these methods provide over-approximations; when combined with under-approximations, they can prove useful e.g. in static analysis [GP07].

3.2 Extended affine arithmetic

There are many possible extensions of IA or AA; here we will illustrate only one. In the following example we present the so-called AF1 formulation by Messine [Mes02] in which, for some constant n, a truncated affine form is given as

$$\hat{x} = x_0 + \sum_{i=1}^{n} x_i e_i + x_{n+1} e_{n+1}.$$

The arithmetic operations are given by

$$\mathbf{c} \pm \hat{x} = (\mathbf{c} \pm x_0) + \sum_{i=1}^{n} x_i e_i + |x_{n+1}| e_{n+1},$$

$$\hat{x} \pm \hat{y} = (x_0 \pm y_0) + \sum_{i=1}^{n} (x_i \pm y_i) e_i + (x_{n+1} + y_{n+1}) e_{n+1},$$

$$\mathbf{c} \times \hat{x} = (\mathbf{c} x_0) + \sum_{i=1}^{n} \mathbf{c} x_i e_i + |c x_{n+1}| e_{n+1},$$

$$\hat{x} \times \hat{y} = (x_0 y_0) + \sum_{i=1}^{n} (x_o y_i + y_0 x_i) e_i + (|x_0 y_{n+1}| + |y_0 x_{n+1}| + rad(\hat{A}) rad(\hat{B})) e_{n+1}.$$

In practice we noticed that this formulation is far more efficient than pure AA, providing decent results even when using $n = 1$ or $n = 2$. Messine also developed an extension of AF1 denoted by AF2; for details see [Mes02].

Previous introduced interval-based approaches were all zero- or first-order. Note that there are also higher-order methods such as the quadratic form (QF) by Messine. [Mes02] (whose formulation becomes more and more complicated) or Taylor-based arithmetics (see Gavriliu's PhD thesis [Gav05]).

4 Discussion on IA-based techniques

After introducing these different interval algebras we might wonder which one should be used in practice. In this section we aim at providing answers to this question (based on experiments) and provide pointers to existing IA-like implementations.

In practice we noticed that despite providing less accurate results, IA is much more robust than AA and, in particular, has better numerical stability. As previously mentioned IA is especially bad when computing interval products. In this case, simple AA can achieve double the performance for the same cost. But for all the other operations the difference is not that important. An interesting approach might be having a hybrid IA/AA technique which would always use the optimal one. Indeed we often have to make a trade-off in practice between accuracy and efficiency. In short, IA has a linear convergence—which is considered slow in most applications—but is robust whereas AA algebras have a better accuracy at a more important computational cost and less stability than IA.

There are many IA-like libraries available and the most popular ones are written in C++, e.g. Filib++ [LTG+06] for interval arithmetic and LibAffa [GCH00] for affine arithmetic. Those libraries are convenient for experimenting with IA—as one basically needs to replace `float` or `double` by `interval`—but they can be slow as carrying a lot of unnecessary operations for the desired application; in this case, one might prefer to implement our own library, e.g. as for the interactive ray tracing algorithm using IA [KHW+07]. If completeness is required there are several references which provide a complete IA, e.g. [Kul08]. Also, if function derivatives are needed in a particular application, IA can be combined with automatic differentiation (AD) and thus evaluate the interval function and its interval derivative at the same time.

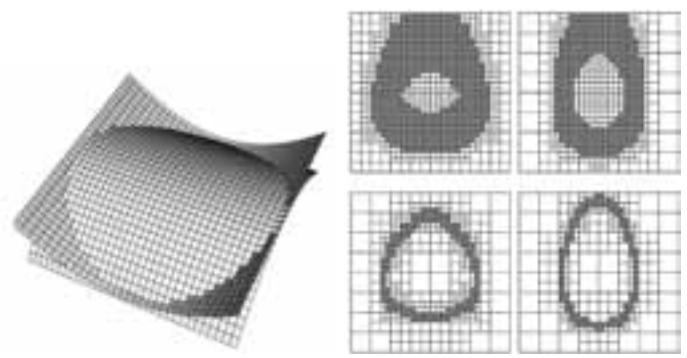
5 Successful applications in computer science

IA-based algorithms have been used successfully to address many problems. Here we review four selected applications which are pattern recognition, computational geometry, mesh extraction and finally ray tracing.

5.1 Pattern recognition and computational geometry

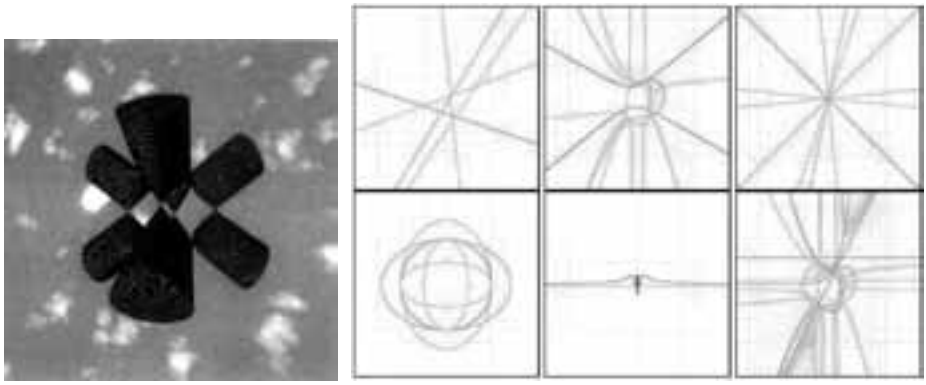
In computer vision, exploration of arrangements by subdivision methods has been used for computing globally optimal solutions to geometric matching problems under bounded error using interval arithmetic as a key ingredient in the algorithm [Bre03].

In computational geometry, e.g. for computing the intersection of curves or surfaces, IA has been employed for robustly finding those roots. In [dF96] de Figueiredo applies AA for robustly intersecting parametric surfaces 3(a). He uses a quadtree decomposition of the domain for the output. He also compares IA and AA in performance 3(b) and comes to the conclusion that AA is better suited for this particular task.



(a) (b)

Figure 3: [dF96]: (a) Surface intersection using AA. (b) IA (top) versus AA (bottom).



(a) (b)

Figure 4: (a) [Duf92]: CSG ray tracing using IA. (b) [HB07]: Arrangement of curves using IA.

Other computational geometry examples using interval arithmetic-based recursive subdivisions are CSG operations for implicit surfaces [Duf92] (see Figure 4(a)) and arrangements of implicit curves [HB07] (see Figure 4(b)). In their paper, Hijazi et al. provide a method for computing arrangements of implicitly defined curves. The new method for computing arrangements is an adaptation of methods successfully used for the exploration of large, higher dimensional, non-algebraic arrangements in computer vision. While broadly similar to subdivision methods in computational geometry, its design and philosophy are different; for example, it replaces exact computations by subdivision and interval arithmetic computations and prefers data-independent subdivisions. It can be used (and is usually used in practice) to compute well-defined approximations of arrangements, but can also yield exact answers for specific problem classes. For a brief survey on arrangement of curves see [Hij06].

5.2 Mesh extraction

Lopes et al. [LJdF01] present an algorithm for computing a robust adaptive polygonal approximation of an implicit curve in the plane. The approximation is adapted to the geometry of the curve as the length of the edges varies with the curvature of the curve (see Figure 5). Robustness is achieved by combining interval arithmetic and automatic differentiation.

This work has been extended to robust surface approximation by Paiva et al. [PLLdF06], leading to a dual marching-cube octree-based algorithm using IA and providing topological guarantees (given a certain precision). Figure 6 illustrates this concept: green regions are topologically guaranteed whereas red regions are uncertain; in this illustration, the precision is pixel-based.

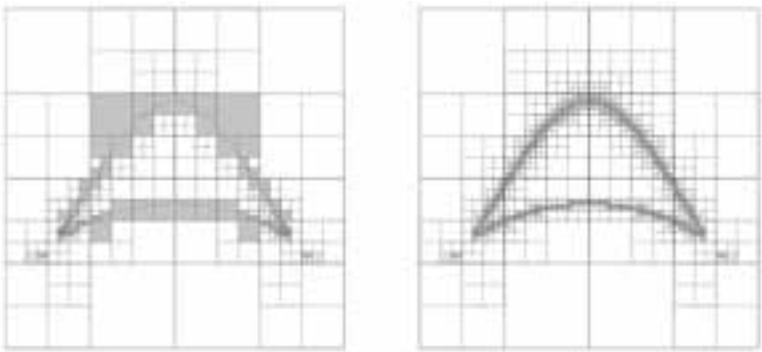
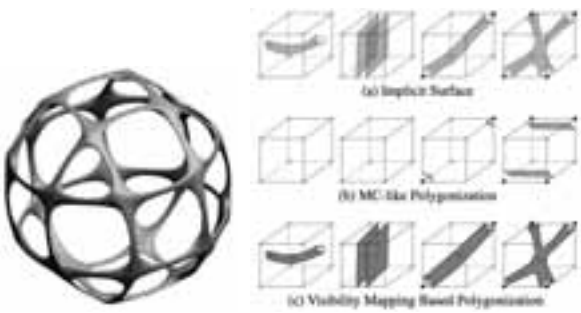


Figure 5: *[LJdF01]: Bicorn curve approximation using IA.*

Figure 6: *[PLLdF06]: Linked tori approximation using IA.*

Another marching-cube like algorithm relying on IA and providing topological guarantees was proposed by Varadhan et al. [VKZM06]. A decocube obtained with this technique is shown in Figure 7(a). On top of interval arithmetic, the method uses a visibility map and dual contouring for extracting the mesh and compares the results with classical marching cubes (MC) in Figure 7(b).



(a) (b)

Figure 7: [VKZM06]: (a) Decocube. (b) Marching cubes.

5.3 Ray tracing

5.3.1 A brief overview

There is a large number of IA-based ray tracers and we only review some of them. The first ones who introduced IA techniques for ray tracing were Toth [JB86] and Mitchell [Mit90]. In his survey [Mit91] Mitchell applied Moore's algorithm (see Figure 8(a)) for root-finding using IA in the context of ray tracing. Indeed, ray tracing often reduces to a root-finding problem [Mit91]. This concept is illustrated by Figure 8(b) showing how ray tracing implicit surfaces reduces to solving a one-dimensional root-finding problem.

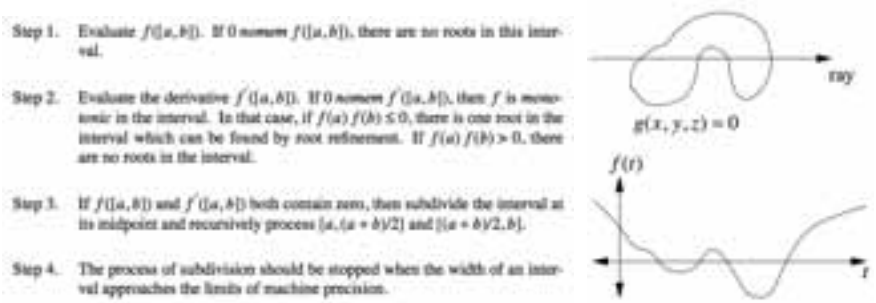


Figure 8: [Mit91]: (a) Moore's root-finding algorithm. (b) Ray tracing and root-finding.

Since the 90s many other publications appeared in this area. Capriani et al. [CHMS00] combined interval bisection with various other iterative schemes, including the Interval Newton method. De Cusatis Junior et al. [dCJdFG99] used affine arithmetic to address the bound overestimation problem of pure interval arithmetic.

Sanjuan-Estrada et al. [SECG03] compared performance of two hybrid interval methods with implementations of the Interval Newton and a recursive point-sampling subdivision method in the POV-Ray framework. Figure 9 shows some results of the so-called MRFro (Minimal Root Finder reorder) algorithm (see [SECG03]). Heidrich & Seidel [HS98] used AA for ray tracing procedural displacement shaders.

5.3.2 Recent results

Recently, Gamito & Maddock [GM07] applied reduced affine arithmetic for ray casting implicit fractal surfaces. Though efficient, the proposed reduced affine arithmetic (RAA) method only preserves inclusion under specific circumstances and can only be applied to a certain class of functions. Figure 10 shows a procedural planet modeled using their technique.

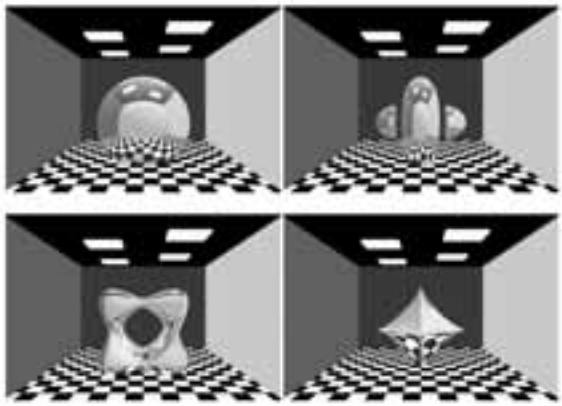
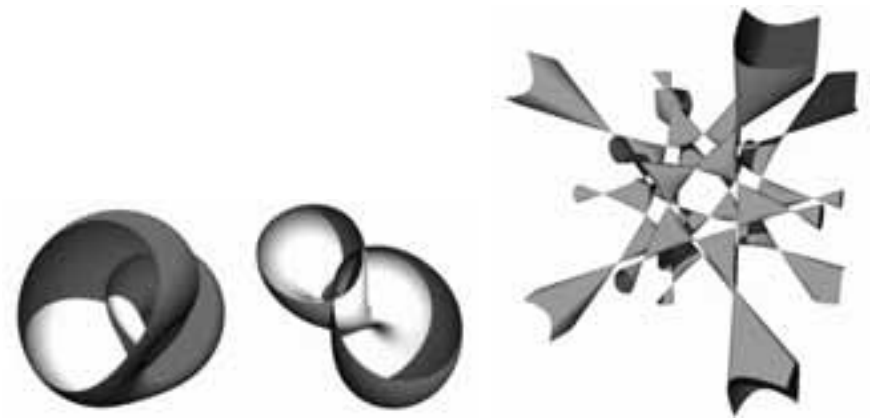


Figure 9: *[SECG03]: Up-left to down-right: sphere, Mitchell, tangle and super-ellipsoid.*

Figure 10: *[GM07]: Procedural modeling using RAA.*

Knoll & Hijazi et al. [KHW+07] applied interval arithmetic for interactive ray tracing of arbitrary implicit functions. This is the first time robustness and interactivity have been jointly achieved for ray tracing implicits. Figures 11(a) and 11(b) show selected implicits ray-traced using Knoll & Hijazi et al.'s technique.

In their paper the authors present an efficient algorithm for interactive ray tracing of arbitrary implicit surfaces where IA is used for both robust root computation and guaranteed



(a) (b)

Figure 11: [KHW$^+$07]: (a) Klein bottle. (b) Barth-sextic implicit.

detection of topological features. In conjunction with ray tracing, this allows for rendering literally any programmable implicit function simply from its definition. The proposed method requires neither special hardware, nor preprocessing or storage of any data structure. Efficiency is achieved through SIMD optimization of both the interval arithmetic computation and coherent ray traversal algorithm, delivering interactive results even for complex implicit functions.

Because they neither pre-compute an explicit representation of the object, nor a physical acceleration structure in memory, they have great flexibility in rendering dynamically changing N-dimensional implicits. Figure 12 demonstrates an animated $4D$ implicit (morphing between a two-sheeted hyperboloid and a torus).

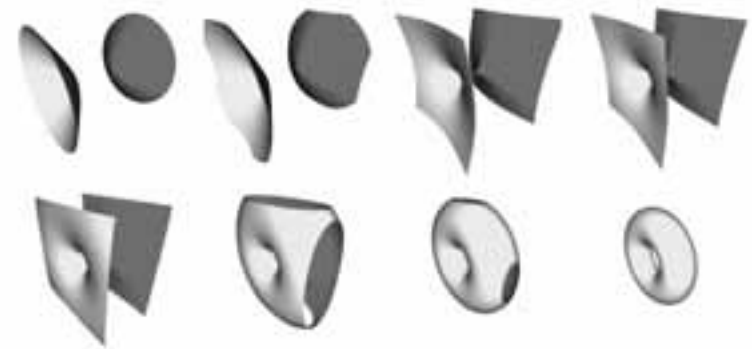


Figure 12: *[KHW$^+$07]: Morphing performed "on the fly".*

6 Conclusion

As we tried to demonstrate in this brief survey, interval-based techniques can be very useful in computer science, e.g. in computer vision, computational geometry, mesh extraction or ray-tracing. Ten years ago, the first ray-tracing algorithms using IA as a key ingredient appeared and took advantage of IA's inherent robustness and adaptivity by applying it to concrete problems. Though robust, many of those algorithms suffered from a lack of speed and thus implying little interest in the graphics community. More recently, with the increasing computational power of CPUs and GPUs, interval techniques are gaining attention as being now able to provide both fast and robust algorithms. We hope that the computer science community, especially in computer graphics and visualization, will from now on tend to increasingly include IA techniques in their algorithms.

