

# Massively Parallel Multigrid for the Simulation of Skin Permeation on Anisotropic Tetrakaidecahedral Cell Geometries

Sebastian Reiter, Arne Nägel, Andreas Vogel, and Gabriel Wittum

**Abstract** Numerical simulation based on mathematical models is an important pillar for enhancing the understanding of permeation processes in the skin. To adequately resolve the complex geometrical structure of the skin, special models based on tetrakaidecahedral cells have been suggested. While these models preserve many of the desirable properties of the underlying geometry, they impose challenges regarding mesh generation and solver robustness.

To improve robustness of the used multigrid solver, we propose a new mesh and hierarchy structure with good aspect ratios and angle conditions. Furthermore, we show how those meshes can be used in scalable massively parallel multigrid based computations of permeation processes in the skin.

## 1 Introduction

Mathematical modeling is one important pillar for enhancing the understanding of permeation processes in the skin [1, 2]. The sub-class of microscopic models consider conservation laws and morphology on the cellular scale and combine both to improve the qualitative and quantitative understanding.

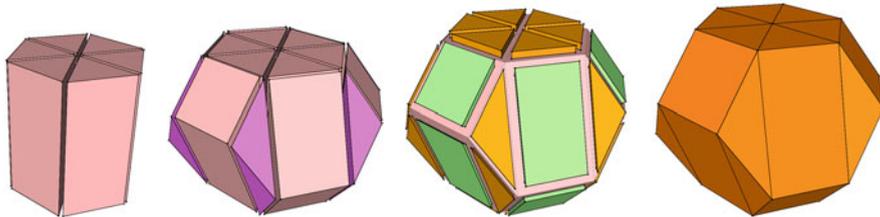
For the epidermis Allen and Potten [3] suggested a morphology model consisting of agglomerated tetrakaidecahedra. A tetrakaidecahedron (*TKD*), depicted in the rightmost figure in Fig. 1, is a polyhedron with 14 faces providing a dense spatial packing. This property has been discovered by Lord Kelvin [4] studying foam cells and makes it an attractive cell template for the construction of idealized biological tissues.

Based on *TKD* shaped cells Feuchter et al. [5] suggested a geometry model for the stratum corneum (SC), i.e., the outermost layer of the epidermis. This region

---

S. Reiter (✉) • A. Nägel • A. Vogel  
G-CSC, Goethe-Universität Frankfurt, Kettenhofweg 139, 60325 Frankfurt (M.), Germany  
e-mail: [sebastian.reiter@gcsc.uni-frankfurt.de](mailto:sebastian.reiter@gcsc.uni-frankfurt.de); [arne.naegel@gcsc.uni-frankfurt.de](mailto:arne.naegel@gcsc.uni-frankfurt.de);  
[andreas.vogel@gcsc.uni-frankfurt.de](mailto:andreas.vogel@gcsc.uni-frankfurt.de)

G. Wittum  
Goethe Center for Scientific Computing (G-CSC), Simulation and Modelling Department,  
Goethe-Universität Frankfurt am Main, Kettenhofweg 139, 60325 Frankfurt am Main, Germany  
e-mail: [wittum@gcsc.uni-frankfurt.de](mailto:wittum@gcsc.uni-frankfurt.de)



**Fig. 1** New simplified meshing of tetrakaidecahedral elements. *Left:* Inner prism structure, *middle left:* full *TKD* mesh with additional prism and tetrahedral elements, *middle right:* Surrounding lipid elements in an explosion visualization, *right:* Surface of a complete building block consisting of the inner *TKD* and the surrounding lipid layer

is believed to be responsible for the barrier property of the skin and has been studied by modeling in [6, 7]. Nowadays, computing interactions in large networks of cells becomes feasible. At the same time, the models do not only focus on the SC only, but extend to the living epidermis [8, 9]. From a computational perspective this is important as the increase in size should be accommodated by a scaling of computational resources. Multigrid solvers were employed before [10], but weak scalability has not been the primary focus of the research in this area.

