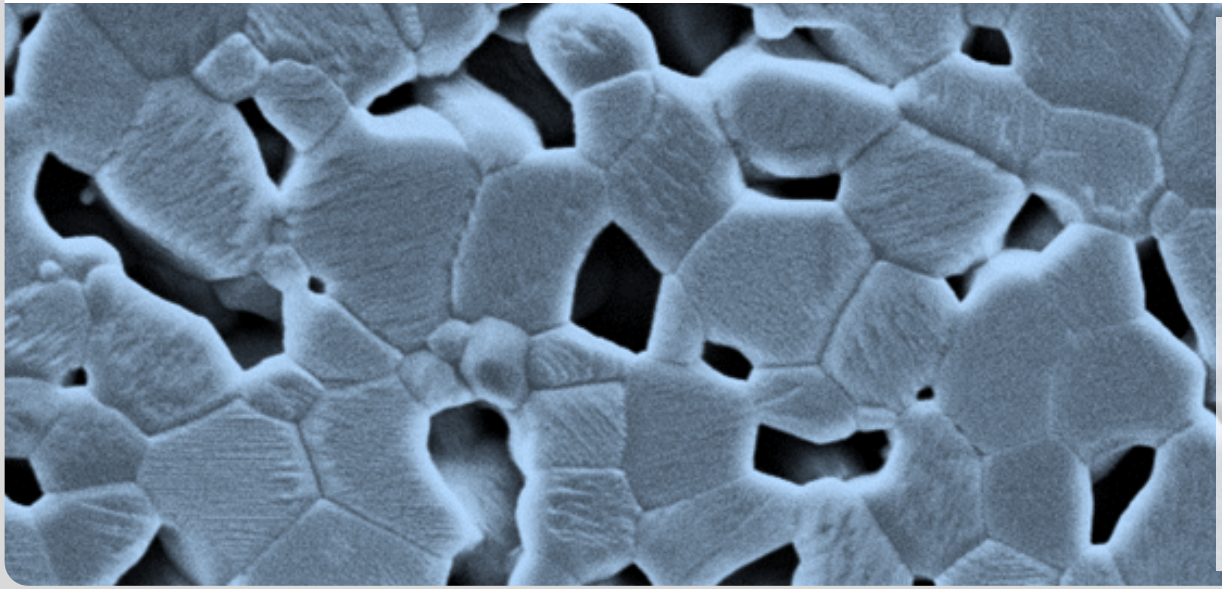
Karlsruhe Institute of Technology

Hochschule Karlsruhe
Technik und Wirtschaft
UNIVERSITY OF APPLIED SCIENCES

An Optimized and Parallelized Phase-Field Solver for Sintering Using the PACE3D Framework

J. Hötzer, H. Hierl, M. Seiz, C. Seer, M. Kellner and B. Nestler

INSTITUTE OF APPLIED MATERIALS - COMPUTATIONAL MATERIALS SCIENCE



Contents:

- Motivation for sintering
- Phase-field model
- Code Optimization
- Performance results
- Simulation results

Part I: Institute

http://zkm.de/en/event/2015/06/schlosslichtspiele

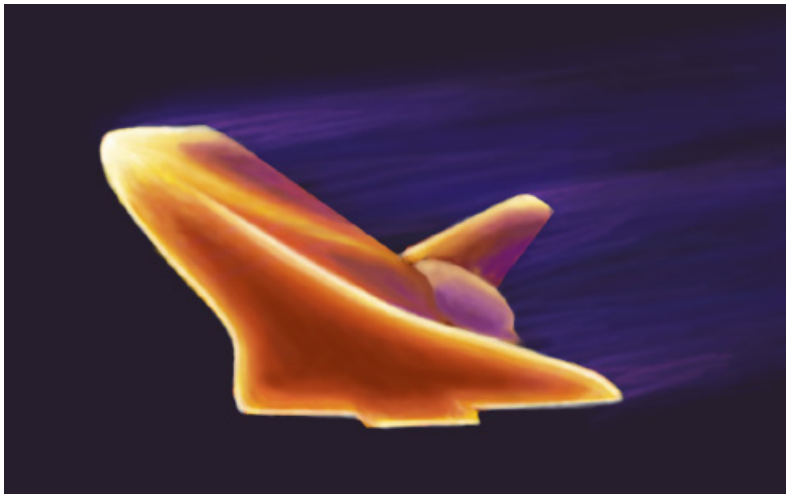
The Group High Performance Materials Computing and Data Science

Part II: Motivation

Applications of ceramics

- everyday items e.g.
  - plates, cups
  - ...

  are “simple” to produce
- high performance materials e.g
  - sensors (e.g oxygen)
  - spark plugs
  - batteries
  - electronics
  - heat shields
  - ...

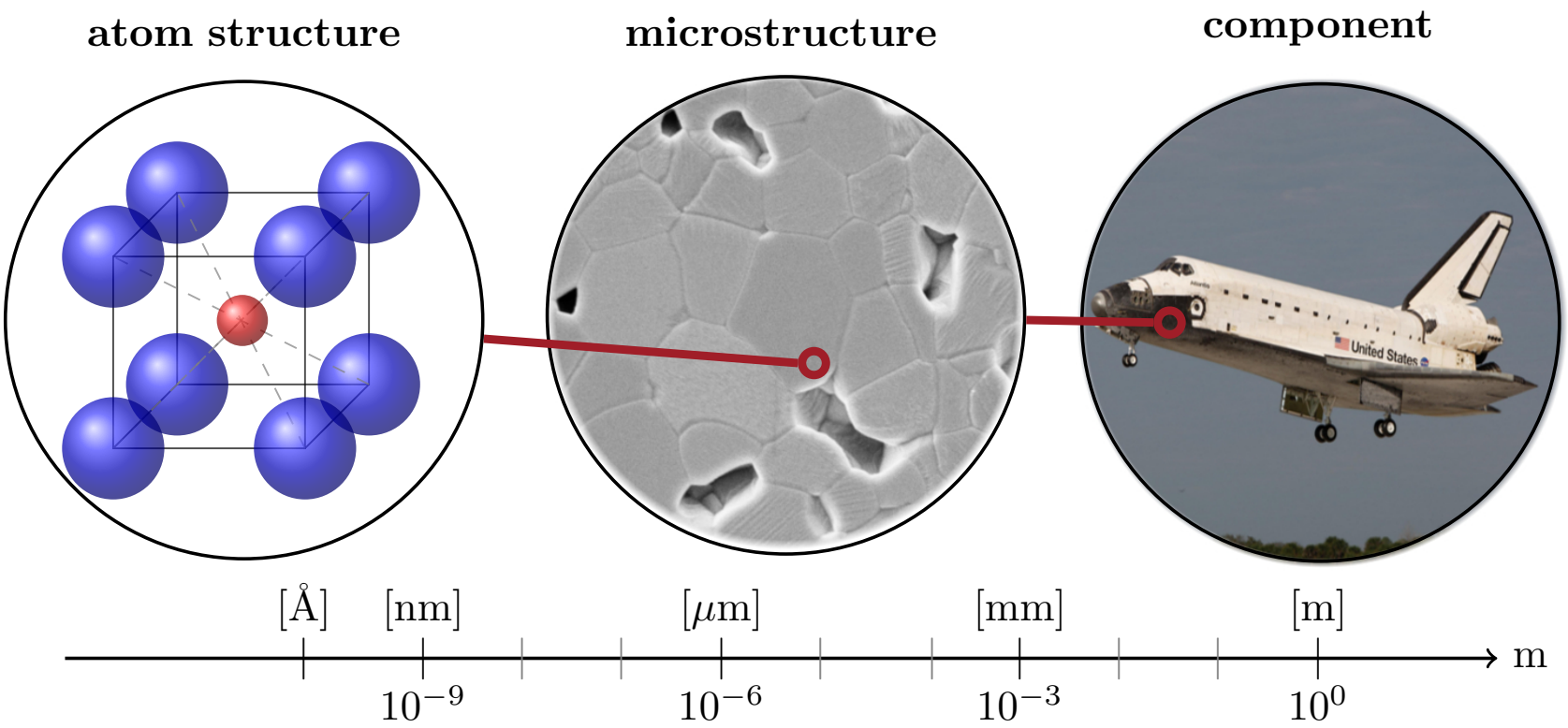


⇒ require defined properties

⇒ most common manufacturing process: **sintering**

https://upload.wikimedia.org/wikipedia/commons/f/fd/Stsheat.jpg
https://upload.wikimedia.org/wikipedia/commons/6/6a/Sparkplug.jpg