Acknowledgements This work was supported by the German Science Foundation (DFG IRTG 1131) as part of the International Research Training Group on *Visualization of Large and Unstructured Data Sets—Applications in Geospatial Planning, Modeling, and Engineering*.

References

[Bre03] T. M. Breuel. On the Use of Interval Arithmetic in Geometric Branch-and-Bound Algorithms. *Pattern Recognition Letters*, 24(9-10):1375–1384, 2003.

[CHMS00] O. Capriani, L. Hvidegaard, M. Mortensen, and T. Schneider. Robust and Efficient Ray Intersection of Implicit Surfaces. *Reliable Computing*, 6:9–21, 2000.

[CMN02] O. Caprani, K. Madsen, and H. B. Nielsen. Introduction to Interval Analysis. *Informatics and Mathematical Modelling*, Technical University of Denmark, Nov 2002.

[CS93] J. L. D. Comba and J. Stolfi. Affine arithmetic and its applications to computer graphics. In *Proc. SIBGRAPI'93 â VI Brazilian Symposium on Computer Graphics and Image Processing, 9â18*, 1993.

[dCJdFG99] A. de Cusatis Junior, L. de Figueiredo, and M. Gattas. Interval Methods for Raycasting Implicit Surfaces with Affine Arithmetic. In *Proceedings of XII SIBGRPHI*, pages 1–7, 1999.

[dF96] L. H. de Figueiredo. Surface intersection using affine arithmetic. In *Graphics Interface*, pages 168–175, May 1996.

[Duf92] T. Duff. Interval arithmetic and recursive subdivision for implicit functions and constructive solid geometry. In *SIGGRAPH '92: Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, pages 131–138, New York, NY, USA, 1992. ACM Press.

[Gav05] M. Gavriliu. *Towards more efficient interval analysis corner forms and a remainder interval Newton method*. PhD thesis, California Institute of Technology, Pasadena, California, 2005.

[GCH00] O. Gay, D. Coeurjolly, and N. J. Hurst. Libaffa—C++ Affine Arithmetic Library for GNU/Linux, 2000.

[GM07] M. N. Gamito and S. C. Maddock. Ray casting implicit fractal surfaces with reduced affine arithmetic. *Vis. Comput.*, 23(3):155–165, 2007.

[GP07] E. Goubault and S. Putot. Under-approximations of computations in real numbers based on generalized affine arithmetic. In *Proceedings of SAS'07*, Copenhagen, August 2007.

[Han75] E. R. Hansen. A Generalized Interval Arithmetic. In *Proceedings of the International Symposium on Interval Mathemantics*, pages 7–18, London, UK, 1975. Springer-Verlag.

[HB07] Y. O. Hijazi and T. M. Breuel. Computing Arrangements using Subdivision and Interval Arithmetic. In *P. Chenin, T. Lyche and L.L. Schumaker (Eds.), Proceedings of the Sixth International Conference on Curves and Surfaces, June 29-July 5, 2006, Avignon, France, Curve and Surface Design: Avignon 2006, Nashboro Press*, pages 173–182, 2007.

[Hij06] Y. O. Hijazi. Arrangements of Planar Curves. In *H. Hagen, A. Kerren, P. Dannenmann (Eds.), Visualization of Large and Unstructured Data Sets, Proceedings of the first workshop of DFG's International Research Training Group "Visualization of Large and Unstructured Data Sets—Applications in Geospatial Planning, Modeling, and Engineering", June 14-16, 2006, Dagstuhl, Germany, GI-Edition, Lecture Notes in Informatics, Seminars Series*, volume S-4, pages 59–68, 2006.

[HS98] W. Heidrich and H.-P. Seidel. Ray-Tracing Procedural Displacement Shaders. In *Graphics Interface*, pages 8–16, 1998.

[JB86] K. I. Joy and M. N. Bhetanabhotla. Ray tracing parametric surface patches utilizing numerical techniques and ray coherence. In *SIGGRAPH '86: Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pages 279–285, New York, NY, USA, 1986. ACM Press.

[KHW+07] A. Knoll, Y. Hijazi, C. Hansen, I. Wald, and H. Hagen. Interactive Ray Tracing of Arbitrary Implicits with SIMD Interval Arithmetic. In *Proceedings of the 2nd IEEE/EG Symposium on Interactive Ray Tracing*, pages 11–18, 2007.

[Kul08] U. Kulisch. Complete Interval Arithmetic and its Implementation. In *Numerical Validation in Current Hardware Architectures*, Dagstuhl Seminar Proceedings, Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany, 2008.

[LJdF01] H. Lopes, O. Jo, and L. H. de Figueiredo. Robust Adaptive Approximation of Implicit Curves. In *SIBGRAPI '01: Proceedings of the XIV Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'01)*, page 10, Washington, DC, USA, 2001. IEEE Computer Society.

[LTG+06] M. Lerch, G. Tischler, J. Wolff Von Gudenberg, W. Hofschuster, and W. Krämer. FILIB++, a fast interval library supporting containment computations. *ACM Trans. Math. Softw.*, 32(2):299–324, 2006.

[Mes02] F. Messine. Extentions of Affine Arithmetic: Application to Unconstrained Global Optimization. *Journal of Universal Computer Science*, 8(11):992–1015, 2002.

[Mit90] D. P. Mitchell. Robust Ray Intersection with Interval Arithmetic. In *Proceedings on Graphics Interface 1990*, pages 68–74, 1990.

[Mit91] D. P. Mitchell. Three applications of interval analysis in computer graphics. Course Notes for Frontiers in Rendering, Siggraph '91, 1991.

[Moo66] R. E. Moore. *Interval Analysis*. Prentice Hall, Englewood Cliffs, NJ, 1966.

[Neu03] A. Neumaier. Taylor Forms-Use and Limits. *Reliable Computing*, 9:43–79(37), February 2003.

[PLLdF06] A. Paiva, H. Lopes, T. Lewiner, and L. H. de Figueiredo. Robust adaptive meshes for implicit surfaces. In *19th Brazilian Symposium on Computer Graphics and Image Processing*, pages 205–212, 2006.

[SECG03] J. F. Sanjuan-Estrada, L. G. Casado, and I. Garcia. Reliable algorithms for ray intersection in computer graphics based on interval arithmetic. In *XVI Brazilian Symposium on Computer Graphics and Image Processing, 2003. SIBGRAPI 2003.*, pages 35–42, 2003.

[VKZM06] G. Varadhan, S. Krishnan, L. Zhang, and D. Manocha. Reliable implicit surface polygonization using visibility mapping. In *SGP '06: Proceedings of the fourth Eurographics symposium on Geometry processing*, pages 211–221, Aire-la-Ville, Switzerland, Switzerland, 2006. Eurographics Association.

[Wal97] G. W. Walster. Introduction to Interval Arithmetic. unpublished note supporting Sun Microsystems compilers for interval programs, May 19, 1997.

A Survey of Implicit Surface Rendering Methods, and a Proposal for a Common Sampling Framework

Aaron Knoll
SCI Institute, University of Utah
IRTG, University of Kaiserslautern

Abstract: We consider several applications of implicit surfaces in visualization, and methods for rendering them. In particular we focus on geometry processing techniques for mesh extraction; and ray casting methods for direct rendering of implicits. Given that both methods rely on sampling the implicit function in question, we design a software framework that could accomodate both algorithms. We conclude by evaluating the time complexity and performance of existing systems, and discuss the long-term potential of both methods for rendering and computational goals.

1 Introduction

An *implicit* is a multivariate function over an $\mathbf{R}^N$ domain. In 3-space, it is given by a function

$$f : \mathbf{R}^3 \rightarrow \mathbf{R}$$

such that, for an isovalue v, the implicit surface, or isosurface, is defined where $f(x, y, z) = v$, or

$$f(x, y, z) - v = 0$$

This formulation defines a level set, specifically a level surface.

In scientific computing and graphics, implicits are frequently employed as reconstruction kernels for filtering discrete data. As such, implicits are the *lingua franca* of data analysis, at least for data that does not lend itself to an immediate explicit visualization modality.

To render implicits in 3D, one is principally given three choices:

- *Extract explicit polygonal (triangle mesh) geometry, and rasterize that trivially on graphics hardware.*
- *Re-sample the implicit into some other proxy geometry, such as points or slices of a volume, and render that proxy geometry.*

- *Ray-trace the implicit directly.*

This paper will survey methods for rendering implicits. Noting that these rendering systems share common algorithmic components, we propose a common framework architecture for sampling implicits and comparing performance. Finally, we compare the implicit sampling performance of several existing systems, and make predictions concerning the long-term suitability of these various methods to various applications.

2 Applications of Implicits

Implicits are used across scientific disciplines such as mathematics, physics, biology and engineering. As computer scientists, our goal is to determine which visualizations of implicits are most meaningful and practical for their specific application. At the core of this issue lies the question of *what* the implicit itself is modeling.

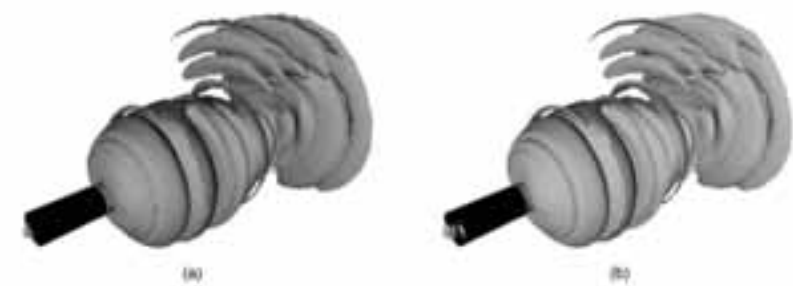


Figure 1: *Marching cubes (a), compared to ray-traced 13th-order Legendre polynomials approximating the computed implicits from the FE reference domain (b). From Nelson & Kirby [NK05].*

In some scenarios, the implicit function itself models natural or physical phenomena, and data is computed to approximate it discretely. This is the case for much volume data in simulation. Theoretically, it is desirable to visualize the same implicit employed in actual computation, as opposed to some arbitrary interpolant of the resulting data. As an example, for first-order finite elements, piecewise-linear planar interpolants of a tet mesh accurately represent the computed implicit surface. For higher-order spectral finite elements, however, this is not the case, and it is desirable to visualize higher-order implicit surfaces over the computational domain, or approximations thereof [NK05] (Figure 1).

For structured data, correct visualization is less clear. Finite differences computations approximate the differential operator and not the solution itself, thus no single implicit is strictly "correct" in visualizing them. Instead, we choose a convolution filter, the simplest being a first-order Lagrangian trilinear interpolant. This yields piecewise-smooth isosurfaces; but as the choice of filter is arbitrary, rendering it correctly is a dubious proposition (Figure 2). Scanned medical data, such as MRI and reconstructed CT scans, pose a similar conundrum of which filter is "best". This has partially been addressed [ML94, MMMY97] for volume rendering.

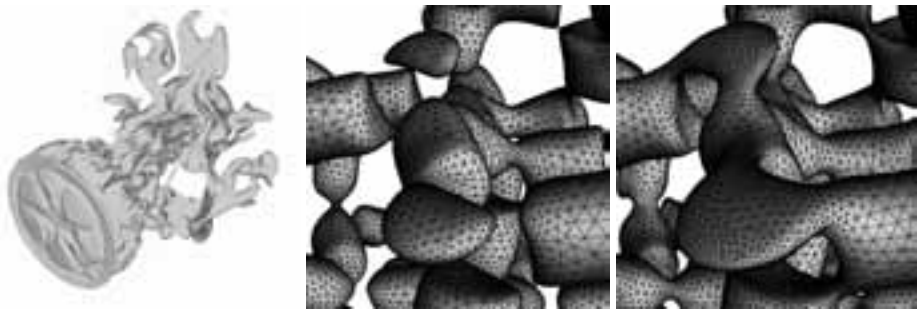


Figure 2: *Left to right: a finite-differences CD simulation ray-traced using a piecewise C^0 trilinear interpolant filter, and centrally differenced gradient normals [KHW07]; Advancing front extraction of structured data using piecewise-C^1 Catmull-Rom spline interpolants; and the same technique using piecewise-C^2 B-spline filtering [SS06]. Aesthetics aside, it is questionable which filter, if any, is "correct".*

In the case of raw point sets, an implicit is used to interpolate data points from a scanner, allowing us to render a smooth surface. Here, the choice of implicit is secondary compared to how well it reproduces features (particularly high-curvature regions, and singularities) on the surface, and how efficient it is to render [CBC$^+$01, OBA$^+$03, AA03].

Figure 3: *Piecewise-smooth partition-of-unity implicit reconstruction of raw point data, reconstructed with high polygonization, coarse polygonization, and sphere traced (e.g. Hart [Har96]). From Ohtake et al. [OBA$^+$03].*

Finally, the implicit function itself can be of interest, for example when rendering superquadrics or blobby objects [Bli82], or graphing arbitrary algebraic [KB89] and even non-algebraic [Mit90, PLLdF06] expressions.

3 Rendering Implicits

In general, four options exist for rendering implicits in 3-space: direct volume rendering; extraction and rasterization; ray casting or ray tracing; and point-based rendering. Volume rendering and point-based methods are proxy methods. In practice these work quite well, due to efficient mapping on graphics hardware. However, they can vary widely in actual implementation, depending on desired fidelity and type of data. Partly for this reason we shall focus primarily on extraction and ray casting, as they represent algorithmic extremes that are either designed to accurately construct a mesh, or render a view-dependent scene.

3.1 Volume Rendering

Direct volume rendering, samples the scalar field (usually regularly) and convolves these samples via a transfer function. When a transfer function is sufficiently sharp, or singular, at a desired value, volume rendering can approximate a surface. However, volume rendering is computationally demanding, and only interactive when implemented efficiently on graphics hardware. For rendering discrete isosurfaces, a near-singular transfer function is imperfect, particularly for large, entropic, data. In practice however, volume rendering of isosurfaces is useful, and worth mentioning as a feasible alternative to methods that render surfaces directly. An efficient implementation combining GPU volume rendering with isosurface rendering is given in [HSS+05].

3.2 Mesh Extraction

For rendering implicit surfaces, the most common method is to extract a mesh from the given data, and rasterize that mesh trivially on Z-buffer graphics hardware. The best-known method for surface extraction is Marching Cubes (MC), developed independently by Cleary & Wyvill [WMW86] and Lorensen & Cline [LC87]. The general idea in extraction is to subdivide the scalar field into convex "cell" primitives, and solve where the isosurface intersects cell boundaries. In conventional marching cubes, cells are voxels of the volume, and the isosurface is approximated by solving where linear interpolation yields the desired isovalue along a voxel edge. By considering the values at vertices of the cell, the algorithm knows which edges will contain a surface; moreover for a trilinear interpolant implicit only 15 different configurations exist for piecewise-linear iso-polygons within a given voxel. As voxels are C^0 at their edges, the resulting mesh is guaranteed to be continuous. More generally, this method allows a mesh to approximate any implicit, as solving for arbitrary $f(x, y, z)$ reduces to a 1D problem when any two coordinate axes are constant.

Mesh Extraction Variants Marching cubes is popular due to its simplicity and ease of implementation. However, it possesses several limitations. Some MC configurations cause ambiguities in orientation that must be resolved to preserve continuity, but this has been addressed [NH91]. Fundamentally, marching cubes is limited in that it samples regularly over a uniform grid. While this is adequate for small structured volumes, it is ill-suited for irregular point sets and unstructured data. Fortunately, a similar algorithm, Marching Tetrahedra (MT), was developed by Doi & Koide [DK91] to operate natively on unstructured tet meshes, though it can equally be applied to structured data due to the dual relationship between tetrahedra and voxels. Though MT addresses some sampling issues of marching cubes with unstructured data, both algorithms generate poor-quality, irregular meshes, and fail to capture topological features such as singularities. Dual contouring [JLSW02] and dual marching cubes [Nie04, SW04] alleviate these problems, reducing overall triangle counts, and effectively capturing singularities. However, "thin" features below a geometric subdivision criteria can still be lost using these techniques, as evident

from Paiva et al. [PLLdF06], and meshes generating using dual marching cubes still suffer from irregularity (Figure 4). For generating high-quality triangle meshes, Scheidegger et al. [SFS05] proposed an advancing front meshing algorithm using a curvature-based guidance field. Schreiner et al. [SS06] extended this method to structured and unstructured scalar volume fields, and modified the guidance field to minimize both geometric error and number of required samples.

Figure 4: *Left: a robust meshing algorithm employing interval arithmetic and dual marching cubes fails to determine the correct connectivity of a surface, likely due to insufficiently fine geometric sampling [PLLdF06]. Right: ray tracing using a similar interval arithmetic technique can also experience difficulty sampling thin features, but samples in a view-dependent fashion and thus better recovers features for closer views [KHH+06].*

Dynamic Extraction and Visualization For larger data, it becomes necessary to manage overall scene complexity, measured in the number of cells or tetrahedra visited by the extraction algorithm. As long as this is controlled, it is generally trivial to render the resulting mesh interactively on graphics hardware. To reduce the visited set of cells, Wilhelms & Van Gelder [WV92] proposed a min/max hierarchy embedded in a branch-on-need octree for trivially ignoring empty regions of a volume. Livnat et al. refined the interval tree concept in creating a span space tree [LSJ96]. Livnat & Hansen [LH98] extended the min-max octree technique with view-dependent frustum culling using a shearing transformation visibility test. Westermann et al. [WKE99] implemented adaptive marching cubes on multiresolution octree data. Pesco et al. [PLPS04] proposed an occlusion test for the implicit itself, further reducing the cost of isosurface extraction at render-time. With few exceptions these techniques enable interactive rendering of data up 512^3. For larger data, interactive visualization is difficult, and extraction techniques often rely on topologically-guided simplification [LMM06] into Morse-Smale complexes. Though this does not render the entire original data set, topological simplification and segmentation is arguably more useful for data analysis.