To fully resolve the highly differing scales between corneocytes and the surrounding lipid layer, very fine grids are required. Considering the complex *TKD* based geometric model and the unstructured nature of the associated grids, massively parallel computers are required to handle the large number of elements and unknowns involved.

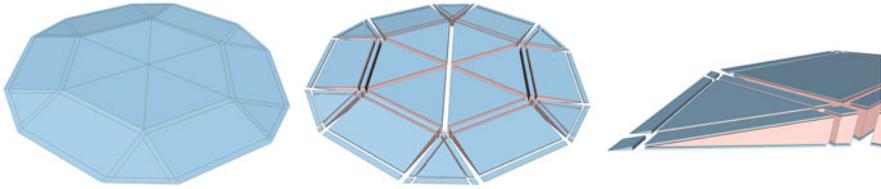
In [11] we presented a massively parallel solver for the problem of drug diffusion through the skin on a much simpler geometric model. In this article, we consider the parallel meshing involved in preparing distributed grid hierarchies for the application of massively parallel multigrid solvers on *TKD* based grids. To this end, we develop an alternative meshing which leads to a very robust and still highly scalable problem setup.

This work is organized as follows: The mathematical model is formulated in Sect. 2, underlying geometries and new meshing approaches are described in Sect. 3. The description of the solver setup is given in Sect. 4 and numerical results are presented in Sect. 5.

## 2 Problem Description

Permeation can be described by the diffusion equation

$$\frac{\partial Ku}{\partial t} + \nabla \cdot [-K \mathbb{D} \nabla u] = 0 \quad (1)$$



**Fig. 2** New meshing of flat tetrakaidecahedral cells with surrounding lipid layer. *Left:* Surface of the new *TKD* meshing, *middle:* Explosion visualization of the new *TKD* meshing, *right:* Element structure near small lipid angles

with suitable initial and boundary conditions. The unknown  $u$  is a normalized reference concentration, and  $K = K(x)$  and  $\mathbb{D} = \mathbb{D}(x)$  are the partition coefficient and diffusion tensor respectively.

Both parameters are position dependent. In general,  $K$  is a function of the chemical potential  $\mu$  (or alternatively the Gibbs energy  $\Delta G := \mu - \mu_0$ ) [12, 13]:

$$K(x) := \exp\left(-\frac{\mu(x) - \mu_0}{RT}\right)$$

Similarly, diffusion depends also on the spatial position  $x \in \mathbb{R}^3$ .

The focus of this work is to demonstrate experimentally, how an efficient multigrid solver can be constructed for this problem. This should be exemplified for the *stratum corneum*, which is the outermost layer of the epidermis, consisting of keratinized cells embedded in a matrix of lipid bilayers. As a consequence the coefficient  $\epsilon(x) := K(x)\mathbb{D}(x)$  is highly variable and jumps between corneocytes and the lipids occur. Moreover, the geometry features a large degree of anisotropy. Both properties enter the discretization and must be resolved by the solver as well.

As a model problem, we consider the equilibrium case with the simplifying assumptions  $K = 1$  and

$$\epsilon(x) = \begin{cases} 1, & x \in \Omega_{LIP} \\ \epsilon_{COR}, & x \in \Omega_{COR} \end{cases} .$$

Here,  $\Omega_{COR}$  and  $\Omega_{LIP}$  correspond to the regions of the corneocytes and the lipids respectively. In Fig. 2 these are shown in light red and light blue respectively.

### 3 Meshing

To create a geometry model of the stratum corneum, we consider so called building blocks consisting of one tetrakaidecahedral cell with one layer of surrounding lipid, following [5]. Stacking those building blocks in the different space directions leads to a mesh which closely resembles the structure of the stratum corneum.

### 3.1 *Previous Mesh Structure*

Feuchter et al. [5, 14] introduced a meshing of a tetrakaidecahedral cell with 3 hexahedral, 36 prism, 6 pyramidal, and 18 tetrahedral elements. Together with the surrounding lipid layer, which was meshed with 18 hexahedral and 36 prism elements, a total of 117 elements per building block was required. The final mesh was then created by shifting copies of this building-block in the different space dimensions.