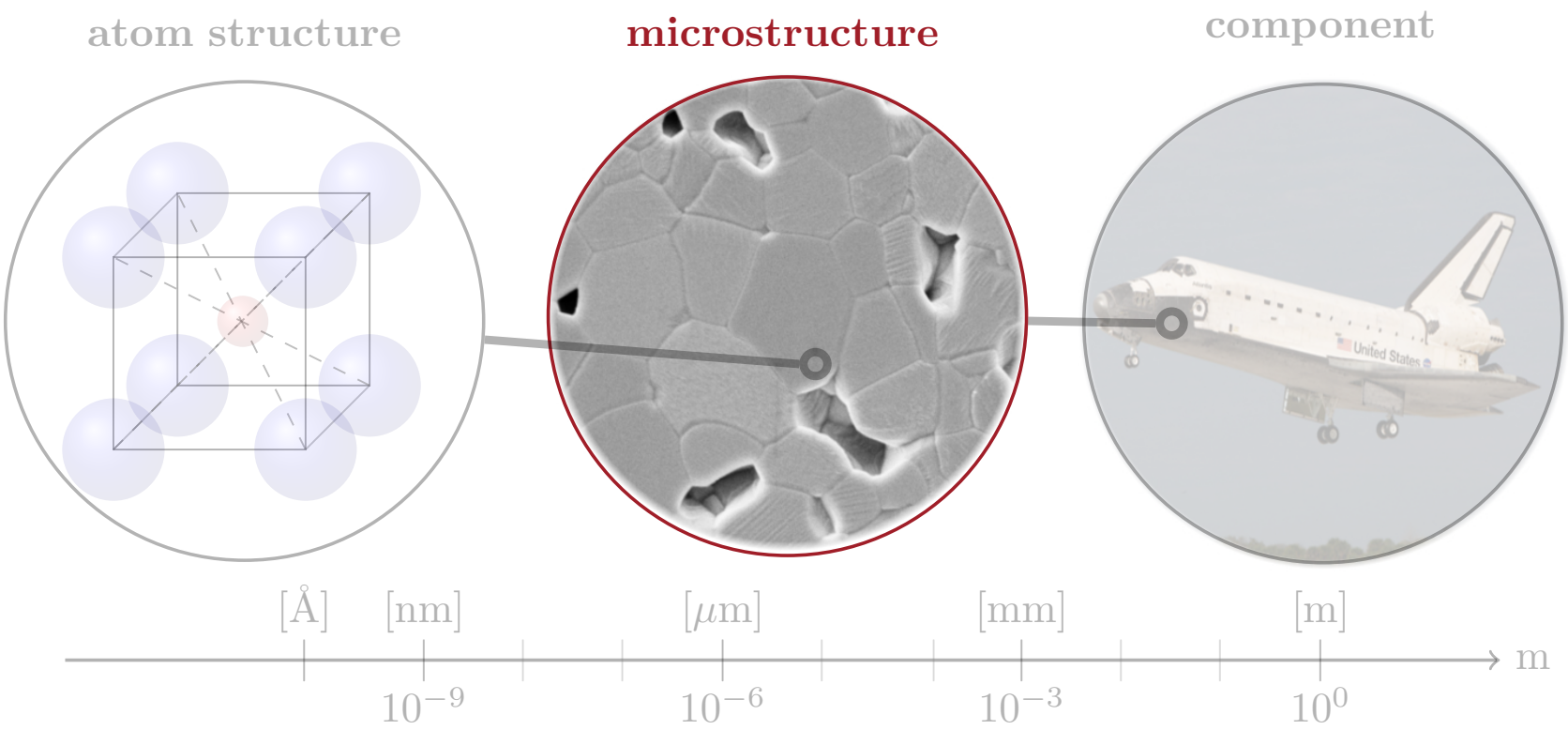
Motivation - Material properties



- **chemical elements** and **microstructure** determine **properties** of the component

→ controlling the **microstructure evolution** = controlling the **material properties**

Motivation - Material properties

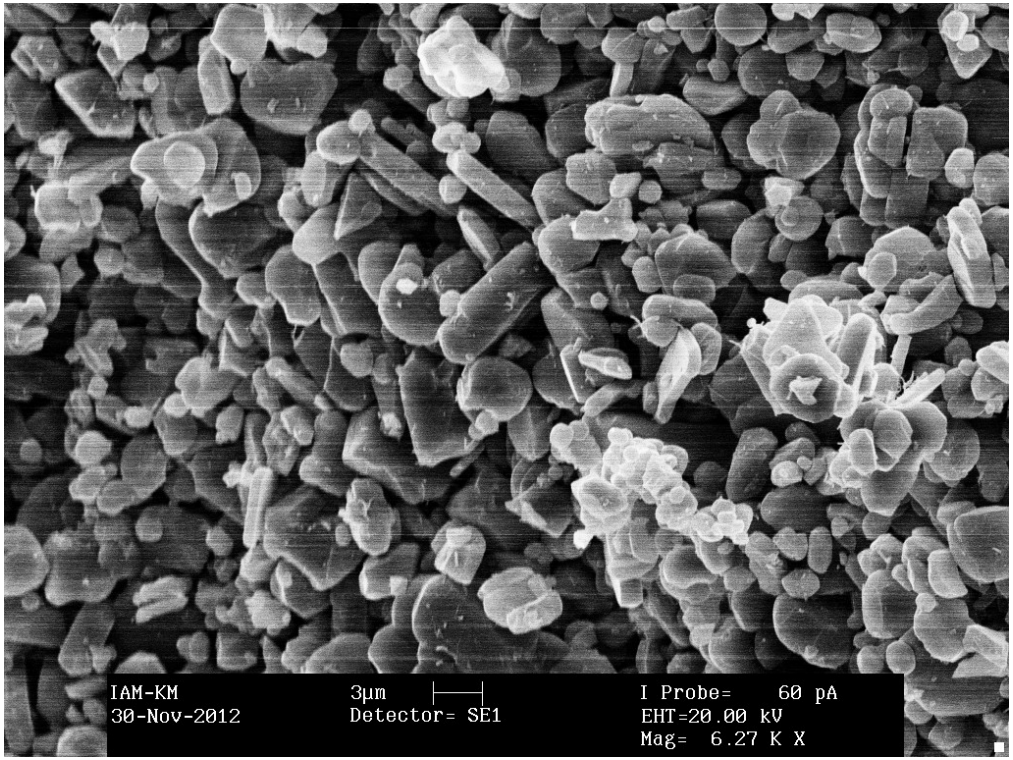


- **chemical elements** and **microstructure** determine **properties** of the component

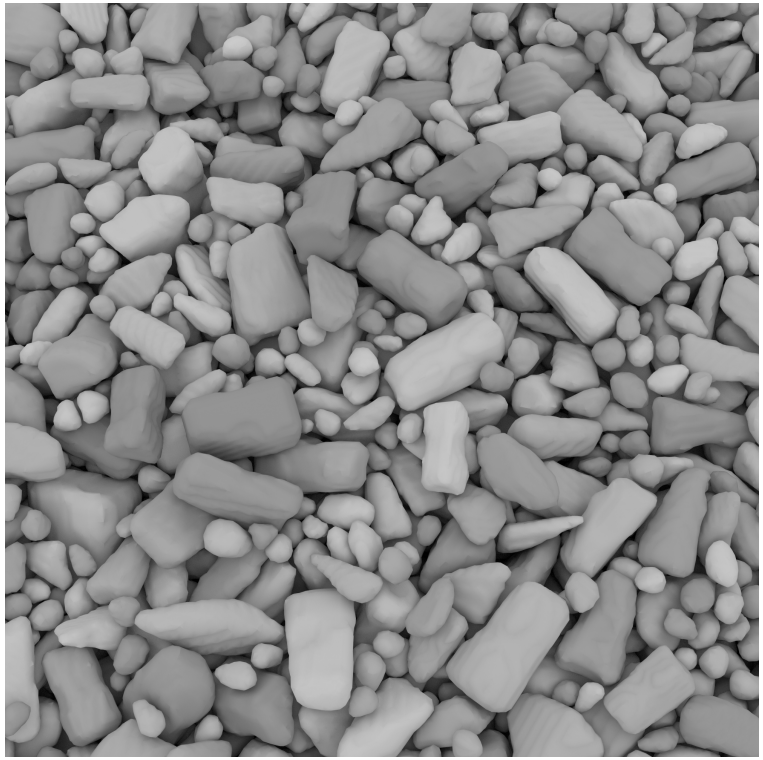
⟶ controlling the **microstructure evolution** = controlling the **material properties**