3.3 Ray Casting

While extraction methods focus on sampling the full scalar field, ray casting samples the implicit directly at the intersection point of a viewing ray and the implicit surface. For

rendering large data, this potentially entails far fewer samples than an adequate extraction method, and "better" samples in the sense that we sample exactly (and only) the regions required to fill a 2D uniform grid in screen space. Thus, ray tracing performs both view frustum and object-occlusion culling automatically. With an efficient acceleration structure, it performs spatial subdivision as well and renders objects in $O(logN)$ time as opposed to $O(N)$. For large volume data, this is extremely important. In structured volumes, the number of voxels multiplies eight-fold every time dimensions double; large data can easily overwhelm an extraction/rasterization method, even one with comprehensive subdivision and occlusion culling algorithms. The main drawback of ray tracing is that, particularly when not optimized, it can be slow. However, the inherent scalability of ray tracing, combined with increasing power of multicore CPU's and flexibility of GPU's, suggests the algorithm will increase in feasibility and popularity.

General implicit ray tracing In ray tracing, all primitives are at some level formulated implicitly, including the plane equation of a triangle. In general, ray tracing solves the implicit function for a parameter t along each ray. This is accomplished by substituting $P_i = O_i + tD_i$ for $i = \{X, Y, Z\}$. Then, $f(x, y, z)$ becomes a one-dimensional function $f_t(t)$, and finally $f_t(t) = v$ may be solved for t. Thus, ray tracing implicits is a root-finding problem; it solves for the first root along a given interval. Simple implicits of second degree and lower can be solved analytically. Blinn's blobbies [Bli82] were the first application of ray tracing higher order or rational implicit surfaces. Kalra & Barr [KB89] devised a general method for ray tracing algebraic surfaces with known Lipschitz-condition bounds for the gradient and curvature. The key issue with ray tracing arbitrary implicits is that globally convergent numerical methods fail to handle discontinuous or non-monotonic functions. One solution is to evaluate intersection by finding and minimizing a signed distance function for the implicit surface, as proposed by Hart [Har96]. Yet more general is the use of an inclusion algebra such as interval arithmetic [Mit90] or affine arithmetic [dCJdFG99]. General methods for implicit ray tracing have historically proven slow, but recent work [KHH+06] using coherent optimizations suggests that pure interval bisection is surprisingly practical and interactive. However, this method has not been thoroughly tested on fitting and filtering point and scalar data, and it should be assumed to be slower than optimized special-case intersection for most third-degree and lower implicits. Higher-order and rational implicits, however, can be rendered efficiently using interval ray tracing methods.

Structured data isosurface ray tracing Parker et al. [PSL+98] were the first to ray trace isosurfaces from structured volumes, employing these piecewise implicit interpolating patches. Though a small supercomputer was necessary to render them at the time, the method proved powerful in the size of volume data it could render at full resolution. The scalability of this method to large data was further demonstrated by DeMarle et al. [DPH+03] on clusters, rendering the $2048x2048x1920$ Richtmyer Meshkov CFD data set in its entirety using a software distributed shared memory layer. More recently, Knoll et al. [KWPH06] implemented a lossless octree compression scheme for structured volumes, allowing both volume data and acceleration structure to be compactly represented

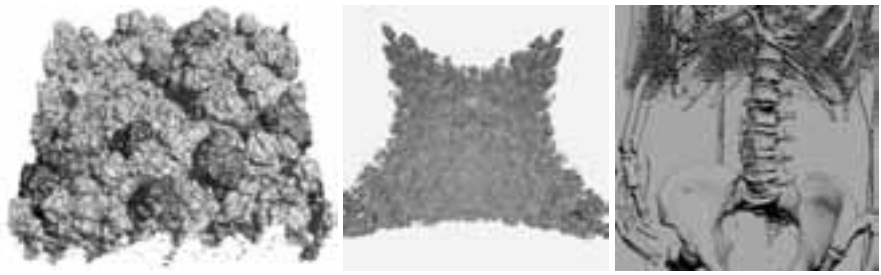


Figure 5: *Left: Morse-Smale topological segementation and simplification of large data [LMM06]. Center:* $2048x2048x1920$ *Richtmyer-Meshkov instability field, losslessly compressed from 7.5 GB to 1.8 GB and rendered at multiple fps on a multicore laptop [KWPH06]. Right: hybrid extraction and point-based rendering system, rendering the* $512x512x1024$ *visible female dataset interactively [LT04].*

in less than 25% the original volume footprint. In conjunction with an efficient ray-octree traversal algorithm, this enables rendering the Richtmyer Meshkov data on a laptop with 2 GB RAM at multiple frames per second, and interactively on a 16-core workstation.

Unstructured and point-set ray tracing Ray tracing is equally suited to visualizing implicit point-set surfaces using radial basis functions, as demonstrated by Wald et al. [WS05]. Large unstructured data has also recently been investigated [WFKH07]. With coherent BVH traversal and an optimized ray-isopolygon intersection similar to marching tetrahedra, CPU ray tracing is surprisingly competitive with GPU shader-based extraction techniques [WKME03, BCCS07], even on modest multicore desktop hardware.

3.4 Point-Based Rendering

Though CPU ray tracing tackles large data easily, its overall performance on moderate-size data is still underwhelming compared to GPU rendering methods. It is worth mentioning the contributions of point-based rendering methods on the GPU, conceived by Levoy & Whitted [LW05] and proven for large point-set data by Rusinkiewicz et al.[RL00] with QSplat. Point-based rendering is spiritually similar to ray tracing in that acceleration structure traversal and primitive intersection are computed out-of-core on the CPU, but shading and actual rasterization occur on the GPU. Co et al. [CHJ03] extended the isosurface ray tracing technique of Parker et al. to point-based rendering on the GPU. Livnat & Tricoche [LT04] took this system one step further, implementing a hybrid isosurface render that combined extraction and point-based methods, and handled reasonably large data interactively. Zhou et al. [ZG06] develop a system for rendering higher-order finite element volumes very efficiently, though it requires pre-segmenting data into point sets.

4 A Unified Framework for Extraction and Ray Tracing

Rasterization and ray tracing ultimately both sample the input data in world space, though their means of sample generation and processing differ. It would not be difficult to share a common spatial subdivision and data sampling framework between an extraction and ray tracing application. At the lower algorithmic level, ray tracing and extraction methods can vary (between themselves, as well as each other) in actual implementation, largely in whether they iteratively evaluate or analytically solve roots on the implicit to generate the sample.

4.1 Framework Overview

The general idea of our framework is to abstract data acquisition, subdivision structure generation, and implicit expression generation, which are used by multiple common extraction and ray tracing implementations. In short, our pipeline is designed to handle two common-case ray tracing and extraction systems with varying degrees of generality; and an unlimited number of fixed-function techniques for ray tracing or rasterization that can use other components as necessary. We require three inputs from the user: data; an implicit to filter that data; and a sampler (either a ray tracing or extraction method). After processing these inputs, the system outputs either a mesh or a ray-traced image, as shown in Figure 6.

4.2 Pairing Ray Tracing and Extraction

In the previous sections 3.2 and 3.3, we have discussed common techniques for extracting and ray tracing implicits. We note that the bounded spectral radius for the minimal guidance field in advancing front extraction [SS06] is similar to the geometric bounds of the L-G surfaces in Kalra & Barr [KB89]. Both techniques assume the filter implicit to be twice differentiable (or at least have constant gradient or curvature) with known bounds on gradient and curvature (Jacobian and Hessian, respectively). Hence, they could logically be paired together as techniques for C^2 algebraic implicits.

For rendering of arbitrary non-algebraic implicit functions, the interval arithmetic method proposed by Paiva et al. [PLLdF06] is roughly analagous to ray casting using interval bisection [Mit90, KHH$^+$06]. Extraction methods using this technique would require a spatial subdivision structure such as an octree or BVH over which to evaluate intervals, followed by application of dual marching cubes or a similar algorithm over these regions.

The term "subdivision structure" in Figure 6 is somewhat vague; particularly in the context of rendering functions as in [KHH$^+$06] and [PLLdF06]. Both techniques involve sampling within the implicit function domain, not merely the data domain. However, in Paiva et al. [PLLdF06] samples are constructed explicitly from dual marching cubes over this domain, whereas Knoll et al. [KHH$^+$06] recommend no explicit spatial subdivision,

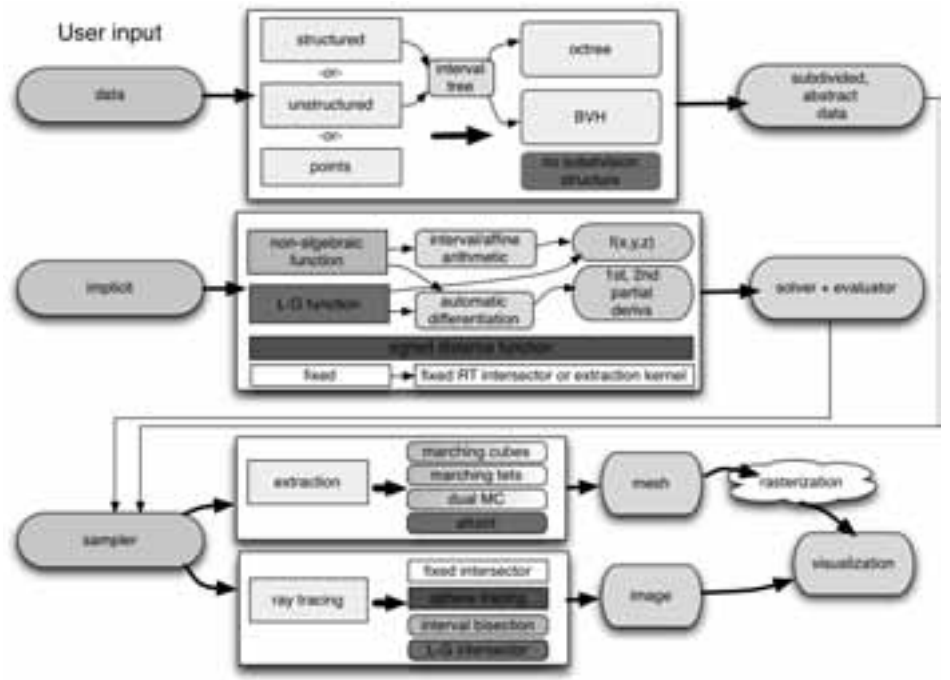


Figure 6: *Proposed framework for extraction and ray-tracing.*

and instead sample the implicit on-the-fly during intersection.

4.3 Fixed-Function Components

We recommend a fixed workflow for certain types of implicits, mostly for comparison purposes. For example, one could implement standard marching cubes using trilinear interpolation over voxels, or optimized ray-trilinear patch intersection using the Marmitt et al. [MFK+04] technique. Fixed functionality need not be absolute throughout the system; the Marmitt intersector could be used in conjunction with either a BVH or an octree, for example. Another practical option, used inherently by point-based implicits such as MPU [OBA+03] and sphere tracing [Har96] would be allowing the user to specify a signed distance equation as opposed to the implicit formulation itself.

4.4 Expected Quality and Performance

Using this framework, it would be fairly simple to benchmark the number of samples collected by ray tracing and mesh extraction techniques for similar views. These numbers would be somewhat unfair unless the rasterization system performed all the appropriate

culling optimizations [WV92, LSJ96, LH98, WKE99, PLPS04]. However, we can make some general assumptions about anticipated behavior. Assuming the implicit is an accurate representation of our data (and more specifically, what we seek to visualize), we expect the quality of the image or mesh to be dependent on the number of samples, and where these samples are spent.

With ray casting, samples are created from a camera by ray generation. For visualization purposes, it is hard to go wrong with this technique, as samples are spent exactly where the user wishes to see. Even when a sample misses the surface, it is still meaningful (and usually cheaper to compute). One could argue that this purely view-dependent sampling strategy is flawed when the projected object feature frequency exceeds the viewing ray frequency, causing aliasing. However, this could be solved either by supersampling rays or downsampling the object (via level-of-detail methods) over high object-frequency regions – both well-known techniques in computer graphics.

In extraction, mesh construction works best when sampling adapts to the curvature of the surface – specifically when regions of high curvature are allocated more samples. When the user only requires one mesh as output, it makes sense to generate the best mesh possible, and employ an adaptive and topologically sensitive method such as advancing front. For dynamic visualization purposes, however, the quality of the mesh is generally secondary to the extraction time; naive marching cubes (perhaps with some view-dependent adaptivity) is likely preferable in this case.

Empirically, existing literature suggests it is more difficult to extract a good mesh than to ray trace a good image. For example, Schreiner et al. [SS06] report the small 256^3 Aneurism data requiring 7 seconds to extract 134K triangles using conventional marching cubes, and over 5 minutes to extract 462K triangles using advancing front. On a single-core 2.4 GHz Opteron CPU, using the Marmitt trilinear patch intersection method, Wald et al. [WFM$^+$05] ray traced an isosurface of this data in over 6 fps at $640x480$. Assuming conservatively that only 1/20 of the samples hit the surface, and linear scalability on on dual-core of an Opteron 2.2 GHz, ray casting processes 112K samples per second, whereas marching cubes generates 10K samples per second, and advancing front around 720 samples per second. The number of samples per second processed in point-based GPU methods are easily two orders of magnitude even than ray casting, though point set data must be precomputed in advance [ZG06]. For higher-resolution data, ray casting should perform similarly well due to its logarithmic acceleration structure traversal complexity. Advancing front should fare better compared to marching cubes, due to its adaptivity. While these comparisons are extremely rough and not completely fair, they are consistent with the findings of Livnat & Tricoche [LT04], in comparing their point-based isosurface rendering technique to full extraction. In their hybrid method, they find that view-dependent extraction works well for moderately near views, but for closeup or far away views point-based methods are more effective. Overall the suggestion is not that ray casting or point-based methods are superior to extraction, but that efficient, adequate and view-dependent sampling is necessary for visualization.

5 Conclusion

Comparing ray tracing and extraction is an apples-and-oranges affair. The decision on which to perform should depend purely on the application. When the isosurface is ultimately used for modeling, for example with rigid-body mechanical simulations, mesh extraction is the most appropriate solution. For rendering animated blobby shapes in a computer game that can be deformed dynamically by in-game physics, extracting a mesh is all but necessary; for general-purpose rendering, rasterization is unlikely to be replaced anytime soon, if ever. When mesh generation is acceptable as an offline process, it is desirable to generate the cleanest mesh possible. For this reason, and their topological and geometric adaptivity, methods such as dual marching cubes and advancing front are more useful than marching cubes, particularly for larger data. For the same reason of effective sample use, ray casting and point-based methods are better suited to dynamic visualization than on-the-fly rasterization and extraction.

References

[AA03] Anders Adamson and Marc Alexa. Ray Tracing Point Set Surfaces. In *SMI '03: Proceedings of the Shape Modeling International 2003*, page 272, 2003.

[BCCS07] F. F. Bernardon, S. P. Callahan, J. L. D. Comba, and C. T. Silva. An Adaptive Framework for Visualizing Unstructured Grids with Time-Varying Scalar Fields. *Parallel Computing*, 2007. to appear.

[Bli82] James Blinn. A Generalization of Algebraic Surface Drawing. *ACM Transactions on Graphics*, 1(3):235–256, July 1982.

[CBC+01] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. Reconstruction and representation of 3D objects with radial basis functions. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 67–76, New York, NY, USA, 2001. ACM Press.

[CHJ03] C. S. Co, B. Hamann, and K. I. Joy. Iso-splatting: A Point-based Alternative to Isosurface Visualization. In J. Rokne, W. Wang, and R. Klein, editors, *Proceedings of Pacific Graphics 2003*, pages 325–334, October 8–10 2003.

[dCJdFG99] A. de Cusatis Junior, L. de Figueiredo, and M. Gattas. Interval Methods for Raycasting Implicit Surfaces with Affine Arithmetic. In *Proceedings of XII SIBGRPHI*, pages 1–7, 1999.

[DK91] Akio Doi and Akoi Koide. An efficient method of triangulating equi-valued surfaces by using tetrahedral cells. *IEICE Trans Commun. Elec. Inf. Syst*, E-74(1):213–224, 1991.

[DPH+03] David DeMarle, Steve Parker, Mark Hartner, Christiaan Gribble, and Charles Hansen. Distributed Interactive Ray Tracing for Large Volume Visualization. In *Proceedings of the IEEE PVG*, pages 87–94, 2003.

[Har96] J. C. Hart. Sphere tracing: A geometric method for the antialiased ray tracing of implicit surfaces. *The Visual Computer*, 12(10):527–545, 1996.