Besides the large number of elements, the proposed structure led to issues when the cells were flattened, as required to resemble the flat structure of the stratum corneum. When flattened, the elements in the surrounding lipid layer exposed highly obtuse angles. Solvers applied to linearized systems on such grids tend to require large iteration numbers or even diverge completely. Multigrid methods, which are required for scalable parallel computations, need complex smoothers and base solvers to address those issues. Such base solvers and smoothers may not scale well or may even fail completely in massively parallel environments.

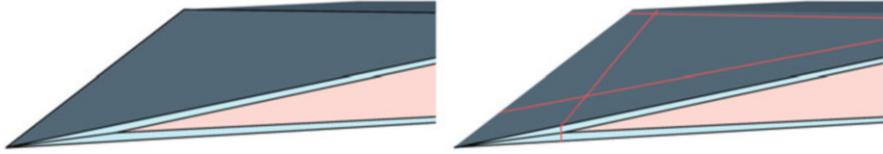
We thus developed an alternative mesh structure which still allows to stack the *TKD* based building blocks in all space directions, yet provides a grid with much better element qualities than previous meshing attempts.

### 3.2 *New Mesh Structure*

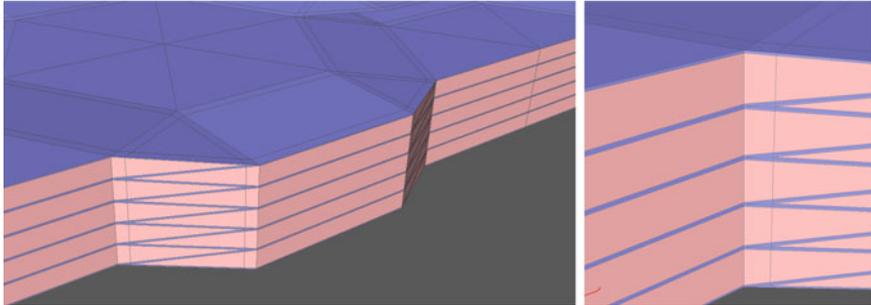
We created the alternative meshing in two steps. First, we created a mesh for a tetrakaidecahedral cell using only 18 elements (12 prisms and 6 tetrahedra) for the corneocyte, and 36 elements (24 prisms and 12 hexahedra) for the surrounding lipid layer, totaling in 54 elements for one building block (cf. Fig. 1).

However, while consisting of less than half of the number of original elements, the new meshing still exhibits the same highly obtuse angles when flattened. We thus further adjusted the mesh structure by subdividing certain elements to remove the obtuse angles. Hereby it was important to subdivide in a way which allows to stack the building blocks in all space directions while guaranteeing mesh consistency. The resulting mesh structure is depicted in Fig. 2. A comparison between the meshing at the outer rim of the old and new mesh structures is given in Fig. 3.

The new mesh structure for a complete building block consists of 6 tetrahedra, 96 prisms, and 72 hexahedra (174 elements in total). No highly obtuse angles are contained in the new mesh anymore. A cut through a mesh resulting from stacking those building blocks is given in Fig. 4. Meshing was performed using the software *ProMesh* (cf. [15]).



**Fig. 3** Comparison of the meshing at the outer rim of the *TKDs*. *Left*: The old meshing exposed highly obtuse angles where now angles close to  $90^\circ$  can be found. *Right*: New meshing with vastly improved angles



**Fig. 4** Cuts through a mesh resulting from stacking *TKDs* with the proposed new mesh structure. *Left*: Overview, *right*: Closeup

### 3.3 Refinement

We are using an anisotropic refinement scheme analogous to [11], where we only refine those edges which are longer than a certain threshold. After each refinement step, we halve the threshold and consequently more edges are considered during refinement. The initial threshold can for example be chosen as  $0.75 \times \text{maxEdgeLen}$ , where *maxEdgeLen* is the length of the longest edge of the given mesh.

We perform five anisotropic refinements followed by further isotropic refinements, depending on the number of intended multigrid levels. With each refinement we introduce a new level in the multigrid hierarchy. During anisotropic refinement element ratios of refined elements are improved compared to their respective parent elements in each consecutive level. Elements which are not refined, since none of their edges are longer than the given threshold, are copied to the next level.