Initial structure - Green body

- experimental green body

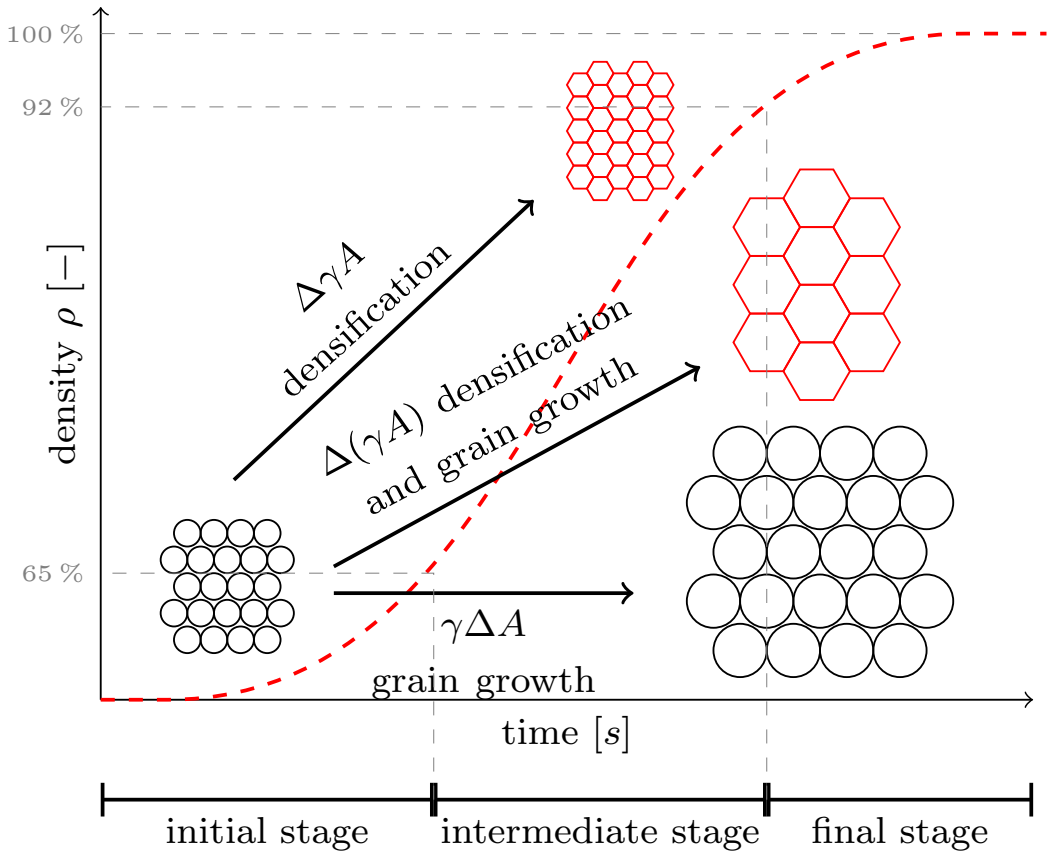


- generated green body



experimental image: Fabian Lemke - IAM

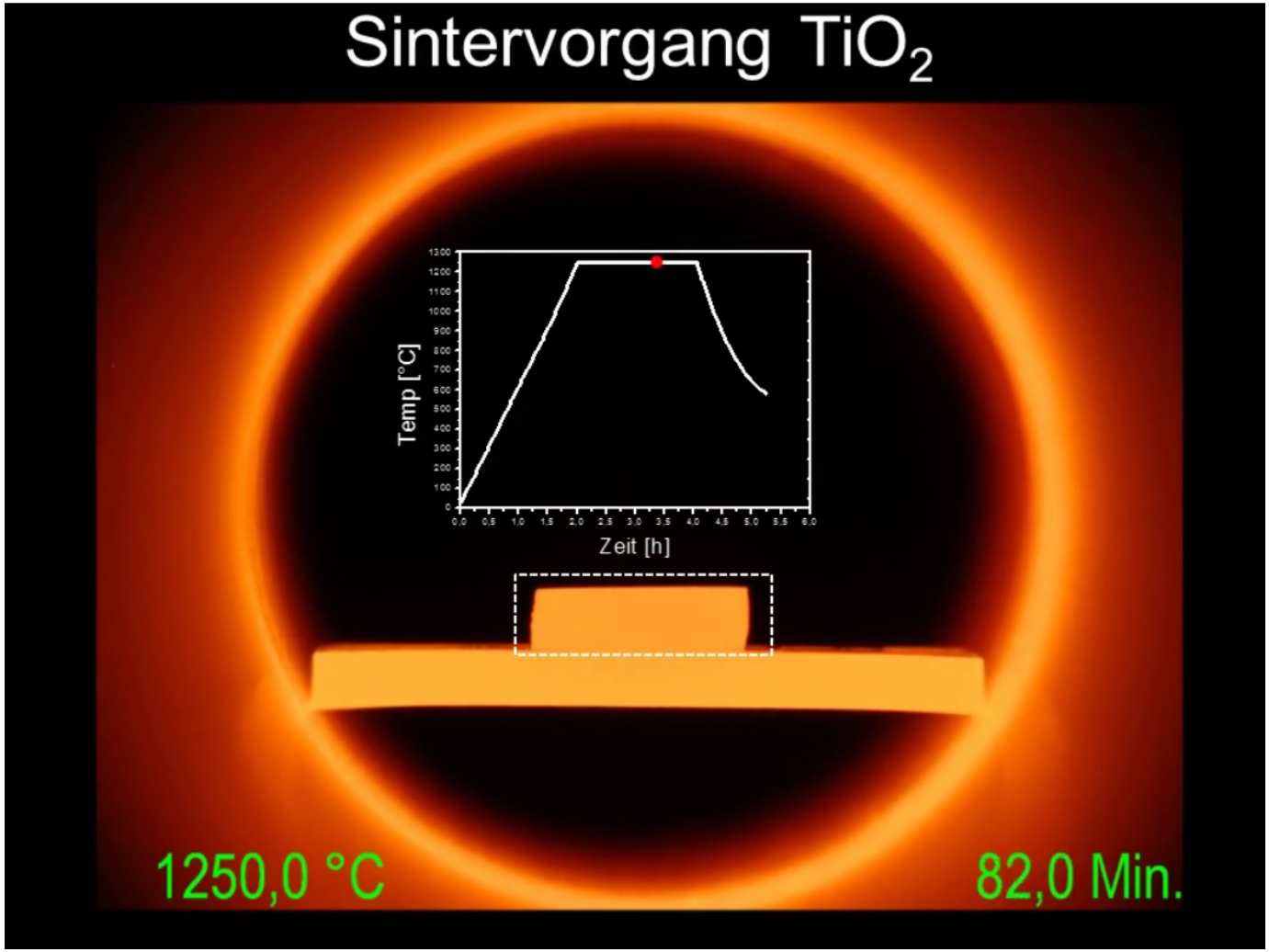
Sintering process



- three stages of sintering:
  - initial sintering stage: building of sintering necks
  - intermediate sintering stage: densification of green body
  - final sintering stage: grain growth

Coble, R.L. (1961). J. Appl. Phys., 32 (5) 787-792

Sintering process of green body

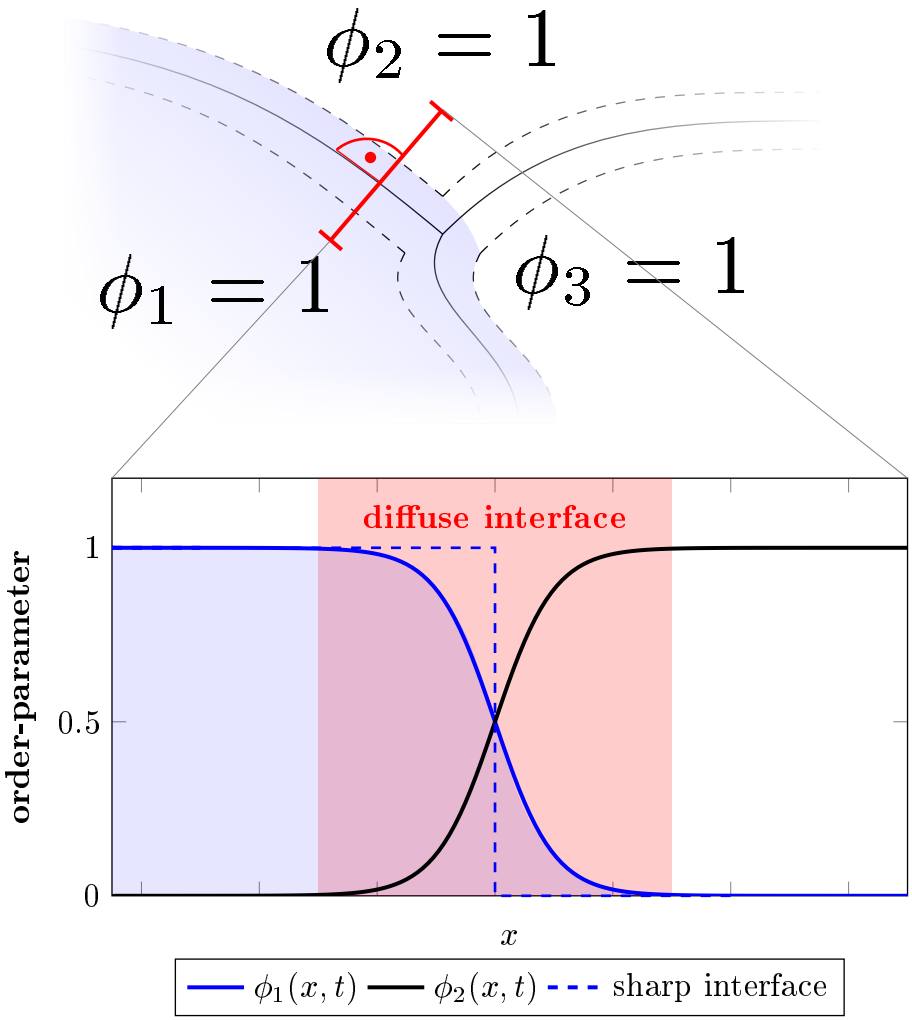


⟶ after exceeding certain temperature **densification** starts

Part III: Phase-Field Model for sintering

Phase-field method

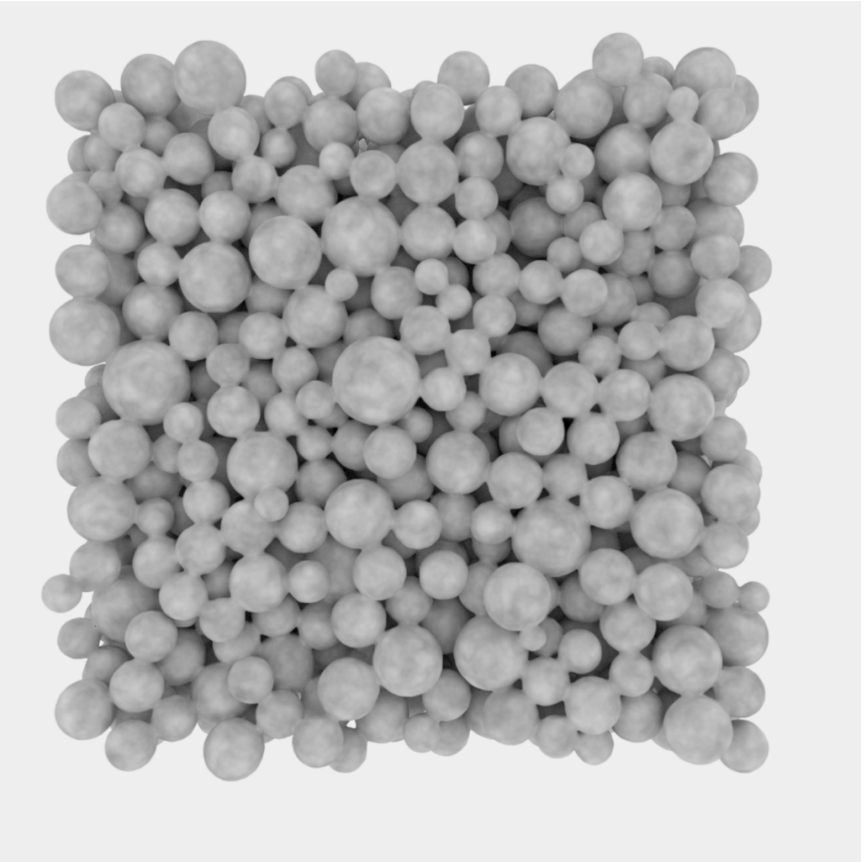
- **minimization** of the **total energy** in a system under constrains
- **coupling** with different **physical quantities** (e.g. concentration, temperature, ...)
- **phase-field vector** $\phi = (\phi_1, \phi_2, ..., \phi_N)^T$
- **order-parameter** ϕ_α describes the volume fraction of the phases with different properties, e.g. orientation, ...
- modeling of a defined **diffuse interface** between the phases
  - → for numerical resolution $\approx$ 10 cells required
  - → high resolution of the structures required