[HSS+05] Markus Hadwiger, Christian Sigg, Henning Scharsach, Khatja Bühler, and Markus Gross. Real-Time Ray-Casting and Advanced Shading of Discrete Isosurfaces. *Computer Graphics Forum*, 24(3):303–312, 2005.

[JLSW02] Tao Ju, Frank Losasso, Scott Schaefer, and Joe Warren. Dual contouring of hermite data. *ACM Trans. Graph.*, 21(3):339–346, 2002.

[KB89] D. Kalra and A. H. Barr. Guaranteed ray intersections with implicit surfaces. In *SIGGRAPH '89: Proceedings of the 16th annual conference on Computer graphics and interactive techniques*, pages 297–306, New York, NY, USA, 1989. ACM Press.

[KHH+06] Aaron Knoll, Younis Hijazi, Charles Hansen, Ingo Wald, and Hans Hagen. Interactive Ray Tracing of Arbitrary Implicit Functions. Technical Report UUSCI-2007-002, SCI Institute, University of Utah, 2006.

[KHW07] Aaron Knoll, Charles Hansen, and Ingo Wald. Coherent Multiresolution Isosurface Ray Tracing. Technical Report UUSCI-07-001, University of Utah, School of Computing, 2007. (submitted for publication).

[KWPH06] Aaron Knoll, Ingo Wald, Steven G Parker, and Charles D Hansen. Interactive Isosurface Ray Tracing of Large Octree Volumes. In *Proceedings of the 2006 IEEE Symposium on Interactive Ray Tracing*, pages 115–124, 2006.

[LC87] William E. Lorensen and Harvey E. Cline. Marching Cubes: A High Resolution 3D Surface Construction Algorithm. *Computer Graphics (Proceedings of ACM SIGGRAPH)*, 21(4):163–169, 1987.

[LH98] Yarden Livnat and Charles D. Hansen. View Dependent Isosurface Extraction. In *Proceedings of IEEE Visualization '98*, pages 175–180. IEEE Computer Society, October 1998.

[LMM06] D. Laney, A. Mascarenhas, and P. Miller. Understanding the Structure of the Turbulent Mixing Layer in Hydrodynamic Instabilities. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1053–1060, 2006. Member-P. -T. Bremer and Member-V. Pascucci.

[LSJ96] Yarden Livnat, Han-Wei Shen, and Christopher R. Johnson. A Near Optimal Isosurface Extraction Algorithm Using the Span Space. *IEEE Transactions on Visualization and Computer Graphics*, 2(1):73–84, 1996.

[LT04] Yarden Livnat and Xavier Tricoche. Interactive Point Based Isosurface Extraction. In *Proceedings of IEEE Visualization 2004*, pages 457–464, 2004.

[LW05] Mark Levoy and Turner Whitted. The use of points as display primitives. Technical report, CS Department, University of North Carolina at Chapel Hill, 2005.

[MFK+04] Gerd Marmitt, Heiko Friedrich, Andreas Kleer, Ingo Wald, and Philipp Slusallek. Fast and Accurate Ray-Voxel Intersection Techniques for Iso-Surface Ray Tracing. In *Proceedings of Vision, Modeling, and Visualization (VMV)*, pages 429–435, 2004.

[Mit90] Don Mitchell. Robust Ray Intersection with Interval Arithmetic. In *Proceedings on Graphics Interface 1990*, pages 68–74, 1990.

[ML94] Stephen R. Marschner and Richard J. Lobb. An evaluation of reconstruction filters for volume rendering. In *Proceedings of Visualization '94*, pages 100–107. IEEE Computer Society Press, 1994.

[MMMY97] Torsten M oller, Raghu Machiraju, Klaus Mueller, and Roni Yagel. Evaluation and Design of Filters Using a Taylor Series Expansion. *IEEE Transactions on Visualization and Computer Graphics*, 3(2):184–199, 1997.

[NH91] Greg Nielson and Bernd Hamann. The Asymptotic Decider: Removing the Ambiguity in Marching Cubes. In G. Nielson and L. Rosenblum, editors, *Proceedings of Visualization '91*, pages 83–91. IEEE Computer Society Press, 1991.

[Nie04] Gregory M. Nielson. Dual Marching Cubes. In *VIS '04: Proceedings of the conference on Visualization '04*, pages 489–496, Washington, DC, USA, 2004. IEEE Computer Society.

[NK05] Blake Nelson and Robert M. Kirby. Ray-Tracing Polymorphic Multi-Domain Spectral/hp Elements for Isosurface Rendering. *IEEE Transactions on Visualization and Computer Graphics (Proceedings IEEE Visualization 2005)*, 12(1):114–125, 2005.

[OBA+03] Yutaka Ohtake, Alexander Belyaev, Marc Alexa, Greg Turk, and Hans-Peter Seidel. Multi-level partition of unity implicits. *ACM Trans. Graph.*, 22(3):463–470, 2003.

[PLLdF06] Afonso Paiva, HâÂlio Lopes, Thomas Lewiner, and Luiz Henrique de Figueiredo. Robust adaptive meshes for implicit surfaces. In *19th Brazilian Symposium on Computer Graphics and Image Processing*, pages 205–212, 2006.

[PLPS04] Sinesio Pesco, Peter Lindstrom, Valerio Pascucci, and Claudio T. Silva. Implicit Occluders. In *IEEE/SIGGRAPH Symposium on Volume Visualization*, pages 47–54, 2004.

[PSL+98] Steven Parker, Peter Shirley, Yarden Livnat, Charles Hansen, and Peter-Pike Sloan. Interactive Ray Tracing for Isosurface Rendering. In *IEEE Visualization '98*, pages 233–238, October 1998.

[RL00] Szymon Rusinkiewicz and Marc Levoy. QSplat: A Multiresolution Point Rendering System for Large Meshes. In *Proc. of ACM SIGGRAPH*, pages 343–352, 2000.

[SFS05] Carlos Scheidegger, Sachar Fleishman, and Claudio Silva. Triangulating Point-Set Surfaces With Bounded Error. In *Proc. of 3rd Eurographics/ACM Symposium on Geometry Processing*, pages 63–72, 2005.

[SS06] John Schreiner and Carlos Scheidegger. High-Quality Extraction of Isosurfaces from Regular and Irregular Grids. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1205–1212, 2006. Member-Claudio Silva.

[SW04] Scott Schaefer and Joe Warren. Dual Marching Cubes: Primal Contouring of Dual Grids. In *Proceedings of Pacific Graphics*, pages 70–76, 2004.

[WFKH07] Ingo Wald, Heiko Friedrich, Aaron Knoll, and Charles D Hansen. Interactive Isosurface Ray Tracing of Time-Varying Tetrahedral Volumes. Technical Report UUSCI-2007-003, SCI Institute, University of Utah, 2007. (submitted for publication).

[WFM+05] Ingo Wald, Heiko Friedrich, Gerd Marmitt, Philipp Slusallek, and Hans-Peter Seidel. Faster Isosurface Ray Tracing using Implicit KD-Trees. *IEEE Transactions on Visualization and Computer Graphics*, 11(5):562–573, 2005.

[WKE99] Rüdiger Westermann, Leif Kobbelt, and Tom Ertl. Real-time Exploration of Regular Volume Data by Adaptive Reconstruction of Iso-Surfaces. *The Visual Computer*, 15(2):100–111, 1999.

[WKME03] Manfred Weiler, Martin Kraus, Markus Merz, and Thomas Ertl. Hardware-Based Ray Casting for Tetrahedral Meshes. In *VIS '03: Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*, pages 333–340, Washington, DC, USA, 2003. IEEE Computer Society.

[WMW86] Geoff Wyvill, Craig McPheeters, and Brian Wyvill. Data structure for soft objects. *The Visual Computer*, 2:227–234, 1986.

[WS05] Ingo Wald and Hans-Peter Seidel. Interactive Ray Tracing of Point Based Models. In *Proceedings of 2005 Symposium on Point Based Graphics (PGB)*, page to appear, 2005.

[WV92] J Wilhelms and A Van Gelder. Octrees for faster isosurface generation. *ACM Transactions on Graphics*, 11(3):201–227, July 1992.

[ZG06] Yuan Zhou and Michael Garland. Interactive Point-Based Rendering of Higher-Order Tetrahedral Data. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1229–1236, 2006.

Survey of Techniques for Data-dependent Triangulations

Burkhard Lehner[1], Georg Umlauf[1], and Bernd Hamann[2]
1. Department of Computer Science, University of Kaiserslautern, Germany,
{lehner|umlauf}@informatik.uni-kl.de
2. Institite for Data Analysis and Visualization (IDAV) and
Department of Computer Science, University of California, Davis, USA,
bhamann@ucdavis.edu

Abstract: We present a survey of different techniques to approximate a color image using a piecewise linear interpolation induced by a triangulation of the image domain. We also include a detailed description of a method we designed. We give a short overview of possible applications and extentions.

1 Introduction

Given a scalar field $f : \Omega \to \mathbb{R}$ over a 2-dimensional domain $\Omega \subset \mathbb{R}^2$, a triangulation T of the domain Ω can be used to approximate the scalar field by linear interpolation of the corner values for every triangle t of T. The approximation quality, i.e., the distance between the approximation and the scalar field, depends on three factors:

- the number of triangles in T
- the vertex positions, i.e., the corners of the triangles
- the connectivity of the vertices, i.e., the vertex combination of the triangles

If T adapts well to features of f, the approximation quality can be high even with a small number of triangles.

Since we are looking for triangulations that define an approximation that has a high approximation quality with respect to f, the criterion to decide which triangulation to use depends on our data, the scalar field f. The result of this search is therefore called a *data-dependent triangulation*. Even if the number of triangles and the position of the vertices is fix, the number of possible triangulations is large, so that an exhaustive search for the optimal data-dependent triangulation is not possible except for trivial cases.

A subset of all possible triangulations is the set of all Delaunay triangulations. They are widely used in fields like finite element methods (FEM) because of their min-max-angle property, i.e., the smallest angle of all angles in the triangulation is as large as possible. This avoids long, skinny triangles which may lead to numerical instabilities. But for the

purpose of approximating f, long, slim triangles may be well suited, if f contains high-gradient regions indicative of feature boundaries (see [Rip92]). Therefore, the restriction to Delaunay triangulations may prevent the detection of the optimal triangulation.

Computer images are usually defined as discrete scalar fields $f : \Omega \rightarrow C$, with a domain $\Omega = \{0, \ldots, w-1\} \times \{0, \ldots, h-1\}$ as a regular grid of width w and height h, into a color space C, usually a three dimensional space, e.g., the RGB space. If the approximation is stored instead of the image f, storage space can be saved. The search for a good data-dependent triangulation can be used as a data compression method for images.

The rest of this paper provides an overview of algorithms to find good data-dependent triangulation. It is organized as follows: Section 2 defines some categories of algorithm types, that will be used later to classify the presented methods. Sections 3 to 5 present algorithms for the search for data-dependent triangulations. Section 6 presents some extensions the authors are working on.

2 Algorithm Classification

Finding the best triangulation for approximating an arbitrary scalar field is a non-linear optimization problem. An exhaustive search could be done using a backtracking algorithm, that lists all possible triangulations systematically, calculates the approximation quality, and finds the optimum. In all non-trivial cases, the search space is too large to enumerate all triangulations within reasonable time.

Instead, algorithms can be used that iteratively select one triangulation out of a limited set of candidates. After a number of iterations, a good triangulation that has at least an approximation quality close to the optimum is found.

One class of algorithms are *greedy algorithms*. In each iteration, they always select the candidate that best approximates the scalar field. Every decision that is made is final, and is never taken back. Because of the non-linearity of the problem, they often converge to a local optimum, but fail to find the global optimum.

Furthermore algorithms can be divided into deterministic and non-deterministic (stochastic) algorithms. In *deterministic* algorithms, every decision is based on a unique predicate, and for the same input the algorithm produces the same result. *Stochastic* algorithms employ a (pseudo) random number generator for decisions. Even for the same input, the result may be different, but the probability of a bad result is very small. Examples for stochastic algorithms are Monte Carlo methods, genetic algorithms (GA), and simulated annealing methods (see Sec. 2.1).

2.1 The Principle of Simulated Annealing

Simulated annealing is a stochastic, iterative method to solve a global optimization problem, i.e., finding the global extremum s^* of a given function $f(s) : D \rightarrow \mathbb{R}$ in a large search space D.

Starting with an arbitrary setting $s_0 \in D$, slightly modify s_0 to get $s'_0 \in D$. Using a probability function $p_{\text{accept}} : \mathbb{R} \times \mathbb{R} \times \mathbb{N} \to [0,1]$, s'_0 is accepted with probability $p_{\text{accept}}(f(s_0), f(s'_0), 0)$, i.e., $s_1 = s'_0$, otherwise it is rejected, i.e., $s_1 = s_0$. Iterating the process of changing s_i to $s'_i \in D$ and accepting it with probability $p_{\text{accept}}(f(s_i), f(s'_i), i)$ yields a sequence of settings s_i which converges to the global extremum s^* under certain assumptions on p_{accept} [KCDGV83].

In order to minimize f the probability function p_{accept} should satisfy the following properties, for $a, b, c \in f(D)$:

- A setting that reduces f has a larger probability to be accepted than a setting that increases f:
$$c > a > b \Rightarrow p_{\text{accept}}(a, b, i) > p_{\text{accept}}(a, c, i).$$
- For the sequence to converge, the probability for accepting settings that increase f must converge to 0, whereas the probability for accepting settings that reduce f must not converge to 0:
$$\lim_{i \to \infty} p_{\text{accept}}(a, b, i) = \begin{cases} 0 & \text{for } b > a \\ \text{const} \in]0, 1] & \text{for } b \leq a \end{cases}.$$

For example, for a greedy method the probability function

$$p_{\text{accept}}(a, b, i) = \begin{cases} 0 & \text{for } b \geq a \\ 1 & \text{for } b < a \end{cases}$$

is used, that accepts new settings only if they improve the result. This method can get stuck in a local minimum. A better choice for p_{accept} is

$$p_{\text{accept}}(a, b, i) = \begin{cases} \exp\left((a-b)/\tau_i\right) & \text{for } b \geq a \\ 1 & \text{for } b < a \end{cases}, \tag{1}$$

where $\tau_i = \tau_0 \tau_{\text{b}}^i$ is a temperature that starts at an initial value τ_0 and decreases exponentially to 0 by a factor $\tau_{\text{b}} \in [0, 1]$ in every iteration. Since also settings that increase f might be accepted, the sequence can escape a local minimum.

The temperatures τ_0 and τ_{b} define the annealing schedule. Their choice is vital for the result of the optimization process. If τ_i decreases too fast, the sequence can get stuck in a local minimum, if it decreases too slow, the sequence converges to a better local minimum (and possibly the global minimum), but it requires one to perform more iterations to reach it. It can be shown that by choosing the right annealing schedule the probability for finding the global minimum converges to one, but this usually implies a large number of iterations [KCDGV83].

3 Refinement Algorithms

Refinement algorithms start with a very coarse triangulation with a small number of triangles. Iteratively they insert more vertices and triangles, refining the triangulation. The

set of candidate triangulations for the next iteration is the set of all triangulations that can be created from the current triangulation by inserting one vertex at all possible positions. Since the number of possible positions can be high (or even infinite, if the domain is infinite), heuristics are employed to limit the number of insertion positions.

The greedy refinement algorithm presented in [GH95] inserts vertices into a Delaunay triangulation. Originally they used their algorithm to create high quality approximations of height fields, but it can easily be generalized to work on arbitrary scalar fields. Starting with a simple triangulation of the domain, consisting of just two triangles for a rectangular domain, in every step a new vertex is inserted at the position of the largest distance between the approximation and the scalar field. So, the set of candidates to select the next triangulation from consists of only one triangulation. This procedure is repeated until a specified error condition is met. Although specified for height fields, they also applied their algorithm to gray-scale images, showing that the approximation looks much better using their triangulation instead of a triangulation that distributes the same number of vertices uniformly over the domain.

Furthermore they modified their algorithm, dropping the Delaunay constraint for the triangulation, and using a locally optimal data-dependent triangulation instead, further improving the results.

The approach discussed in [SHB$^+$01] is similar. They are using a Sobolev norm to calculate the distance between the approximation and the original data, which emphasizes the regions of high curvature. In each iteration, they subdivide the triangle with the largest error. For this triangle, they detect "significant points" (data sites with high distance to the triangulation) near the midpoints of the edges and insert these into the triangulation, subdividing the selected triangle and its neighbors.

Since both approaches are greedy algorithms, they tend to get stuck in local optima. The approach described in [Ped01] attempts to improve this behavior. It uses Delaunay triangulations, and calculates a first triangulation by adding vertices at the site with largest distance from the current triangulation, just in the same way as in [GH95]. After falling below an error threshold, another greedy method is used to remove vertices, until the threshold is exceeded again. Some iterations of these refinement and decimation procedures are performed. This way, an approximation with less triangles is found, that also meets the error threshold. Often the iterations quickly fall into a loop, adding exactly those vertices that were removed in the last step, limiting the improvement of this approach over [GH95]. Its restriction to Delaunay triangulations is another drawback.