Refinement is performed in parallel and interweaved with parallel redistribution, using the techniques described in [11]. Using this technique, we can efficiently generate large meshes spanning thousands or hundred thousands of processes.

**Table 1** Solver components: Solvers on lower levels serve as base solvers for the preconditioners on higher levels, (cf. [11])

Levels	2	2–4	4–top
Solver	LU (exact)	BiCGStab	BiCGStab
Preconditioner	–	GMG	GMG
Smoother	–	ILU	Jacobi
Cycle	–	V (3,3)	V (3,3)

## 4 Solver Setup

We are using the same solver setup as described in [11]. It consists of two stacked multigrid preconditioned BiCGStab solvers, one serving as base solver for the other. The base-multigrid thereby only spans up to 64 processes and uses an ILU smoother, which works very efficiently on anisotropic geometries on small process numbers. Since the ILU smoother is not required for the isotropic elements encountered in higher levels, we are using a perfectly scalable damped Jacobi smoother for the upper multigrid. The solver setup is shown in Table 1. For details on the highly scalable MPI based parallelization of the underlying multigrid solver for hundreds of thousands of processes, please refer to [16].

## 5 Results

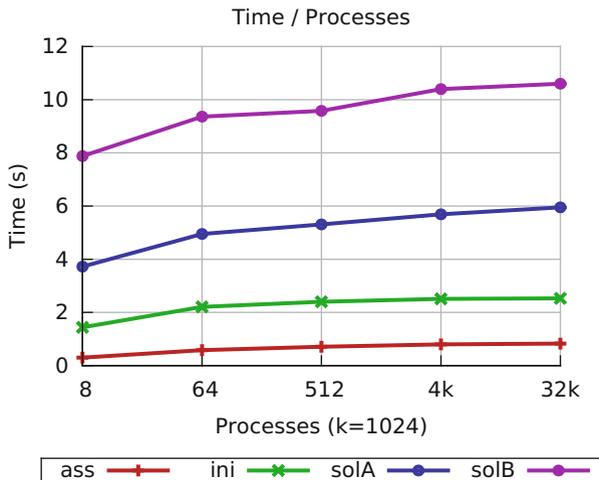
We performed weak scaling studies of the described model problem on the proposed *TKD* based mesh. The computations were executed on the Cray XC40 super computer *Hazel Hen* at the HLRS. *Hazel Hen* has a peak performance of 7420 TFlops. It features 7712 compute nodes which provide 128 GB of memory each. On each node 24 cores are available (up to 48 through hyperthreading).

Runs were performed starting at 8 processes and scaling up to 32,768 processes, increasing the number of processes by a factor of 8 between each run. With each run we also increased the number of elements by refining the grid. However, due to the presence of prism elements, the number of elements grows slightly slower than the number of processes. Between runs 1 and 2 the number of elements grows by a factor of 7.5, between runs 4 and 5 by a factor of 7.93. The workload per process is thus only approximately constant over all runs.

In each run we computed the solution of the steady state of the described model problem. To investigate the robustness of the solver regarding jumps in diffusion coefficients, we performed two studies with different coefficients  $\epsilon_{\text{COR}} = 1$  (study *A*) and  $\epsilon_{\text{COR}} = 0.001$  (study *B*). The distribution of the underlying multigrid hierarchy and the general solver setup was identical in both studies, only the number of iterations required to reach a given defect varied slightly.