Phase-field model

- **solid state sintering**
- **phase-field model** based on Grand potential approach
  - evolution of phase-fields $\partial_t \phi_\alpha$
  - evolution of chemical potentials $\partial_t \boldsymbol{\mu}$
  - constant temperature gradient
- different **diffusion mechanisms**: surface, grain boundary and volume diffusion
- **multiple** thousand **phases/grains**
- **optimized** on different levels
- massiv-parallele PACE3D framework

$\longrightarrow$ **efficient simulation** of large domains



Phase-field model

- **solid state sintering**
- **phase-field model** based on Grand potential approach
  - evolution of phase-fields $\partial_t \phi_\alpha$
  - evolution of chemical potentials $\partial_t \boldsymbol{\mu}$
  - constant temperature gradient
- different **diffusion mechanisms**: surface, grain boundary and volume diffusion
- **multiple** thousand **phases/grains**
- **optimized** on different levels
- massiv-parallele PACE3D framework

$\longrightarrow$ **efficient simulation** of large domains

$$\tau\varepsilon\frac{\partial\phi_\alpha}{\partial t} = \underbrace{-\varepsilon\left(\frac{\partial a(\boldsymbol{\phi},\nabla\boldsymbol{\phi})}{\partial\phi_\alpha} - \nabla\cdot\frac{\partial a(\boldsymbol{\phi},\nabla\boldsymbol{\phi})}{\partial\nabla\phi_\alpha}\right) - \frac{1}{\varepsilon}\frac{\partial\omega(\boldsymbol{\phi})}{\partial\phi_\alpha}}_{\text{interface contribution } := \text{rhsl}_\alpha}$$

$$\underbrace{-\frac{\partial\psi(\boldsymbol{\phi},\boldsymbol{\mu},T)}{\partial\phi_\alpha}}_{\text{driving force } := \text{rhsf}_\alpha} - \frac{1}{N}\sum_{\beta=1}^{N}(\text{rhsl}_\beta + \text{rhsf}_\beta)$$

Phase-field model

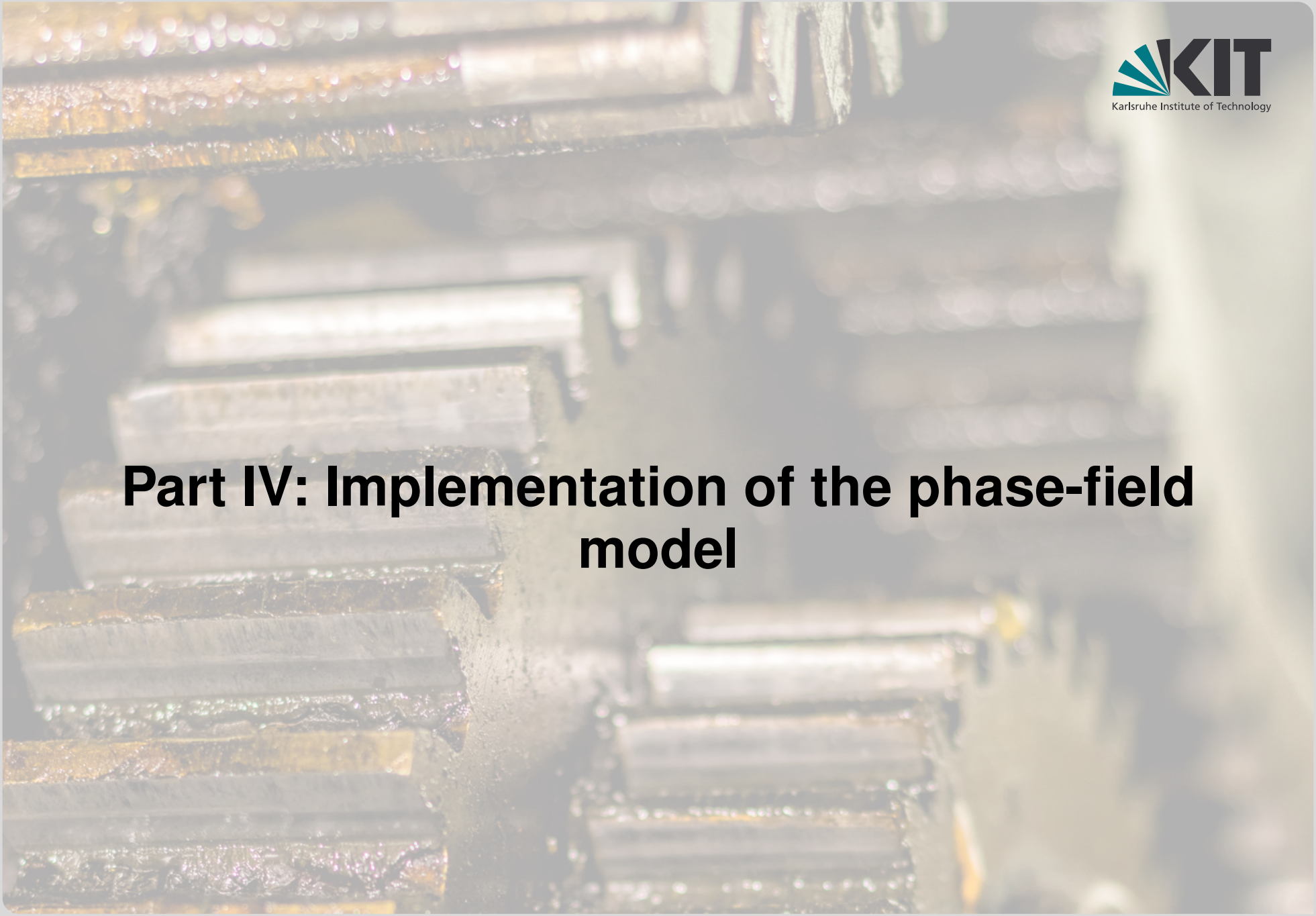
- **solid state sintering**
- **phase-field model** based on Grand potential approach
  - evolution of phase-fields $\partial_t \phi_\alpha$
  - evolution of chemical potentials $\partial_t \boldsymbol{\mu}$
  - constant temperature gradient
- different **diffusion mechanisms**: surface, grain boundary and volume diffusion
- **multiple** thousand **phases/grains**
- **optimized** on different levels
- massiv-parallele PACE3D framework

⟶ **efficient simulation** of large domains

$$\tau\varepsilon\frac{\partial\phi_\alpha}{\partial t} = \underbrace{-\varepsilon\left(\frac{\partial a(\boldsymbol{\phi},\nabla\boldsymbol{\phi})}{\partial\phi_\alpha} - \nabla\cdot\frac{\partial a(\boldsymbol{\phi},\nabla\boldsymbol{\phi})}{\partial\nabla\phi_\alpha}\right) - \frac{1}{\varepsilon}\frac{\partial\omega(\boldsymbol{\phi})}{\partial\phi_\alpha}}_{\text{interface contribution } := \text{rhsl}_\alpha}$$

$$\underbrace{-\frac{\partial\psi(\boldsymbol{\phi},\boldsymbol{\mu},T)}{\partial\phi_\alpha}}_{\text{driving force } := \text{rhsf}_\alpha} - \frac{1}{N}\sum_{\beta=1}^{N}(\text{rhsl}_\beta + \text{rhsf}_\beta)$$

$$\frac{\partial\boldsymbol{\mu}}{\partial t} = \left[\sum_{\alpha=1}^{N} h_\alpha(\vec{\phi})\left(\frac{\partial\vec{c}_\alpha(\boldsymbol{\mu},T)}{\partial\boldsymbol{\mu}}\right)\right]^{-1} \left(\nabla\cdot\left(\boldsymbol{M}(\vec{\phi},\boldsymbol{\mu},T)\nabla\boldsymbol{\mu}\right) - \sum_{\alpha=1}^{N}\vec{c}_\alpha\frac{\partial h_\alpha(\boldsymbol{\phi})}{\partial t} - \sum_{\alpha=1}^{N} h_\alpha(\vec{\phi})\left(\frac{\partial\vec{c}_\alpha}{\partial T}\right)\frac{\partial T}{\partial t}\right)$$