4 Decimation Algorithms

Decimation algorithms start with a very fine triangulation with a lot of triangles, and iteratively remove elements from it. The set of candidates to select the next triangulation is the set of all triangulations that can be created from the current triangulation by removing one element.

One example of a greedy decimation algorithm are the progressive meshes, described in

[Hop96]. Although defined for 2-manifolds, in [Hop96] it is also applied to images. Starting with a full triangulation that has a vertex at every pixel position of the image, one vertex after the other is removed using the edge collapse operation (see Fig. 1). From the set of triangulations that result from collapsing one of the edges of the current triangulation, the one with the best approximation quality is selected. A priority queue can be used to optimize this selection process: For every edge e_i of the triangulation the change in approximation quality $\Delta(e_i)$ is computed and stored. The edge e with the smallest change ($\forall i : \Delta(e) \leq \Delta(e_i)$) is collapsed. $\Delta(e_i)$ has to be recomputed only for those e_i that are incident to a triangle that was changed by the edge collapse operation.

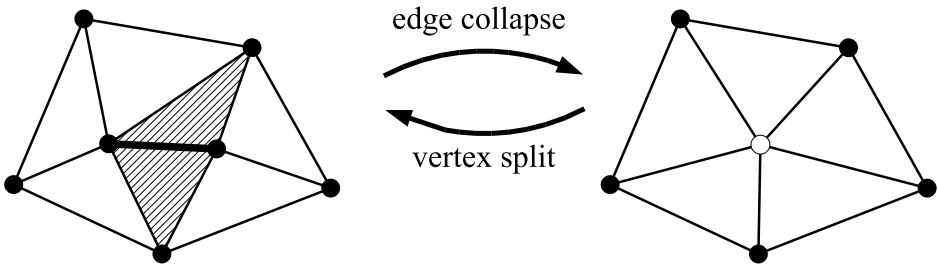


Figure 1: An edge collapse operation. The thick edge is collapsed to the unfilled vertex, the marked triangles are removed.

One example of the application of this approach to an image is shown in Fig. 4(f), the numerical results are presented in Table 1.

The inverse operation of the edge collapse is the vertex split. If the information for reversing the edge collapse is stored in a list, a *progressive mesh* can be defined as a series of triangulations, each having one vertex more than the previous one. This data structure can be used to provide a fine-grain set of triangulations with different approximation quality, and also other methods like selective refinement are possible.

5 Modification Algorithms

The class of *modification algorithms* starts with an arbitrary initial triangulation, and improves the approximation quality by performing a number of modification operations. The algorithms differ in the set of modification operations they utilize and the type of decision finding what operation to perform next.

In [Law77] the theoretical basis for algorithms that use an edge swap as the only modification operation was presented. An edge swap operation replaces two triangles (v_a, v_b, v_c) and (v_b, v_d, v_c) that share the edge (v_b, v_d) and that form a convex quadrilateral with the triangles (v_a, v_b, v_d) and (v_a, v_d, v_c) that share the other diagonal (v_a, v_d) of the quadrilateral (see Fig. 2). This operation keeps the number and position of the vertices fixed, and only modifies their connectivity. It is discussed in [Law77], what conditions must be met to guarantee the validity of the triangulation after performing the edge swap opera-

tion. Furthermore, the following general algorithm is discussed in detail: From the set of edges E an edge $e \in E$ is selected that fulfills a specified predicate. e is swapped, and this procedure is repeated until there is no $e \in E$ that satisfiesfulfills the predicate. It is shown in [Law77] that this algorithm always terminates when the predicate is the circumcircle criterion, and that the final triangulation is the Delaunay triangulation of the sites.

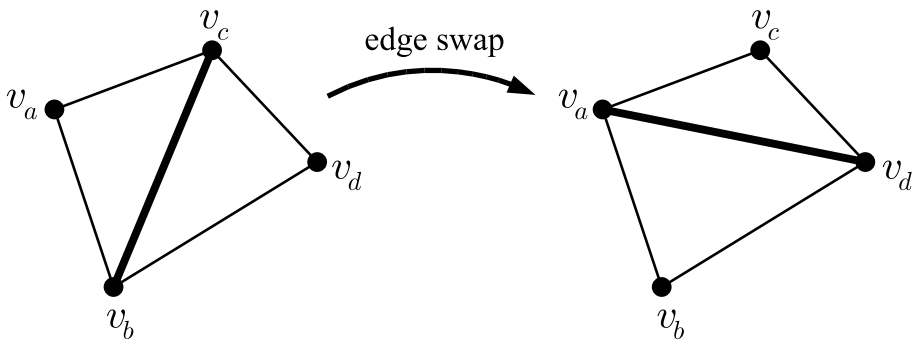


Figure 2: An edge swap. The thick edge is swapped.

In [DLR90] a set of fixed samples (vertices) of a scalar function $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ is given, and the triangulation connecting these vertices is sought, so that the piecewise linear approximation induced by the triangulation fits f best. Different cost functions are defined that judge the quality of the representation. To find the best triangulation with respect to the cost function, an arbitrary starting triangulation is improved iteratively: An edge of the triangulation is chosen randomly, and if swapping that edge (if possible) reduces the global cost function, it is swapped. This edge is then called "locally optimal." This procedure is repeated until every edge in the triangulation is locally optimal. An edge may be swapped several times, because swapping an edge of an incident triangle can lead to the edge being not locally optimal any more. The algorithm is guaranteed to stop, because every swapped edge reduces the cost which has a lower bound of 0, and there is only a finite set of triangulations. If f is known, choosing the L_2 distance between approximation and f as the cost function achieves results that are superior to cost functions that do not take f into account and use angles between triangles instead. Since this method is a greedy method, only making moves that reduce the cost function, it easily falls into a local minimum of the cost function, where all the edges are locally optimal, but which is not the global minimum of the cost function. Furthermore, the result depends on the order of edges that are tested for swapping.

In [Sch93], a simulated annealing approach (see Section 2.1) is used to improve the results of [DLR90] and to find a lower local (and hopefully the global) minimum of the cost function.

The methods in [Law77], [DLR90], and [Sch93] all have the number and position of the vertices fixed. Even better approximations can be found, if also the position of the vertices can be modified during the optimization procedure. In [KH01] a single simulated annealing approach is used to optimize both the vertex positions and their connectivity at

the same time. Additionally to the edge swap operation, a vertex move operation is used to change the position of one vertex of the triangulation (see Fig. 3). For each iteration of the simulated annealing loop the type of operation (edge swap or local vertex move) is chosen randomly. The algorithm is applied to scattered data problems, and also to color images.

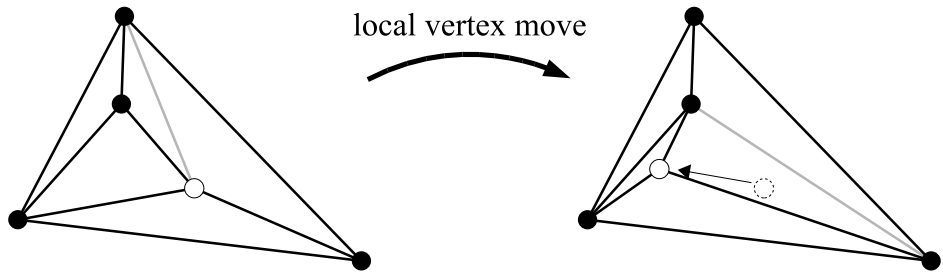


Figure 3: A local vertex move. The unfilled vertex is moved, the gray edge is swapped to prevent a degenerate triangle.

The approach of [PK03] specializes on images. It simplifies the modification operations of [KH01]. Furthermore, it uses the following greedy refinement strategy to start with a good initial triangulation: Starting with a primitive triangulation of the image domain, consisting of two triangles, the next site is added within the triangle with the largest approximation error. The position of the inserted site is the error barycenter of the pixels of that triangle, i.e., the average of all pixel coordinates weighted by their approximation error. This process is iterated until the specified number of sites is reached.

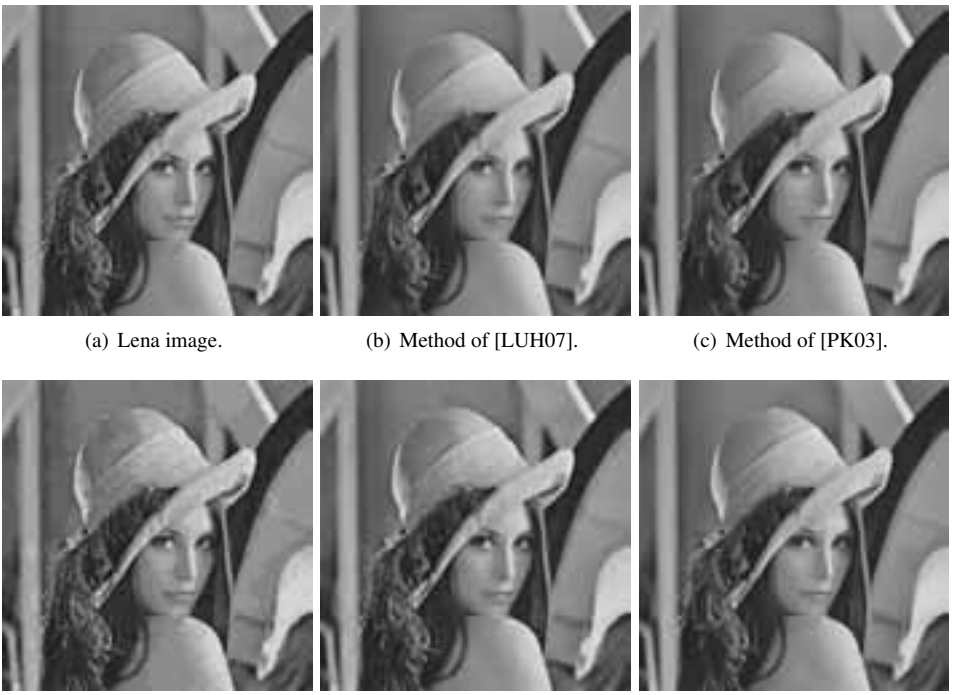
Further improvements are implemented by the authors and described in [LUH07]. In that approach, additionally to the position and connectivity of the vertices, also the number of triangles in the triangulation is modified. For this, a total approximation error is specified by the user, and the algorithm adds or removes vertices during the simulated annealing process to achieve this error as good as possible. So this approach is the first to include all three factors that have an influence to the approximation quality into the optimization process. The approximation quality is measured using the CIEL*a*b* color space that takes the reception of the human eye into account.

Furthermore, we implemented so-called *guides*, heuristics to improve the convergence of the simulated annealing, and with this speeding up the calculation significantly. The guides assign higher probabilities for being selected for edge swap and vertex move to edges / vertices in regions of high approximation error, because modifications in these regions are more likely to improve the approximation. The *selection guides* assign higher probabilities to be selected to edges for edge swap operation and to vertices for vertex move operations, respectively. The *placement guides* assign probabilities to the destination position for vertex move operations. Experiments show that the number of iterations can be reduced by a factor of 8.

We also defined a file format for storing the information necessary to reconstruct the approximation, and compared the compression ratio of this method with the well-known image compression standards JPEG and JPEG2000. We showed that the image compres-

sion method using a data-dependent triangulation can compete with these standards for a large variety of color images. Fig. 4 shows the results of image compression using different data-dependent triangulations (Figs. 4(b), 4(c) and 4(f)), with JPEG (Fig. 4(d)), and with JPEG2000 (Fig. 4(e)). All compressed images have a file size of $\sim 5\,750$ bytes.

Table 1 shows some numerical results of the comparison for the images in Fig. 4. The approximation error is measured as the RMSE error of the approximation, measured in the CIEL*a*b* color space. Only for comparison of our method to the method of [PK03] we measured the error in the RGB color space that was employed in the original paper. The method of [LUH07] has the best approximation quality of the five methods. Especially JPEG compression gives poor results for this high compression ratio of 1:140 (0.18 bits per pixel).



(a) Lena image. (b) Method of [LUH07]. (c) Method of [PK03].

(d) JPEG. (e) JPEG2000. (f) Progressive Meshes [Hop96].

Figure 4: Original image ($\sim$770 kB) (a), and approximations ($\sim$5 750 bytes) using Simulated Annealing [LUH07] (b), Simulated Annealing [PK03] (c), JPEG (d), JPEG2000 (e), and Progressive Meshes [Hop96] (f), (courtesy USC SIPI)

	Sim. Anneal. [PK03]	Sim. Anneal. [LUH07]		Prog. Meshes [Hop96]	JPEG	JPEG 2000
Colormodel	RGB	RGB	Lab	Lab	Lab	Lab
RMSE	17.26	16.45	6.00	6.16	10.38	6.81

Table 1: Numerical results of approximation error (Root Mean Square Error, RMSE) for different algorithms for data-dependent triangulations, and for JPEG and JPEG2000 compression.

6 Work in Progress

The authors are currently working on extending the concept introduced in [LUH07] to video clips and video streams. They apply the image approximation technique to the frames of a video clip, and gain a speed-up by using the data-dependent triangulation of one frame as initial triangulation for the next one.

Furthermore, they extend the concept of [LUH07] to tetrahedrizations for approximating a scalar field defined over a 3-dimensional domain. This could be applied to a video clip or stream, where two dimensions are the x- and y-coordinates of the image, and the third dimension is time, i.e., the frame index.

Acknowledgments This work was supported by the DFG IRTG 1131 "Visualization of Large and Unstructured Data Sets", University of Kaiserslautern, and NSF contract ACI 9624034 (CAREER Award) and a large ITR grant. We thank the members of the Visualization and Computer Graphics Research Group at IDAV, and the Geometric Algorithms Group at the University of Kaiserslautern.

References

[DLR90] Nira Dyn, David Levin, and Shmuel Rippa. Data Dependent Triangulations for Piecewise Linear Interpolations. *IMA J. of Numerical Analysis*, 10(1):137–154, Jan 1990.

[GH95] Michael Garland and Paul Heckbert. Fast Polygonal Approximation of Terrains and Height Fields. Technical report, CS Department, Carnegie Mellon University, September 1995.

[Hop96] Hugues Hoppe. Progressive meshes. In *SIGGRAPH '96*, pages 99–108, 1996.

[KCDGV83] S. Kirkpatrick, Jr. C. D. Gelatt, and M. P. Vecchi. Optimization by Simulated Annealing. *Science Magazine*, pages 671–680, may 1983.

[KH01] Oliver Kreylos and Bernd Hamann. On Simulated Annealing and the Construction of Linear Spline Approximations for Scattered Data. *IEEE TVCG*, 7(1):17–31, 2001.

[Law77] C. L. Lawson. Software for C^1 surface interpolation. In *Mathematical Software III*, pages 161–194. Academic Press, New York, 1977.

[LUH07] B. Lehner, G. Umlauf, and B. Hamann. Image Compression Using Data-dependent Triangulations. In G. Bebis et al., editor, *International Symposium on Visualization*

and Computer Graphics (ISVC) 2007, Part I, LNCS 4841, to appear, Lecture Notes on Computer Science, pages 351–362. Springer, 2007.

[Ped01] H. Pedrini. An improved Refinement and Decimation Method for Adaptive Terrain Surface Approximation. In *Proceedings of WSCG*, pages 103–109. Czech Republic, 2001.

[PK03] Vid Petrovic and Falko Kuester. Optimized Construction of Linear Approximations to Image Data. In *Proc. 11th Pacific Conf. on Comp. Graphics and Appl.*, pages 487–491, 2003.

[Rip92] Shmuel Rippa. Long and thin triangles can be good for linear interpolation. *SIAM J. Numer. Anal.*, 29(1):257–270, 1992.

[Sch93] Larry L. Schumaker. Computing Optimal Triangulations Using Simulated Annealing. *Computer Aided Geometric Design*, 10(3-4):329–345, 1993.

[SHB+01] Rene Schaetzl, Hans Hagen, James Barnes, Bernd Hamann, and Kenneth Joy. Data-Dependent Triangulation in the Plane with Adaptive Knot Placement. In Guido Brunnett, H. Bieri, and Gerald Farin, editors, *Geometric Modelling, Comp. Suppl. 14*, pages 199–218. Springer, 2001.