**Table 2** Each line corresponds to an individual run. Recorded are the number of processes (*PEs*), the number of levels (*Levels*), the number of unknowns (*DoFs*), the run times of assembly ( $T_{ass}$ ), solver initialization ( $T_{ini}$ ), and solving in study A ( $T_{sol}^A$ ), and study B ( $T_{sol}^B$ ) in seconds

PEs	Levels	DoFs	$T_{ass}$	$T_{ini}$	$T_{sol}^A$	$T_{sol}^B$
8	5	374,223	0.30	1.44	3.72	7.88
64	6	2,807,351	0.58	2.21	4.95	9.36
512	7	21,703,869	0.71	2.40	5.31	9.58
4096	8	170,592,761	0.80	2.51	5.69	10.4
32,768	9	1,352,552,433	0.83	2.53	5.95	10.6



**Fig. 5** Scaling of the run times of assembly (*ass*), solver initialization (*ini*), and solving of study A (*solA*), and study B (*solB*)

The number of CG-iterations were fixed to 10 in each run of study A and to 14 in each run of study B to allow for a good comparability of the timings. In each run we achieved a reduction of the initial defect by a factor of  $10^{-5}$ .

The number of unknowns and the timings of the assembly and solution process in each run are given in Table 2. Since assembly and solver initialization are mostly identical in studies A and B, we only show the different timings in the actual solving stage  $T_{sol}$ . The scaling behavior of assembly, solver initialization, and solving is also shown in Fig. 5.

In Table 3 the levelwise distribution qualities for each run are shown. The distribution quality  $q_l$  of a level  $l$  of the hierarchy is computed as

$$q_l := \frac{n_l^{total} - n_l^{max}}{n_l^{max} \cdot (P_l - 1)},$$

**Table 3** Distribution qualities for each level of the multigrid hierarchy for the different runs

PE	lv-0	lv-1	lv-2	lv-3	lv-4	lv-5	lv-6	lv-7	lv-8	lv-9
8	1	1	1	0.96	0.97	0.99	—	—	—	—
64	1	1	1	0.59	0.81	0.89	0.88	—	—	—
512	1	1	1	0.71	0.96	0.86	0.85	0.84	—	—
4096	1	1	1	0.71	0.96	0.85	0.84	0.82	0.80	—
32,768	1	1	1	0.71	0.96	0.87	0.89	0.85	0.84	0.82

**Table 4** Number of processes used on each level for the individual runs

PE	lv-0	lv-1	lv-2	lv-3	lv-4	lv-5	lv-6	lv-7	lv-8	lv-9
8	1	1	1	8	8	8	—	—	—	—
64	1	1	1	64	64	64	64	—	—	—
512	1	1	1	64	64	512	512	512	—	—
4096	1	1	1	64	64	4096	4096	4096	4096	—
32,768	1	1	1	64	64	4096	4096	32,768	32,768	32,768

where  $P_l > 1$  is the number of processes of the given process-hierarchy on level  $l$ ,  $n_l^p$  is the number of elements in level  $l$  on process  $p$ , and

$$n_l^{total} := \sum_{p=1}^{P_l} n_l^p,$$

$$n_l^{max} := \max_{p=1, \dots, P_l} n_l^p.$$

For  $P_l = 1$  numerator and denominator both vanish and we define  $q_l = 1$ .  $q_l$  is thus in the range  $[0, 1]$ , where  $q_l = 0$  means that all elements of level  $l$  are contained on one process only and  $q_l = 1$  reflects an equal share of elements among all processes.

Since we are using a hierarchical distribution scheme (cf. [16]), not all processes are contributing on all levels. Table 4 gives the number of involved processes on each level of the multigrid hierarchy for each run.

Runtimes of the solver are higher in study *B* compared to study *A*. Besides the fact that more solver iterations are performed in study *B*, the problem is now more difficult to solve due to the jumping coefficients. Due to this, the iteration numbers of the base solver increase, which affects the overall performance. Nevertheless, runtimes and scalability can be considered good for both studies.

While the first run (8 PEs) is contained on one node only, subsequent runs on 64 and more PEs are distributed onto several nodes. The overhead introduced through message passing between nodes is already seen in the jump in runtimes between the first two runs.

Given the highly scalable solver setup (cf. [16] and [11]), the biggest issues regarding scalability are to be expected from load imbalances. Indeed, correlating Table 3 and timings in Table 2, we see that imbalances on higher levels affect

the efficiency on large process numbers. Still, overall efficiency is very good considering the unstructured nature of the problem.

While improvements regarding load balancing of unstructured multigrid hierarchies on massively parallel architectures could further improve scalability of the presented simulation, the high efficiency and robustness of the current approach is more than sufficient to perform large scale computations of complex biological problems on massively parallel computers.