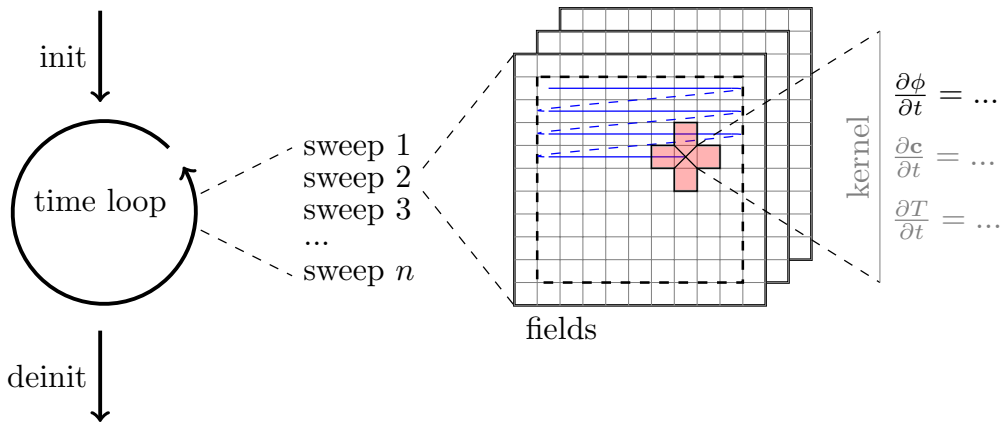
Part IV: Implementation of the phase-field model

Phase-field algorithm

- lattice fields
  - two AoS for phase-field (ϕ_{src}, ϕ_{dst})
  - two SoA for chemical potential (μ_{src}, μ_{dst})
  - communication hiding for both sweeps
- storing new values calculated from *src* in *dst*

Algorithm 1 time step with overlapping communication

1: END: COMMUNICATION (ϕ_{src})
2: $\phi_{dst} \leftarrow \phi$-**sweep**$\left(\phi_{src}, \mu_{src}\right)$
3: ϕ_{dst}-BOUNDARY CONDITIONS
4: START: COMMUNICATION (ϕ_{dst})
5: END: COMMUNICATION (μ_{src})
6: $\mu_{dst} \leftarrow \mu$-**sweep**$\left(\mu_{src}, \phi_{src}, \phi_{dst}\right)$
7: μ_{dst}-BOUNDARY CONDITIONS
8: START: COMMUNICATION (μ_{dst})
9: SWAP $\phi_{src} \leftrightarrow \phi_{dst}$ and $\mu_{src} \leftrightarrow \mu_{dst}$



ϕ-kernel

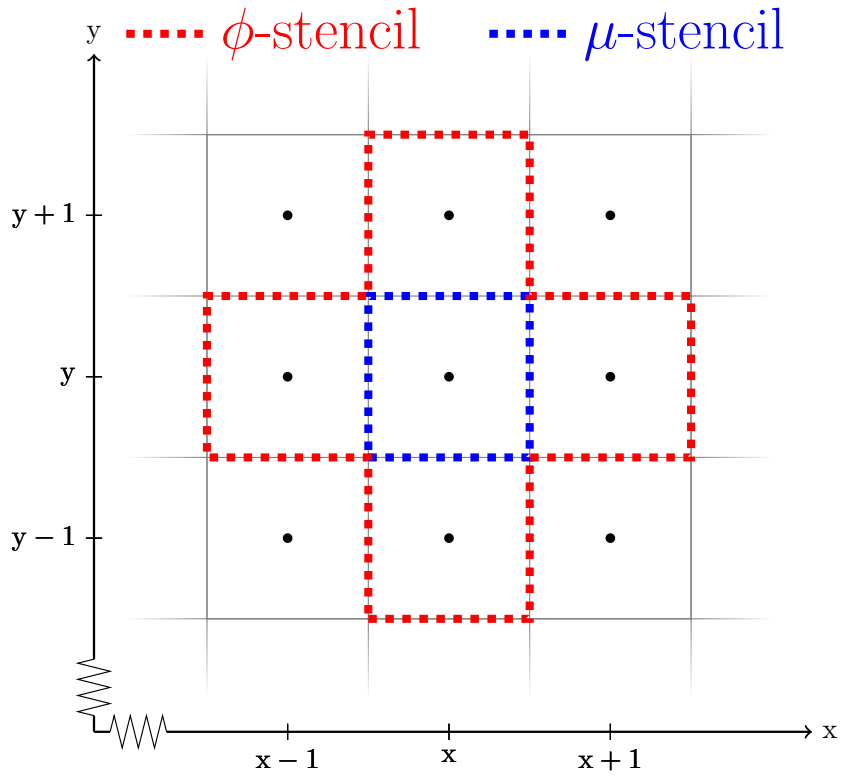
$$\tau\varepsilon\frac{\partial\phi_\alpha}{\partial t} = \underbrace{-\varepsilon\left(\frac{\partial a(\boldsymbol{\phi},\nabla\boldsymbol{\phi})}{\partial\phi_\alpha} + \nabla\cdot\frac{\partial a(\boldsymbol{\phi},\nabla\boldsymbol{\phi})}{\partial\nabla\phi_\alpha}\right)}_{\text{D3C7}} \underbrace{-\frac{1}{\varepsilon}\frac{\partial\omega(\boldsymbol{\phi})}{\partial\phi_\alpha} - \frac{\partial\psi(\boldsymbol{\phi},\boldsymbol{\mu})}{\partial\phi_\alpha} + \lambda}_{\text{D3C1}}$$

- finite differences scheme for space
- explicit Euler scheme for the time discretization
- roofline performance model:

FLOP/*cell* (likwid)	665
loads & stores	168 byte/cell

→ probably **compute bound**

$$\phi(\mathbf{x},t) \xrightarrow{\text{D3C7}} \phi(\mathbf{x},t+\Delta t)$$
$$\mu(\mathbf{x},t) \xrightarrow{\text{D3C1}} \phi(\mathbf{x},t+\Delta t)$$



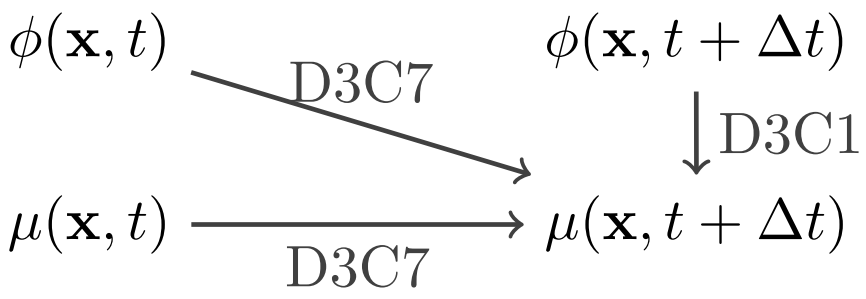
μ-kernel

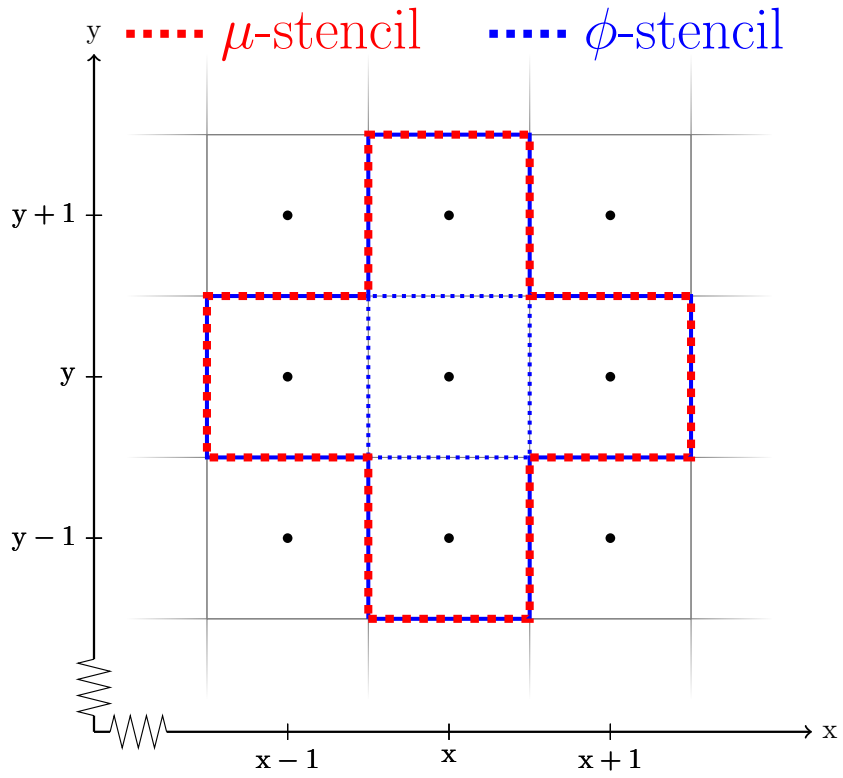
$$\frac{\partial \boldsymbol{\mu}}{\partial t} = \underbrace{\left[\sum_{\alpha=1}^{N} h_\alpha(\vec{\phi}) \left(\frac{\partial \vec{c}^\alpha(\boldsymbol{\mu})}{\partial \boldsymbol{\mu}}\right)\right]^{-1}}_{\text{D3C1}} \left(\underbrace{\nabla \cdot \left(\boldsymbol{M}(\vec{\phi}, \boldsymbol{\mu}) \nabla \boldsymbol{\mu}\right)}_{\text{D3C7}} - \underbrace{\sum_{\alpha=1}^{N} \vec{c}^\alpha(\boldsymbol{\mu}) \frac{\partial h_\alpha(\boldsymbol{\phi})}{\partial t}}_{\text{D3C1}} \right)$$