Colorplates

A Framework for Visualizing Multivariate Geodata
A. Middel (p. 13)

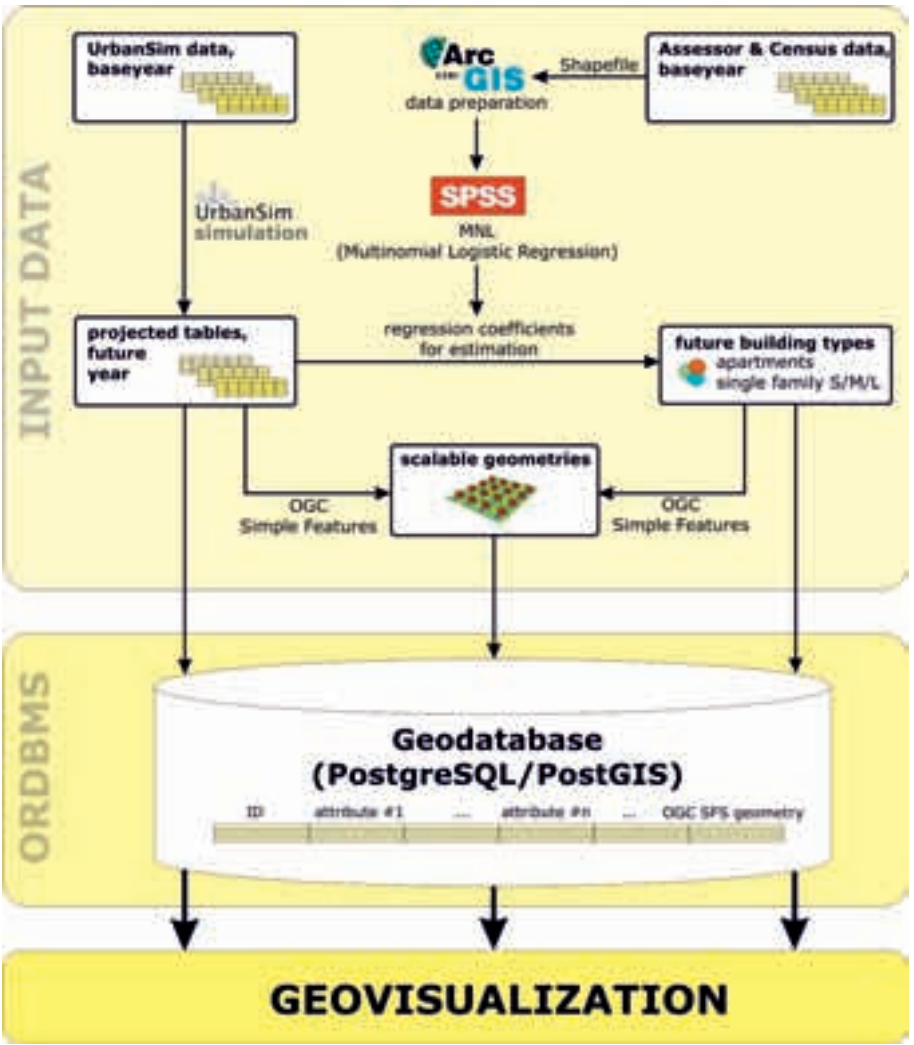


Figure 5: System Architecture

Comparative Tensor Visualisation
within the Framework of Consistent Time-Stepping Schemes
R. Mohr, T. Bobach, Y. Hijazi, G. Reis, P. Steinmann, H. Hagen (p. 36)

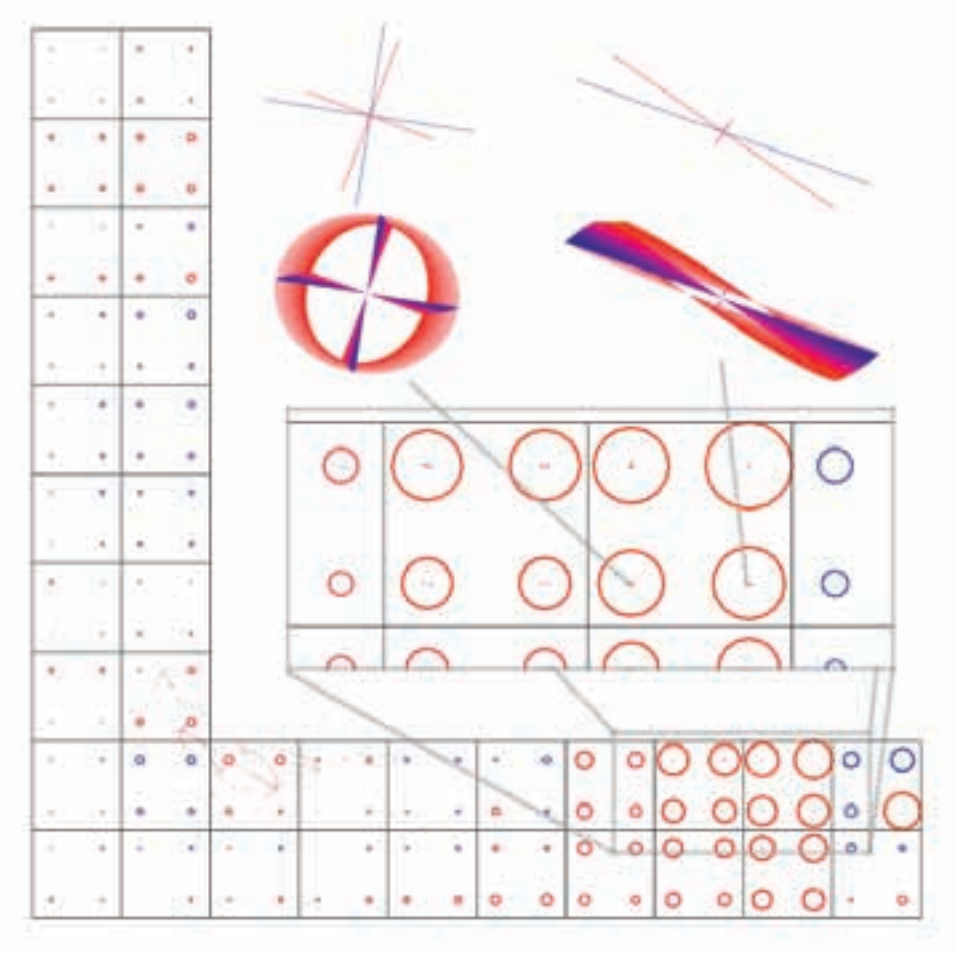


Figure 6: Level-of-detail visualisation, showing more qualitative facts at a large scale while depicting the concrete deformation on a per-tensor basis after an appropriate zoom. The pixel clutter visible in the angle of the 'L' is actually a swept ellipse similar to the one depicted in the second top row, yet very big and very thin.

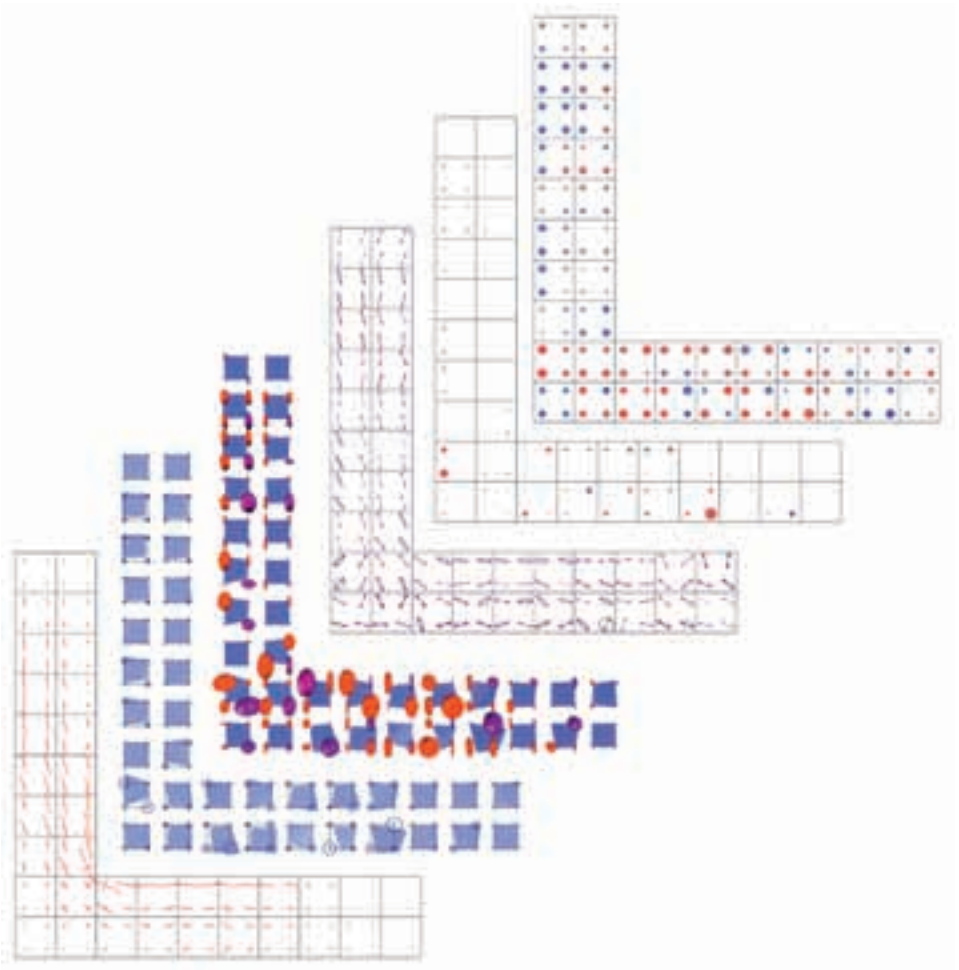
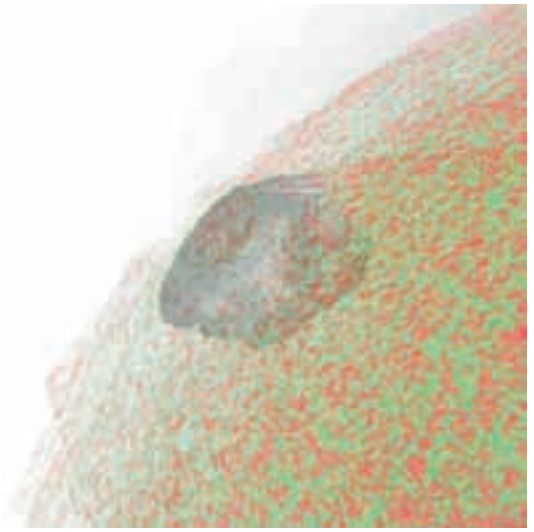
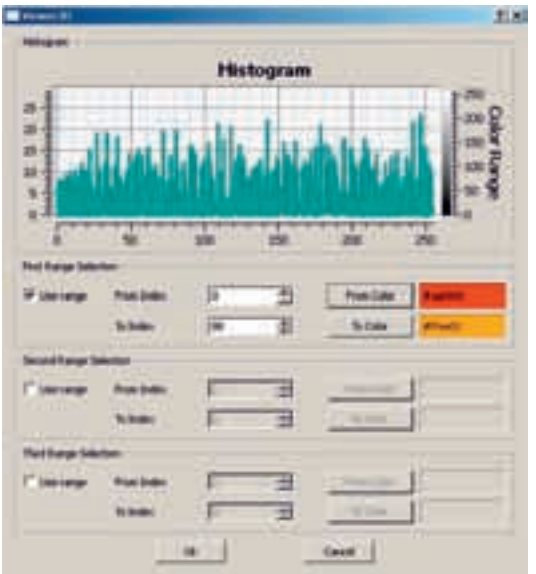


Figure 7: A collection of all presented visualisation tools applied to the same data set and time step.

A Framework for the Visualization of Brain Structures
S. Thelen, T. Bierz, B. Müller, H. Hagen, A. Ebert, E. Friauf, J. Meyer (p. 54)



(a) Example of a color-mapped scene



(b) The framework's dialog for specifying a transfer function. Three disjoint sets of grayscale values can be defined over which RGB-colors are interpolated linearly.

(c) Captured Image of pointing gesture

(d) Captured Image of a grabbing gesture

Figure 8: Opening of a binary image

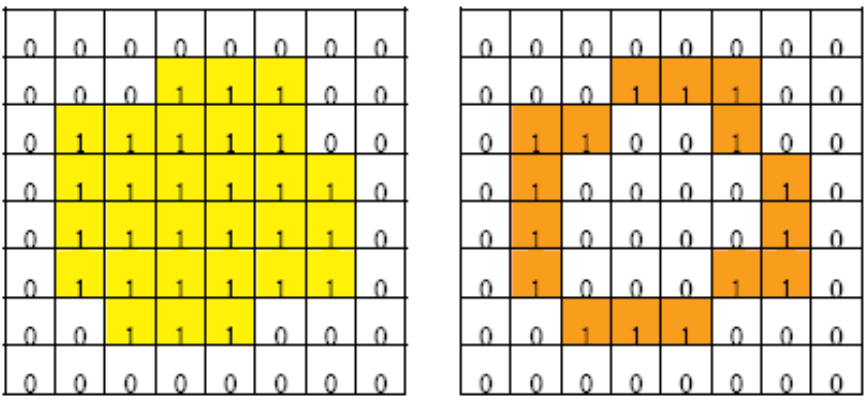
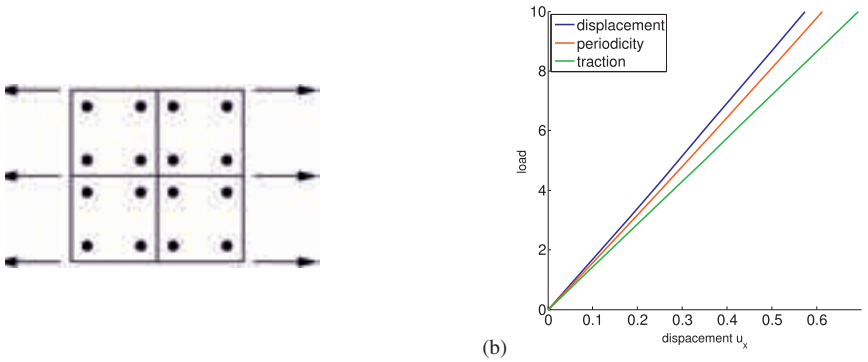


Figure 9: Original image and detected contours

Deriving Global Material Properties of a Microscopically Heterogeneous Medium – Computational Homogenisation and Opportunities in Visualisation
C. B. Hirschberger, S. Ricker, P. Steinmann, N. Sukumar (p. 76)



(a) (b)

Figure 10: (a) Macro boundary value problem discretised with four elements with four integration points per element. (b) Comparison of macro load displacement curves for different RVE boundary conditions.

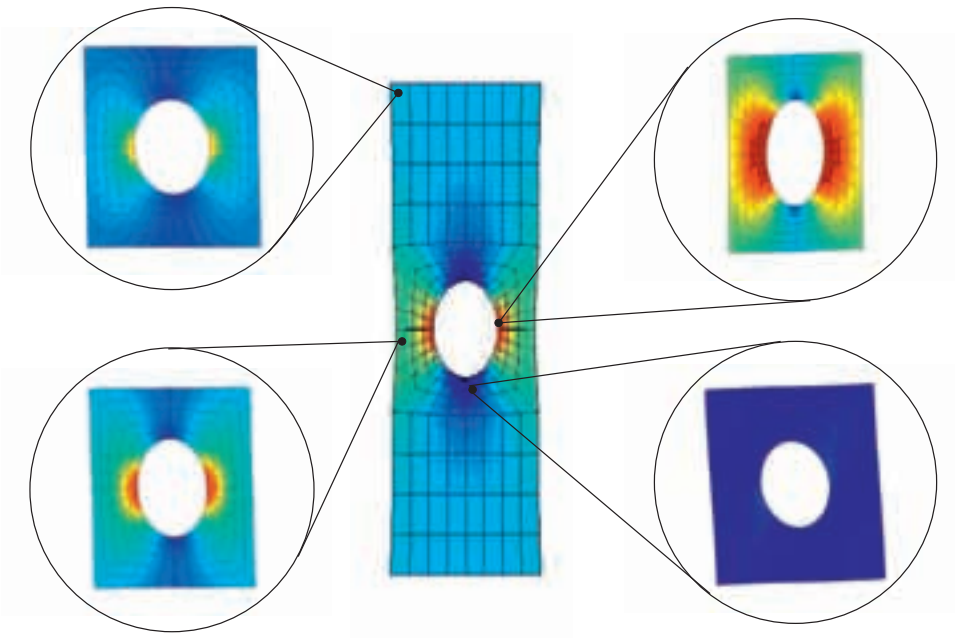


Figure 11: Deformed specimen with circular hole under longitudinal tension and deformed RVEs with logitudinal normal stress component $\bar{\sigma}_{22}$ and σ_{22}, respectively.