**Acknowledgements** This work has been supported by the DFG in the *German Priority Programme 1648—Software for Exascale Computing* in the project *Exasolvers* (WI 1037/24-2) and by the *German Ministry of Economics and Technology (BMWi)* (02E11476B). We thank the HLRS for the opportunity to use *Hazel Hen* and their kind support.

## References

1. S. Mitragotri, Y.G. Anissimov, A.L. Bunge, H.F. Frasch, J. Hadgraft, R.H. Guy, G.B. Kasting, M.E. Lane, M.S. Roberts, Mathematical models of skin permeability: an overview. *Int. J. Pharm.* **418**, 115–129 (2011)
2. A. Naegel, M. Heisig, G. Wittum, Detailed modeling of skin penetration - an overview. *Adv. Drug Deliv. Rev.* **65**(2), 191–207 (2013)
3. T.D. Allen, C.S. Potten, Significance of cell shape in tissue architecture. *Nature* **264**(5586), 545–547 (1976)
4. W. Thomson, On the division of space with minimum partitional area. *Philos. Mag.* **24**, 503 (1887)
5. D. Feuchter, M. Heisig, G. Wittum, A geometry model for the simulation of drug diffusion through the stratum corneum. *Comput. Vis. Sci.* **9**, 117–130 (2006)
6. A. Nägel, M. Heisig, G. Wittum, A comparison of two- and three-dimensional models for the simulation of the permeability of human stratum corneum. *Eur. J. Pharm. Biopharm.* **72**(2), 332–338 (2009)
7. I. Muha, A. Naegel, S. Stichel, A. Grillo, M. Heisig, G. Wittum, Effective diffusivity in membranes with tetrakaidecahedral cells and implications for the permeability of human stratum corneum. *J. Membr. Sci.* **368**, 18–25 (2010)
8. M. Yokouchi, T. Atsugi, M. Van Logtestijn, R.J. Tanaka, M. Kajimura, M. Suematsu, M. Furuse, M. Amagai, A. Kubo, Epidermal cell turnover across tight junctions based on Kelvin's tetrakaidecahedron cell shape. *eLife* **5**:e19593 (2016). <https://doi.org/10.7554/eLife.19593>
9. R. Wittum, A. Naegel, M. Heisig, G. Wittum, Mathematical modelling of the viable epidermis: impact of cell shape and vertical arrangement (2017, submitted)
10. A. Nägel, *Schnelle Löser für große Gleichungssysteme mit Anwendungen in der Biophysik und den Lebenswissenschaften*, PhD thesis, Universität Heidelberg, 2010
11. S. Reiter, A. Vogel, A. Nägel, G. Wittum, A massively parallel multigrid method with level dependent smoothers for problems with high anisotropies, in *High Performance Computing in Science and Engineering '16* (Springer, Cham, 2016), pp. 667–675
12. Y.G. Anissimov, M.S. Roberts, Diffusion modeling of percutaneous absorption kinetics: 3. Variable diffusion and partition coefficients, consequences for stratum corneum depth profiles and desorption kinetics. *J. Pharm. Sci.* **93**(2), 470–487 (2004)
13. R. Schulz, K. Yamamoto, A. Klossek, R. Flesch, S. Hönzke, F. Rancan, A. Vogt, U. Blume-Peytavi, S. Hedtrich, M. Schäfer-Korting et al., Data-based modeling of drug penetration relates human skin barrier function to the interplay of diffusivity and free-energy profiles. *Proc. Natl. Acad. Sci.* **114**(14), 3631–3636 (2017)

14. D. Fechter, Geometrie- und Gittererzeugung für anisotrope Schichtengebiete, PhD thesis, Universität Heidelberg, 2008
15. Promesh, <http://www.promesh3d.com>
16. S. Reiter, A. Vogel, I. Heppner, M. Rupp, G. Wittum, A massively parallel geometric multigrid solver on hierarchically distributed grids. *Comput. Vis. Sci.* **16**(4), 151–164 (2013)