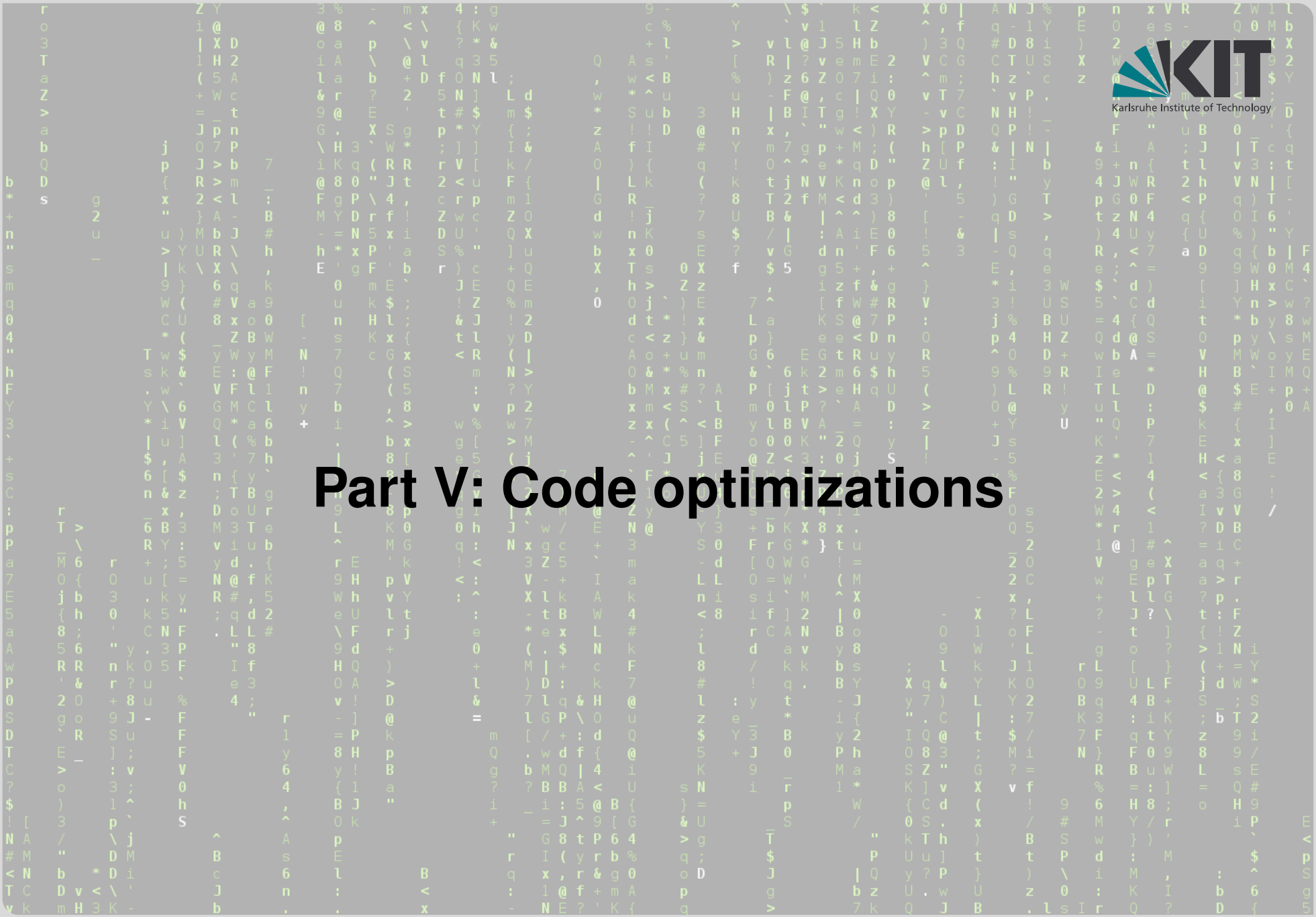
- finite differences scheme for space
- explicit Euler scheme for the time discretization
- roofline performance model:

FLOP/*cell* (likwid)	576
loads & stores	360 byte/cell

⟶ probably **memory bound**





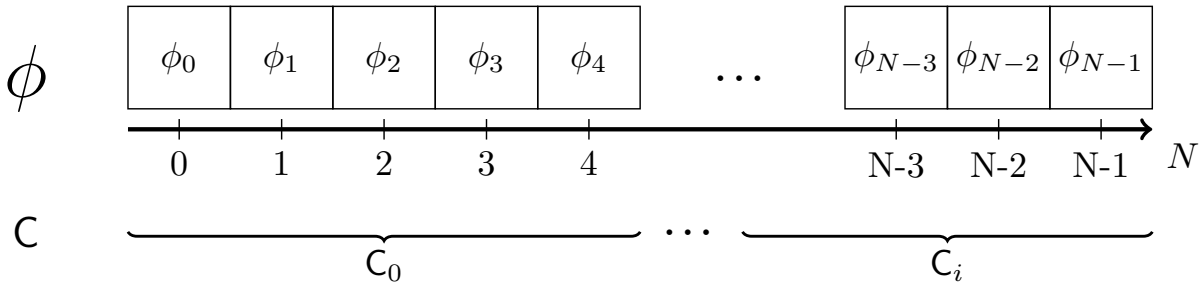


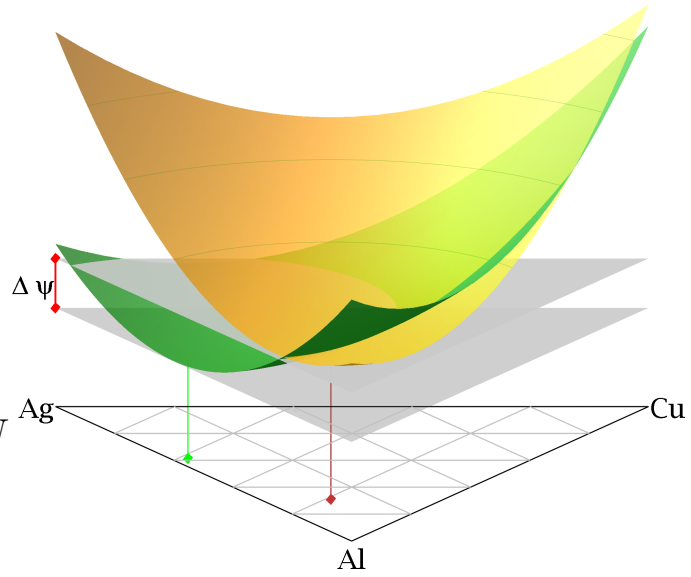
Part V: Code optimizations

Optimizations I

Parameter layer

- fitting of Gibbs energies from CALPHAD databases with parabolic approach to calculate the driving forces
- reduction of the parameter matrices with the size $N \times N$ and $N \times N \times N$ to a class based concept of 2×2 and $2 \times 2 \times 2$

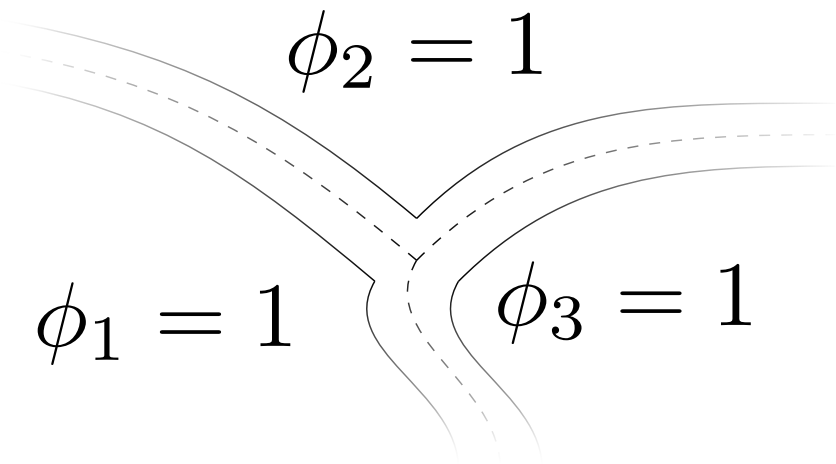




Optimizations II

Model layer

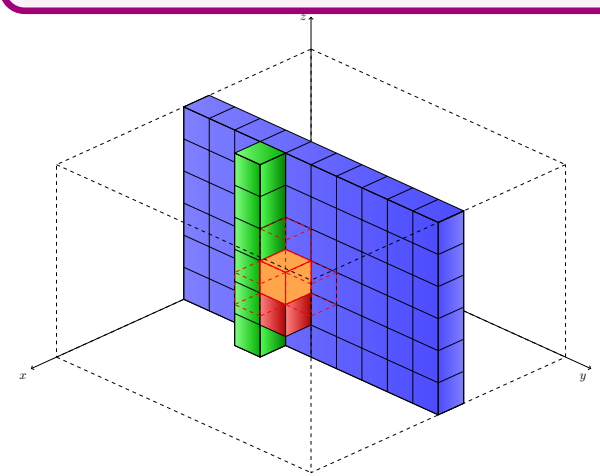
- simplifications due to defined setup
  (e.g. fix number of concentrations)
- classification of cells $\longrightarrow$skip terms
  $\partial_t \phi = \ldots$ needs only calculated in the diffuse interface
- scalar mobility function instead of tensor mobility
- reformulation of terms for better vectorization