Towards completeness, a multiscale approach of confined particulate systems.
H. A. Meier, P. Steinmann, E. Kuhl (p. 103)

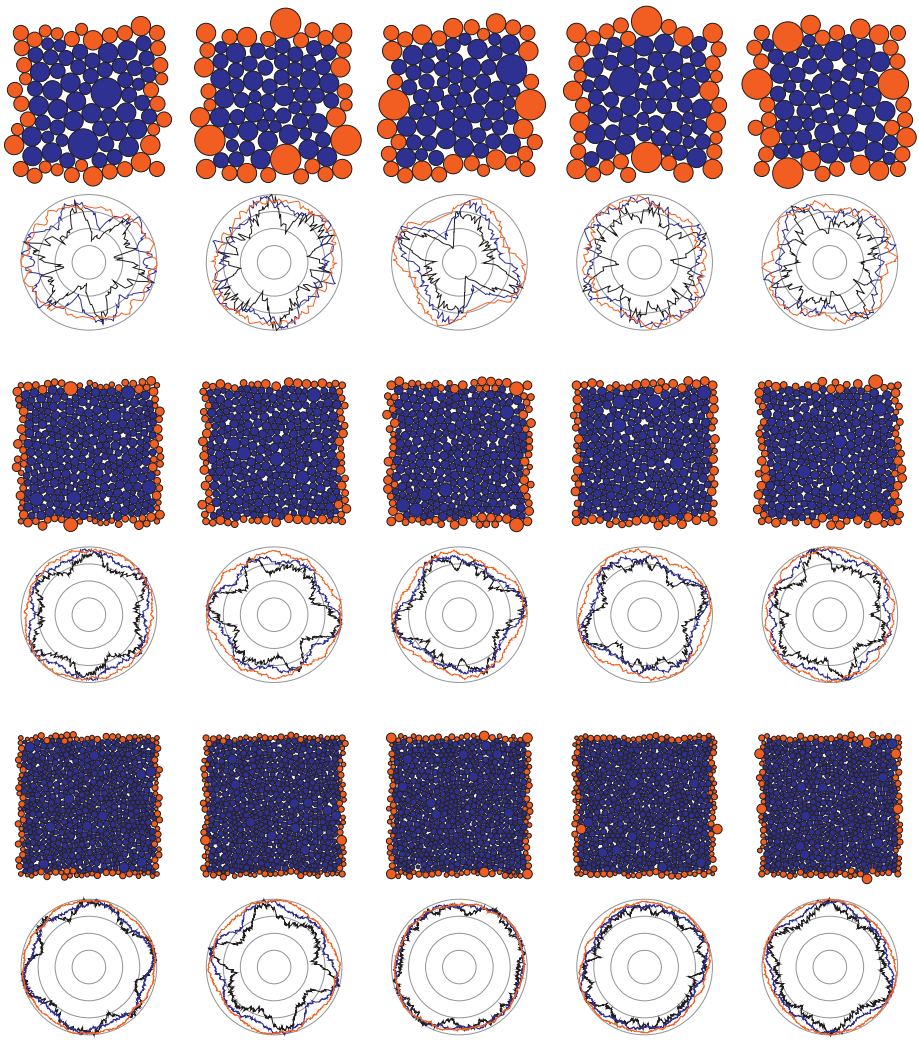


Figure 12: rves and their corresponding contact normal density functions. Black colored functions relate to an angle of influence equal to 10°, while blue and red colored functions correspond to an angle of influence equal to 20° and 30°, respectively. First set: Five rves, containing each 70 primary particles. Second set: Five rves, containing 350 primary particles. Third set: Five rves containing 700 primary particles.

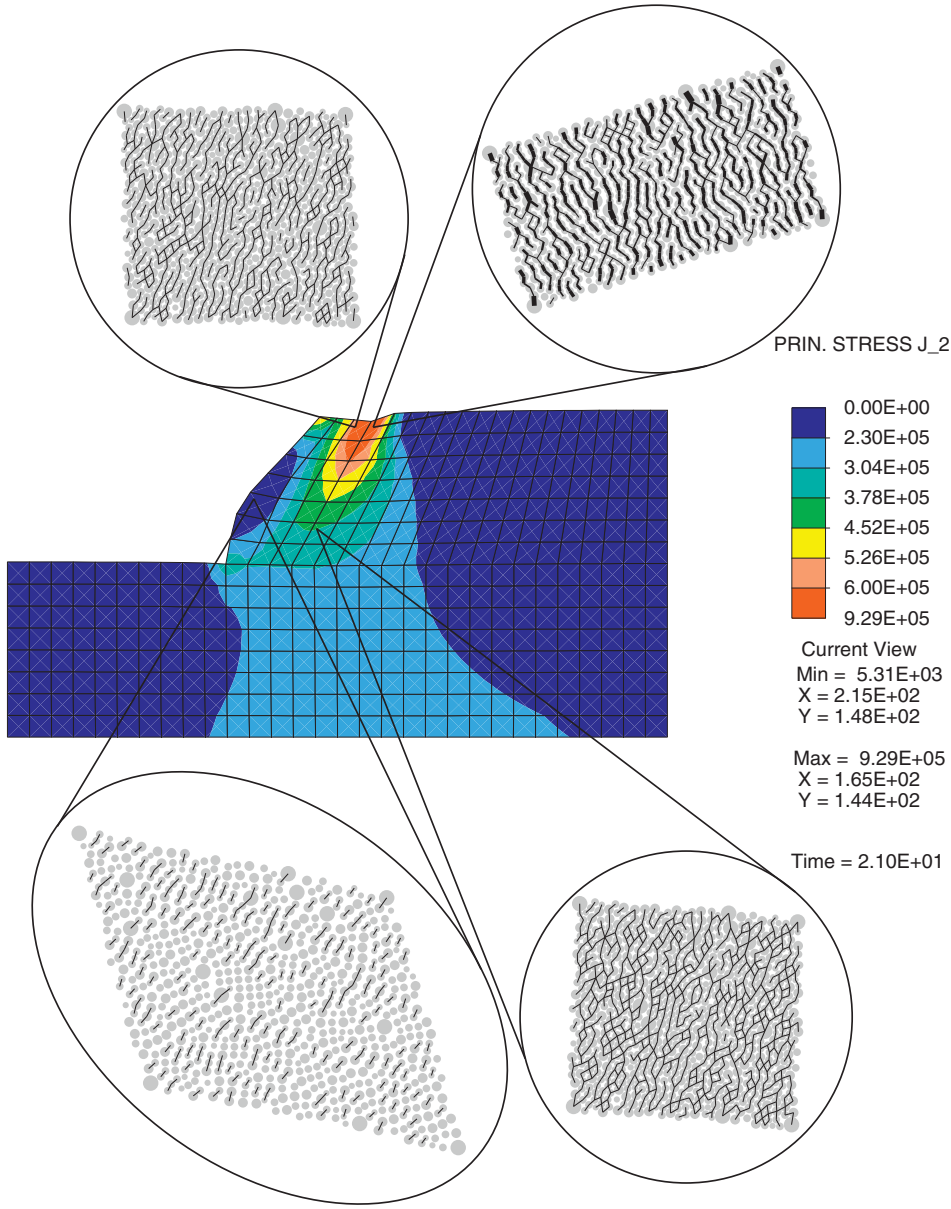


Figure 13: Final, deformed slope. The macroscopic von Mises stress is plotted. Initial localization is visible. Four deformed rves containing their contact networks are depicted. The tickness of the branches corresponds to the magnitude of the contact forces. The scaling factors for the branch thickness as well as for the deformation are equal for all four particle plots.

PointCloudXplore 2: Visual Exploration of 3D Gene Expression

O. Rübel, G.H. Weber, M.-Y. Huang, E.W. Bethel , S.V.E. Keränen, C.C. Fowlkes,
C.L. Luengo Hendriks, A. H. DePace, L. Simirenko, M.B. Eisen, M.D. Biggin, H. Hagen,
J. Malik, D. W. Knowles, B. Hamann (p. 125)

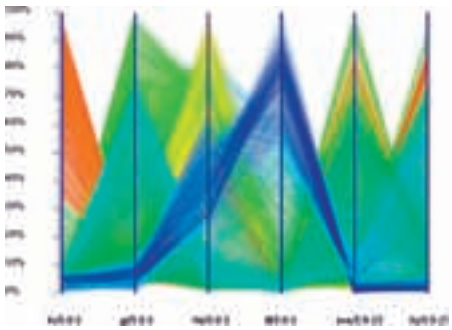


Figure 3: 2D Parallel coordinate view showing the expression of six selected genes in all blastoderm cells of the embryo.

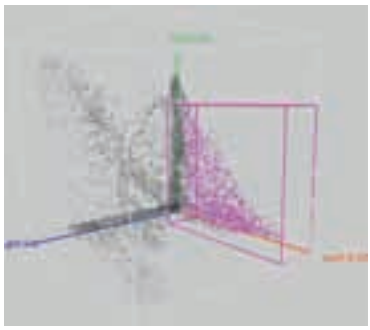


Figure 6: By drawing an axes aligned box in a scatter-plot one can select cells with respect to the expression of up to three genes.

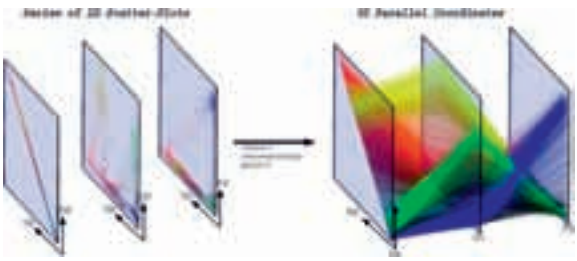


Figure 4: 3D Parallel coordinates are a direct extension to regular parallel coordinates and combine the principle of 2D-scatter-plots and 2D-parallel coordinates.

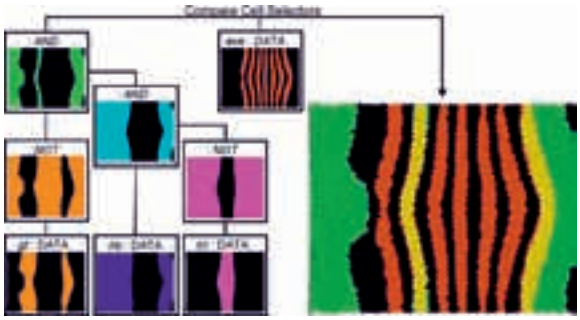


Figure 9: Combining cell selectors defining the patterns of *gt*, *hb*, and *Kr* using logical operators. The selection result is then compared with a cell selector that selects all cells expressing *eve* at medium to high levels by defining the overlay of both selections.

Geomodeling and Geovisualizations in Urban Planning und Real Estate Industry: The Example of Office Market Research
C. v. Malottki (p. 138)

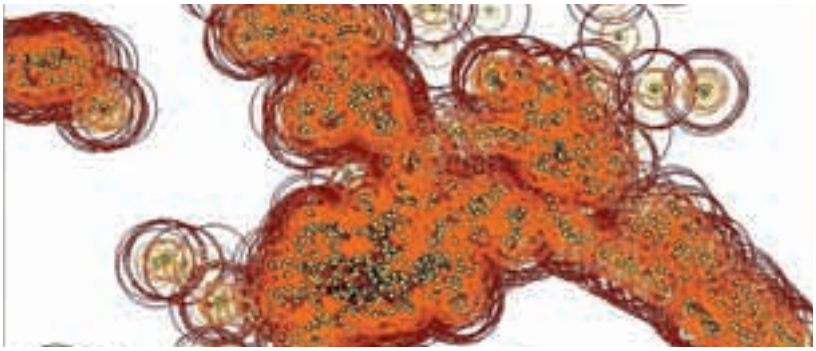


Figure 10: Methodology: The geographic influence factors are raised by influence zones around each building in the sample.

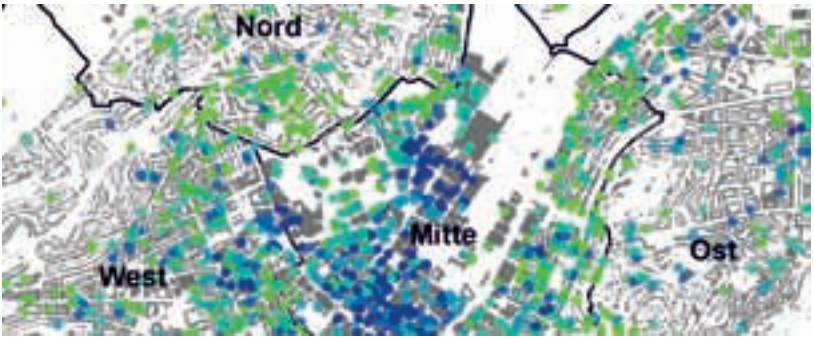


Figure 11: Forecasting price levels for all office buildings in Stuttgart (green = low price level, blue = high price level)

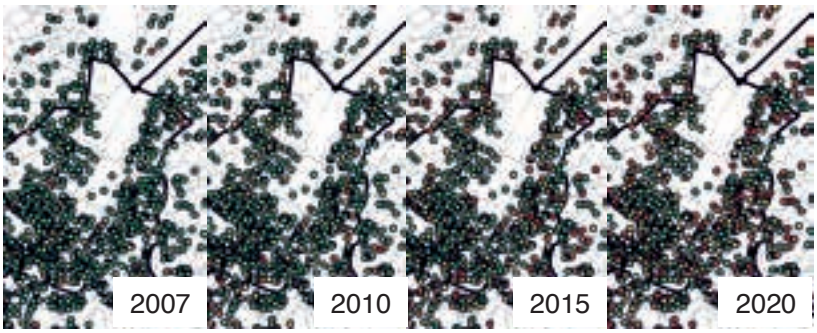


Figure 12: Exemplary simulation results for a leaving scenario. The colors indicate the vacancy rates of the buildings.

GI-Edition Lecture Notes in Informatics – Proceedings

P-1 Gregor Engels, Andreas Oberweis, Albert Zündorf (Hrsg.): Modellierung 2001.

P-2 Mikhail Godlevsky, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications, ISTA'2001.

P-3 Ana M. Moreno, Reind P. van de Riet (Hrsg.): Applications of Natural Language to Information Systems, NLDB'2001.

P-4 H. Wörn, J. Mühling, C. Vahl, H.-P. Meinzer (Hrsg.): Rechner- und sensorgestützte Chirurgie; Workshop des SFB 414.

P-5 Andy Schürr (Hg.): OMER – Object-Oriented Modeling of Embedded Real-Time Systems.

P-6 Hans-Jürgen Appelrath, Rolf Beyer, Uwe Marquardt, Heinrich C. Mayr, Claudia Steinberger (Hrsg.): Unternehmen Hochschule, UH'2001.

P-7 Andy Evans, Robert France, Ana Moreira, Bernhard Rumpe (Hrsg.): Practical UML-Based Rigorous Development Methods – Countering or Integrating the extremists, pUML'2001.

P-8 Reinhard Keil-Slawik, Johannes Magenheim (Hrsg.): Informatikunterricht und Medienbildung, INFOS'2001.

P-9 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Innovative Anwendungen in Kommunikationsnetzen, 15. DFN Arbeitstagung.

P-10 Mirjam Minor, Steffen Staab (Hrsg.): 1st German Workshop on Experience Management: Sharing Experiences about the Sharing Experience.

P-11 Michael Weber, Frank Kargl (Hrsg.): Mobile Ad-Hoc Netzwerke, WMAN 2002.

P-12 Martin Glinz, Günther Müller-Luschnat (Hrsg.): Modellierung 2002.

P-13 Jan von Knop, Peter Schirmbacher and Viljan Mahni_ (Hrsg.): The Changing Universities – The Role of Technology.

P-14 Robert Tolksdorf, Rainer Eckstein (Hrsg.): XML-Technologien für das Semantic Web – XSW 2002.

P-15 Hans-Bernd Bludau, Andreas Koop (Hrsg.): Mobile Computing in Medicine.

P-16 J. Felix Hampe, Gerhard Schwabe (Hrsg.): Mobile and Collaborative Busi-ness 2002.

P-17 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Zukunft der Netze –Die Verletzbarkeit meistern, 16. DFN Arbeitstagung.

P-18 Elmar J. Sinz, Markus Plaha (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2002.

P-19 Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3.Okt. 2002 in Dortmund.

P-20 Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3.Okt. 2002 in Dortmund (Ergänzungsband).

P-21 Jörg Desel, Mathias Weske (Hrsg.): Promise 2002: Prozessorientierte Methoden und Werkzeuge für die Entwicklung von Informationssystemen.

P-22 Sigrid Schubert, Johannes Magenheim, Peter Hubwieser, Torsten Brinda (Hrsg.): Forschungsbeiträge zur "Didaktik der Informatik" – Theorie, Praxis, Evaluation.