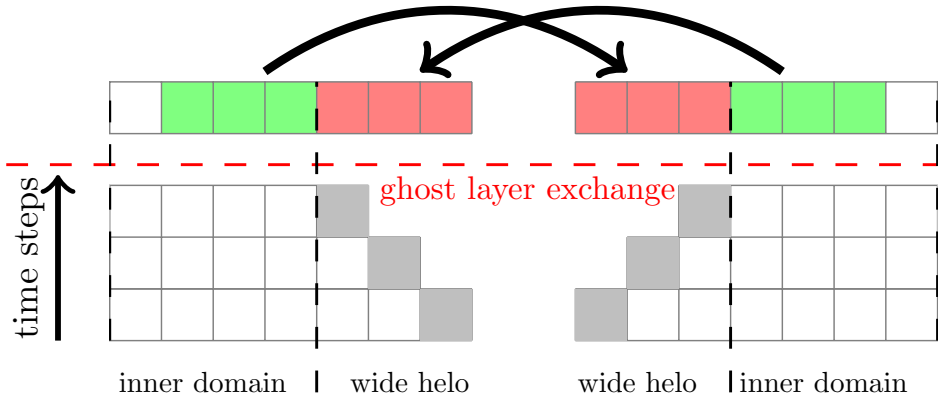


Optimizations III

Algorithm layer

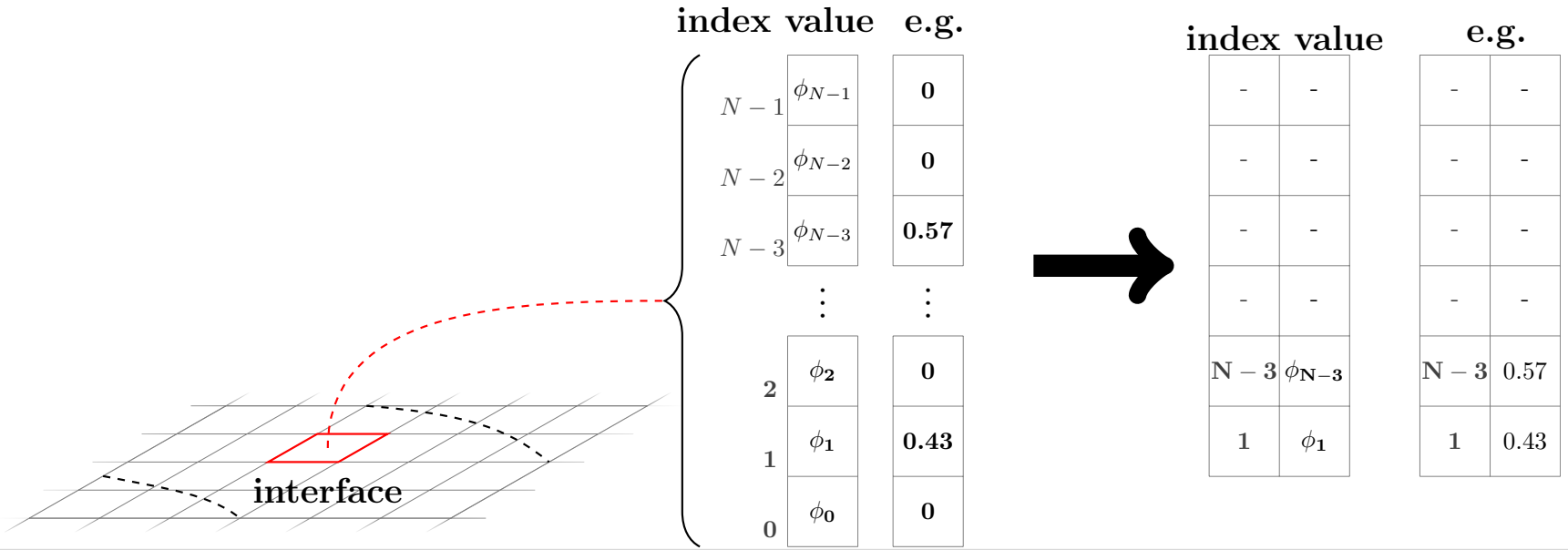
- access patterns / stencils (streaming)
- advanced buffering techniques
- domain decomposition (MPI)
- communication hiding
- wide hello approach
- MPI-IO and reduced mesh output
- local reduction of order parameter (LROP) for ϕ





Local reduced order parameter (LROP)

- **maximal six phases** in one cell are enough
  (Kim, Kim, Kim and Park, (2006), Physical Review E, 74, 061605)
- only **storage** phase **values** $\phi_\alpha \neq 0$ **and their index** in the phase-field vector ϕ instead of all *N* elements
- **other phases** are assumed to be **zero**
  $\longrightarrow$ memory requirements independent from number of phases
  $\longrightarrow$ reduction of calculation time $\sum_\alpha^N \ldots \rightarrow \sum_\alpha^{\max(6)} \ldots$



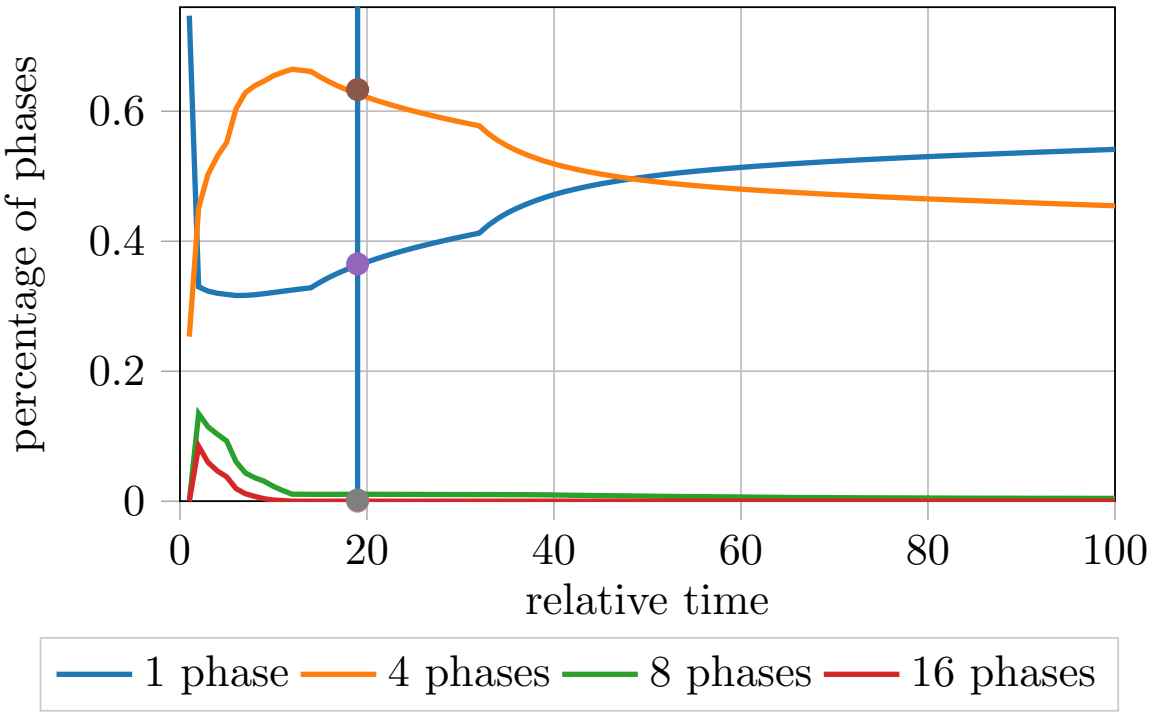
Optimizations II

Hardware layer

- **explicit** vectorization with SIMD **intrinsics**
- light weight macro layer to support **AVX 2** (and **QPX**)
- for $\partial_t \mu$ **classical approach**, calculate multiple cells at once
- for $\partial_t \phi$ the **calculation per cell** is **vectorized**
  $\longrightarrow$ calculate multiple phases at once
  $\longrightarrow$ still possible to use all optimizations (e.g. classification)
  $\longrightarrow$ **LROP cells differ** between neighboring cells, but for **vectorization** they **need the same structure** which results in **complex sorting**
  $\longrightarrow$ good experience with vectorization of four phases (**up to** 25% **peak performance**)
  (Bauer, Hötzer, ... (2015) In High Performance Computing, Networking, Storage and Analysis. IEEE)