P-23 Thorsten Spitta, Jens Borchers, Harry M. Sneed (Hrsg.): Software Management 2002 – Fortschritt durch Beständigkeit

P-24 Rainer Eckstein, Robert Tolksdorf (Hrsg.): XMIDX 2003 – XML-Technologien für Middleware – Middleware für XML-Anwendungen

P-25 Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Commerce – Anwendungen und Perspektiven – 3. Workshop Mobile Commerce, Universität Augsburg, 04.02.2003

P-26 Gerhard Weikum, Harald Schöning, Erhard Rahm (Hrsg.): BTW 2003: Datenbanksysteme für Business, Technologie und Web

P-27 Michael Kroll, Hans-Gerd Lipinski, Kay Melzer (Hrsg.): Mobiles Computing in der Medizin

P-28 Ulrich Reimer, Andreas Abecker, Steffen Staab, Gerd Stumme (Hrsg.): WM 2003: Professionelles Wissensmanagement – Erfahrungen und Visionen

P-29 Antje Düsterhöft, Bernhard Thalheim (Eds.): NLDB'2003: Natural Language Processing and Information Systems

P-30 Mikhail Godlevsky, Stephen Liddle, Heinrich C. Mayr (Eds.): Information Systems Technology and its Applications

P-31 Arslan Brömme, Christoph Busch (Eds.): BIOSIG 2003: Biometric and Electronic Signatures

P-32 Peter Hubwieser (Hrsg.): Informatische Fachkonzepte im Unterricht – INFOS 2003

P-33 Andreas Geyer-Schulz, Alfred Taudes (Hrsg.): Informationswirtschaft: Ein Sektor mit Zukunft

P-34 Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 1)

P-35 Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 2)

P-36 Rüdiger Grimm, Hubert B. Keller, Kai Rannenberg (Hrsg.): Informatik 2003 – Mit Sicherheit Informatik

P-37 Arndt Bode, Jörg Desel, Sabine Rathmayer, Martin Wessner (Hrsg.): DeLFI 2003: e-Learning Fachtagung Informatik

P-38 E.J. Sinz, M. Plaha, P. Neckel (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2003

P-39 Jens Nedon, Sandra Frings, Oliver Göbel (Hrsg.): IT-Incident Management & IT-Forensics – IMF 2003

P-40 Michael Rebstock (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2004

P-41 Uwe Brinkschulte, Jürgen Becker, Dietmar Fey, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle, Thomas Runkler (Edts.): ARCS 2004 – Organic and Pervasive Computing

P-42 Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Economy – Transaktionen und Prozesse, Anwendungen und Dienste

P-43 Birgitta König-Ries, Michael Klein, Philipp Obreiter (Hrsg.): Persistance, Scalability, Transactions – Database Mechanisms for Mobile Applications

P-44 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): Security, E-Learning. E-Services

P-45 Bernhard Rumpe, Wofgang Hesse (Hrsg.): Modellierung 2004

P-46 Ulrich Flegel, Michael Meier (Hrsg.): Detection of Intrusions of Malware & Vulnerability Assessment

P-47 Alexander Prosser, Robert Krimmer (Hrsg.): Electronic Voting in Europe – Technology, Law, Politics and Society

P-48 Anatoly Doroshenko, Terry Halpin, Stephen W. Liddle, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications

P-49 G. Schiefer, P. Wagner, M. Morgenstern, U. Rickert (Hrsg.): Integration und Datensicherheit – Anforderungen, Konflikte und Perspektiven

P-50 Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 1) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm

P-51 Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 2) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm

P-52 Gregor Engels, Silke Seehusen (Hrsg.): DELFI 2004 – Tagungsband der 2. e-Learning Fachtagung Informatik

P-53 Robert Giegerich, Jens Stoye (Hrsg.): German Conference on Bioinformatics – GCB 2004

P-54 Jens Borchers, Ralf Kneuper (Hrsg.): Softwaremanagement 2004 – Outsourcing und Integration

P-55 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): E-Science und Grid Adhoc-Netze Medienintegration

P-56 Fernand Feltz, Andreas Oberweis, Benoit Otjacques (Hrsg.): EMISA 2004 – Informationssysteme im E-Business und E-Government

P-57 Klaus Turowski (Hrsg.): Architekturen, Komponenten, Anwendungen

P-58 Sami Beydeda, Volker Gruhn, Johannes Mayer, Ralf Reussner, Franz Schweiggert (Hrsg.): Testing of Component-Based Systems and Software Quality

P-59 J. Felix Hampe, Franz Lehner, Key Pousttchi, Kai Ranneberg, Klaus Turowski (Hrsg.): Mobile Business – Processes, Platforms, Payments

P-60 Steffen Friedrich (Hrsg.): Unterrichtskonzepte für inforrmatische Bildung

P-61 Paul Müller, Reinhard Gotzhein, Jens B. Schmitt (Hrsg.): Kommunikation in verteilten Systemen

P-62 Federrath, Hannes (Hrsg.): „Sicherheit 2005“ – Sicherheit – Schutz und Zuverlässigkeit

P-63 Roland Kaschek, Heinrich C. Mayr, Stephen Liddle (Hrsg.): Information Systems – Technology and ist Applications

P-64 Peter Liggesmeyer, Klaus Pohl, Michael Goedicke (Hrsg.): Software Engineering 2005

P-65 Gottfried Vossen, Frank Leymann, Peter Lockemann, Wolffried Stucky (Hrsg.): Datenbanksysteme in Business, Technologie und Web

P-66 Jörg M. Haake, Ulrike Lucke, Djamshid Tavangarian (Hrsg.): DeLFI 2005: 3. deutsche e-Learning Fachtagung Informatik

P-67 Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 1)

P-68 Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 2)

P-69 Robert Hirschfeld, Ryszard Kowalcyk, Andreas Polze, Matthias Weske (Hrsg.): NODe 2005, GSEM 2005

P-70 Klaus Turowski, Johannes-Maria Zaha (Hrsg.): Component-oriented Enterprise Application (COAE 2005)

P-71 Andrew Torda, Stefan Kurz, Matthias Rarey (Hrsg.): German Conference on Bioinformatics 2005

P-72 Klaus P. Jantke, Klaus-Peter Fähnrich, Wolfgang S. Wittig (Hrsg.): Marktplatz Internet: Von e-Learning bis e-Payment

P-73 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): “Heute schon das Morgen sehen“

P-74 Christopher Wolf, Stefan Lucks, Po-Wah Yau (Hrsg.): WEWoRC 2005 – Western European Workshop on Research in Cryptology

P-75 Jörg Desel, Ulrich Frank (Hrsg.): Enterprise Modelling and Information Systems Architecture

P-76 Thomas Kirste, Birgitta König-Riess, Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Informationssysteme – Potentiale, Hindernisse, Einsatz

P-77 Jana Dittmann (Hrsg.): SICHERHEIT 2006

P-78 K.-O. Wenkel, P. Wagner, M. Morgenstern, K. Luzi, P. Eisermann (Hrsg.): Land- und Ernährungswirtschaft im Wandel

P-79 Bettina Biel, Matthias Book, Volker Gruhn (Hrsg.): Softwareengineering 2006

P-80 Mareike Schoop, Christian Huemer, Michael Rebstock, Martin Bichler (Hrsg.): Service-Oriented Electronic Commerce

P-81 Wolfgang Karl, Jürgen Becker, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle (Hrsg.): ARCS´06

P-82 Heinrich C. Mayr, Ruth Breu (Hrsg.): Modellierung 2006

P-83 Daniel Huson, Oliver Kohlbacher, Andrei Lupas, Kay Nieselt and Andreas Zell (eds.): German Conference on Bioinformatics

P-84 Dimitris Karagiannis, Heinrich C. Mayr, (Hrsg.): Information Systems Technology and its Applications

P-85 Witold Abramowicz, Heinrich C. Mayr, (Hrsg.): Business Information Systems

P-86 Robert Krimmer (Ed.): Electronic Voting 2006

P-87 Max Mühlhäuser, Guido Rößling, Ralf Steinmetz (Hrsg.): DELFI 2006: 4. e-Learning Fachtagung Informatik

P-88 Robert Hirschfeld, Andreas Polze, Ryszard Kowalczyk (Hrsg.): NODe 2006, GSEM 2006

P-90 Joachim Schelp, Robert Winter, Ulrich Frank, Bodo Rieger, Klaus Turowski (Hrsg.): Integration, Informationslogistik und Architektur

P-91 Henrik Stormer, Andreas Meier, Michael Schumacher (Eds.): European Conference on eHealth 2006

P-92 Fernand Feltz, Benoît Otjacques, Andreas Oberweis, Nicolas Poussing (Eds.): AIM 2006

P-93 Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 1

P-94 Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 2

P-95 Matthias Weske, Markus Nüttgens (Eds.): EMISA 2005: Methoden, Konzepte und Technologien für die Entwicklung von dienstbasierten Informationssystemen

P-96 Saartje Brockmans, Jürgen Jung, York Sure (Eds.): Meta-Modelling and Ontologies

P-97 Oliver Göbel, Dirk Schadt, Sandra Frings, Hardo Hase, Detlef Günther, Jens Nedon (Eds.): IT-Incident Mangament & IT-Forensics – IMF 2006

P-98 Hans Brandt-Pook, Werner Simonsmeier und Thorsten Spitta (Hrsg.): Beratung in der Softwareentwicklung – Modelle, Methoden, Best Practices

P-99 Andreas Schwill, Carsten Schulte, Marco Thomas (Hrsg.): Didaktik der Informatik

P-100 Peter Forbrig, Günter Siegel, Markus Schneider (Hrsg.): HDI 2006: Hochschuldidaktik der Informatik

P-101 Stefan Böttinger, Ludwig Theuvsen, Susanne Rank, Marlies Morgenstern (Hrsg.): Agrarinformatik im Spannungsfeld zwischen Regionalisierung und globalen Wertschöpfungsketten

P-102 Otto Spaniol (Eds.): Mobile Services and Personalized Environments

P-103 Alfons Kemper, Harald Schöning, Thomas Rose, Matthias Jarke, Thomas Seidl, Christoph Quix, Christoph Brochhaus (Hrsg.): Datenbanksysteme in Business, Technologie und Web (BTW 2007)

P-104 Birgitta König-Ries, Franz Lehner, Rainer Malaka, Can Türker (Hrsg.) MMS 2007: Mobilität und mobile Informationssysteme

P-105 Wolf-Gideon Bleek, Jörg Raasch, Heinz Züllighoven (Hrsg.) Software Engineering 2007

P-106 Wolf-Gideon Bleek, Henning Schwentner, Heinz Züllighoven (Hrsg.) Software Engineering 2007 – Beiträge zu den Workshops

P-107 Heinrich C. Mayr, Dimitris Karagiannis (eds.) Information Systems Technology and its Applications

P-108 Arslan Brömme, Christoph Busch, Detlef Hühnlein (eds.) BIOSIG 2007: Biometrics and Electronic Signatures

P-109 Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, Marc Ronthaler (Hrsg.) INFORMATIK 2007 Informatik trifft Logistik Band 1

P-110 Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, Marc Ronthaler (Hrsg.) INFORMATIK 2007 Informatik trifft Logistik Band 2

P-111 Christian Eibl, Johannes Magenheim, Sigrid Schubert, Martin Wessner (Hrsg.) DeLFI 2007: 5. e-Learning Fachtagung Informatik

P-112 Sigrid Schubert (Hrsg.) Didaktik der Informatik in Theorie und Praxis

P-113 Sören Auer, Christian Bizer, Claudia Müller, Anna V. Zhdanova (Eds.) The Social Semantic Web 2007 Proceedings of the 1st Conference on Social Semantic Web (CSSW)

P-114 Sandra Frings, Oliver Göbel, Detlef Günther, Hardo G. Hase, Jens Nedon, Dirk Schadt, Arslan Brömme (Eds.) IMF2007 IT-incident management & IT-forensics Proceedings of the 3rd International Conference on IT-Incident Management & IT-Forensics

P-115 Claudia Falter, Alexander Schliep, Joachim Selbig, Martin Vingron and Dirk Walther (Eds.) German conference on bioinformatics GCB 2007

P-116 Witold Abramowicz, Leszek Maciszek (Eds.) Business Process and Services Computing 1st International Working Conference on Business Process and Services Computing BPSC 2007

P-117 Ryszard Kowalczyk (Ed.) Grid service engineering and manegement The 4th International Conference on Grid Service Engineering and Management GSEM 2007

P-118 Andreas Hein, Wilfried Thoben, Hans-Jürgen Appelrath, Peter Jensch (Eds.) European Conference on ehealth 2007

P-119 Manfred Reichert, Stefan Strecker, Klaus Turowski (Eds.) Enterprise Modelling and Information Systems Architectures Concepts and Applications

P-120 Adam Pawlak, Kurt Sandkuhl, Wojciech Cholewa, Leandro Soares Indrusiak (Eds.) Coordination of Collaborative Engineering - State of the Art and Future Challenges

P-121 Korbinian Herrmann, Bernd Bruegge (Hrsg.) Software Engineering 2008 Fachtagung des GI-Fachbereichs Softwaretechnik

P-122 Walid Maalej, Bernd Bruegge (Hrsg.) Software Engineering 2008 - Workshopband Fachtagung des GI-Fachbereichs Softwaretechnik

P-123 Michael H. Breitner, Martin Breunig, Elgar Fleisch, Ley Pousttchi, Klaus Turowski (Hrsg.)
Mobile und Ubiquitäre Informationssysteme – Technologien, Prozesse, Marktfähigkeit
Proceedings zur 3. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2008)

P-124 Wolfgang E. Nagel, Rolf Hoffmann, Andreas Koch (Eds.)
9th Workshop on Parallel Systems and Algorithms (PASA)
Workshop of the GI/ITG Speciel Interest Groups PARS and PARVA

P-125 Rolf A.E. Müller, Hans-H. Sundermeier, Ludwig Theuvsen, Stephanie Schütze, Marlies Morgenstern (Hrsg.)
Unternehmens-IT: Führungsinstrument oder Verwaltungsbürde
Referate der 28. GIL Jahrestagung

P-126 Rainer Gimnich, Uwe Kaiser, Jochen Quante, Andreas Winter (Hrsg.)
10th Workshop Software Reengineering (WSR 2008)

P-127 Thomas Kühne, Wolfgang Reisig, Friedrich Steimann (Hrsg.)
Modellierung 2008

P-128 Ammar Alkassar, Jörg Siekmann (Hrsg.)
Sicherheit 2008
Sicherheit, Schutz und Zuverlässigkeit
Beiträge der 4. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI)
2.-4. April 2008
Saarbrücken, Germany

P-129 Wolfgang Hesse, Andreas Oberweis (Eds.)
Sigsand-Europe 2008
Proceedings of the Third AIS SIGSAND European Symposium on Analysis, Design, Use and Societal Impact of Information Systems

P-130 Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)
1. DFN-Forum Kommunikations-technologien Beiträge der Fachtagung

P-131 Robert Krimmer, Rüdiger Grimm (Eds.)
3rd International Conference on Electronic Voting 2008
Co-organized by Council of Europe, Gesellschaft für Informatik and E-Voting.CC

P-132 Silke Seehusen, Ulrike Lucke, Stefan Fischer (Hrsg.)
DeLFI 2008:
Die 6. e-Learning Fachtagung Informatik

P-133 Heinz-Gerd Hegering, Axel Lehmann, Hans Jürgen Ohlbach, Christian Scheideler (Hrsg.)
INFORMATIK 2008
Beherrschbare Systeme – dank Informatik
Band 1

P-134 Heinz-Gerd Hegering, Axel Lehmann, Hans Jürgen Ohlbach, Christian Scheideler (Hrsg.)
INFORMATIK 2008
Beherrschbare Systeme – dank Informatik
Band 2

P-135 Torsten Brinda, Michael Fothe, Peter Hubwieser, Kirsten Schlüter (Hrsg.)
Didaktik der Informatik – Aktuelle Forschungsergebnisse

P-136 Andreas Beyer, Michael Schroeder (Eds.)
German Conference on Bioinformatics GCB 2008

P-137 Arslan Brömme, Christoph Busch, Detlef Hühnlein (Eds.)
BIOSIG 2008: Biometrics and Electronic Signatures

P-138 Barbara Dinter, Robert Winter, Peter Chamoni, Norbert Gronau, Klaus Turowski (Hrsg.)
Synergien durch Integration und Informationslogistik
Proceedings zur DW2008

P-139 Georg Herzwurm, Martin Mikusz (Hrsg.)
Industrialisierung des Software-Managements
Fachtagung des GI-Fachausschusses Management der Anwendungsentwick-lung und -wartung im Fachbereich Wirtschaftsinformatik

P-140 Oliver Göbel, Sandra Frings, Detlef Günther, Jens Nedon, Dirk Schadt (Eds.)
IMF 2008 - IT Incident Management & IT Forensics

P-141 Peter Loos, Markus Nüttgens, Klaus Turowski, Dirk Werth (Hrsg.)
Modellierung betrieblicher Informations-systeme (MobIS 2008)
Modellierung zwischen SOA und Compliance Management

GI-Edition Lecture Notes in Informatics – Seminars

S-1 Johannes Magenheim, Sigrid Schubert (Eds.):
Informatics and Student Assessment
Concepts of Empirical Research and Standardisation of Measurement in the Area of Didactics of Informatics

S-2 Gesellschaft für Informatik (Hrsg.)
Informationstage 2005
Fachwissenschaftlicher Informatik-Kongress

S-3 Gesellschaft für Informatik (Hrsg.)
Informationstage 2006
Fachwissenschaftlicher Informatik-Kongress

S-4 Hans Hagen, Andreas Kerren, Peter Dannenmann (Eds.)
Visualization of Large and Unstructured Data Sets
First workshop of the DFG's International Research Training Group "Visualization of Large and Unstructured Data Sets – Applications in Geospatial Planning, Modeling and Engineering"

S-5 Gesellschaft für Informatik (Hrsg.)
Informationstage 2007
Fachwissenschaftlicher Informatik-Kongress

S-6 Gesellschaft für Informatik (Hrsg.)
Informationstage 2008
Fachwissenschaftlicher Informatik-Kongress

S-7 Hans Hagen, Martin Hering-Bertram, Christoph Garth (Eds.)
Visualization of Large and Unstructured Data Sets
Second Workshop of the DFG´s International Research Training Group "Visualization of Large and Unstructured Data Sets – Applications in Geospatial Planning, Modeling, and Engineering"

The titles can be purchased at:

Köllen Druck + Verlag GmbH
Ernst-Robert-Curtius-Str. 14 · D-53117 Bonn
Fax: +49 (0)228/9898222
E-Mail: druckverlag@koellen.de