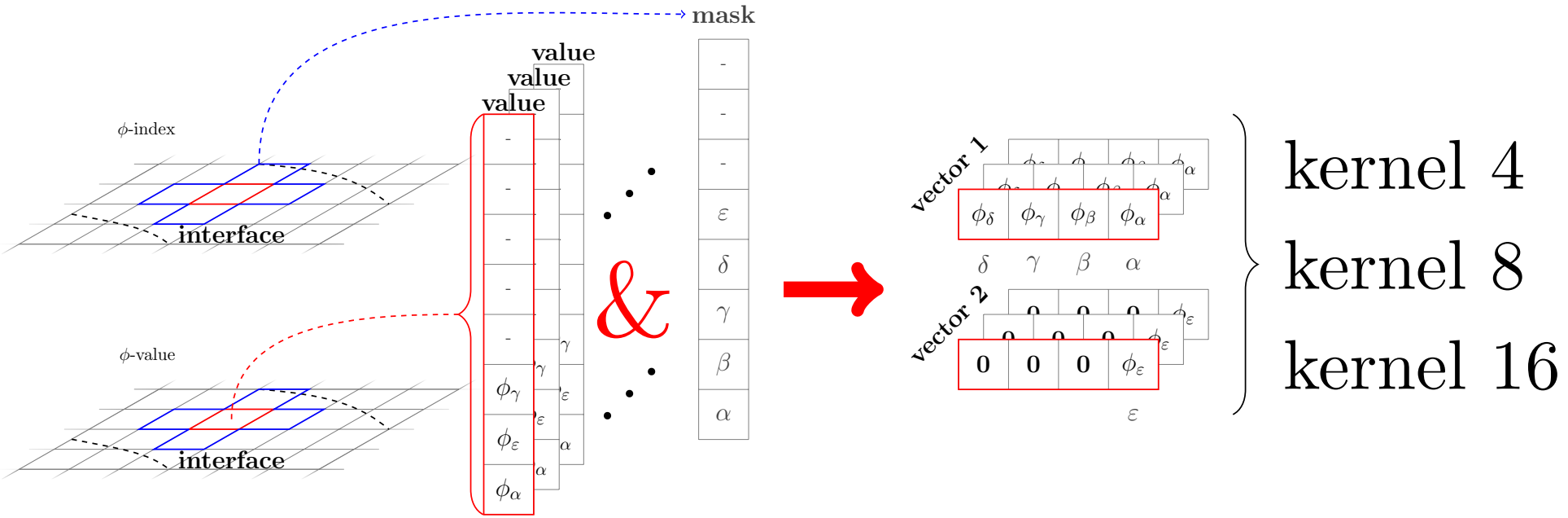
Vectorization

- many **vector matrix multiplications** of the form $\mathbf{y} = \sum \mathbf{Ax}$
- optimized **pattern approach** to pre-rotate all combinations of *x* for four, eight and sixteen phases
- **three kernels** depending on the number of phases *N* to calculate in current cell
  - ⟶ vectorized kernel for **4** phases
  - ⟶ vectorized kernel for **8** phases
  - ⟶ vectorized kernel fot **16** phases

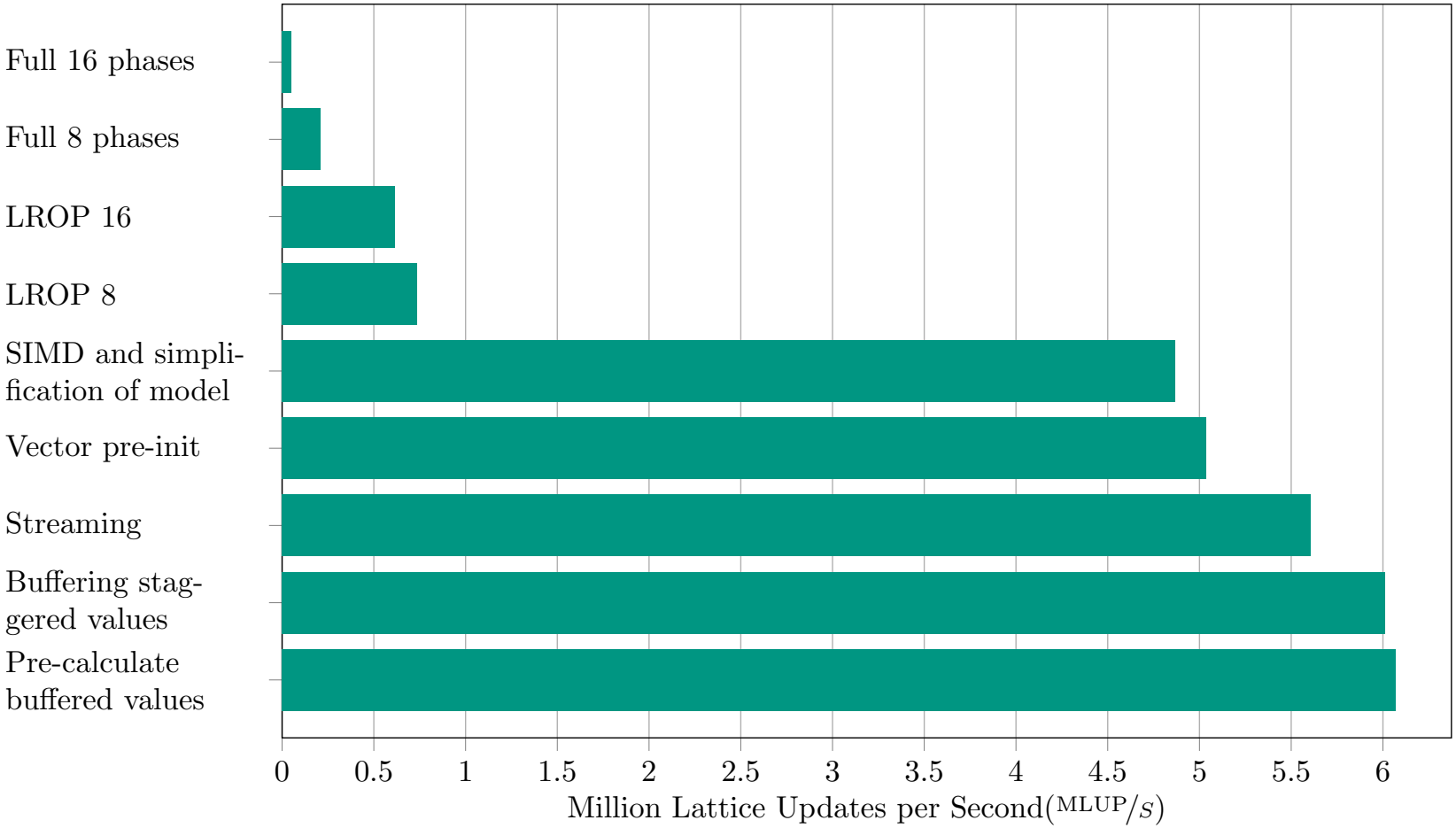


Vectorization

- **mapping** of **LROP cell** to **SIMD vector**
  $\longrightarrow$ all local ϕ vectors of the stencil and matrices need the same order to calculate e.g. $\nabla\phi$
- create **index mask** depending on stencil
- create **SIMD vectors** from LROP cell **based on mask**
- depending on size of mask select the optimal kernel

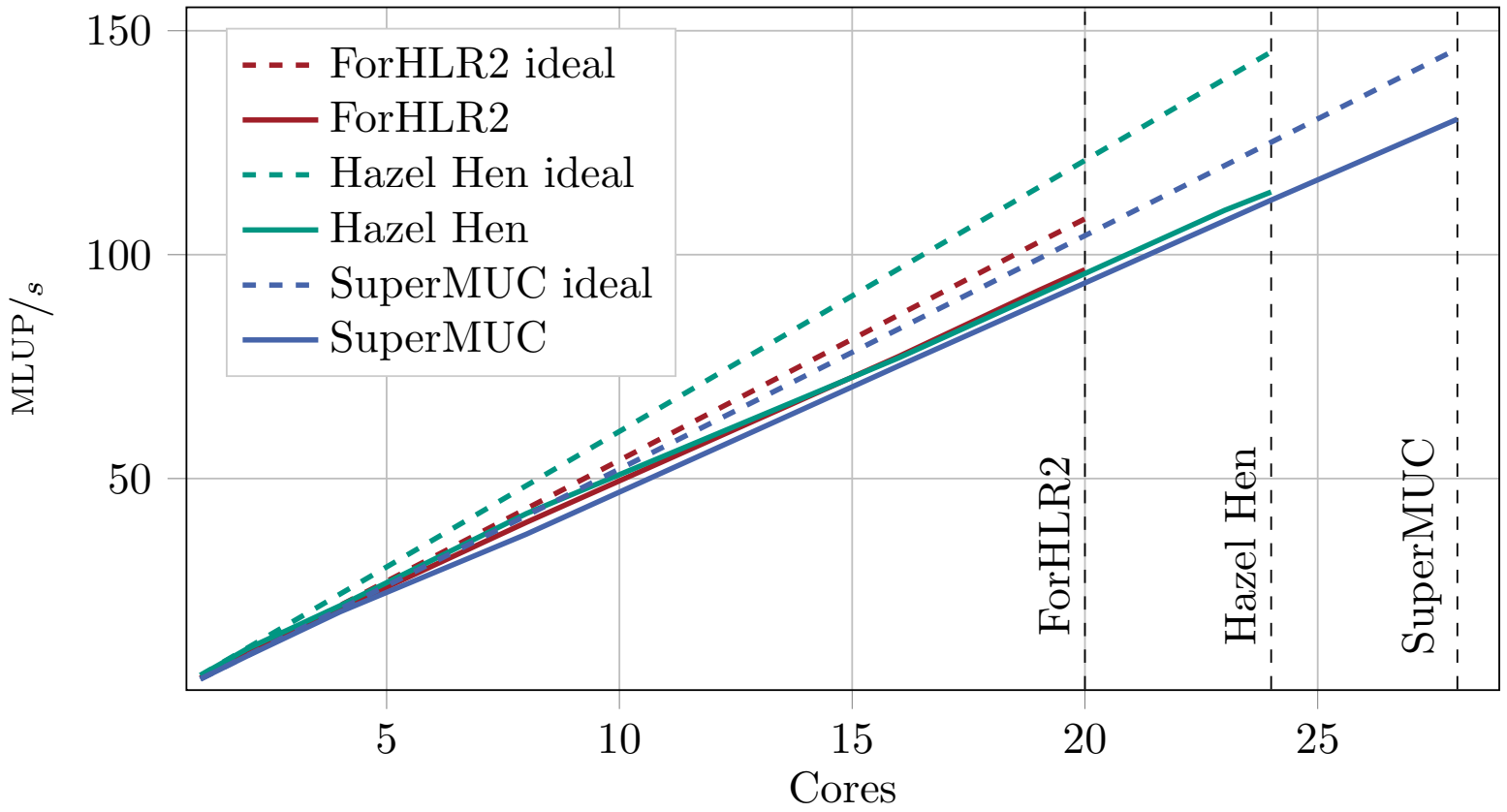


Optimization results – 80^3 voxel cell domain – Hazel Hen



⇒ 22 times faster than the common C code

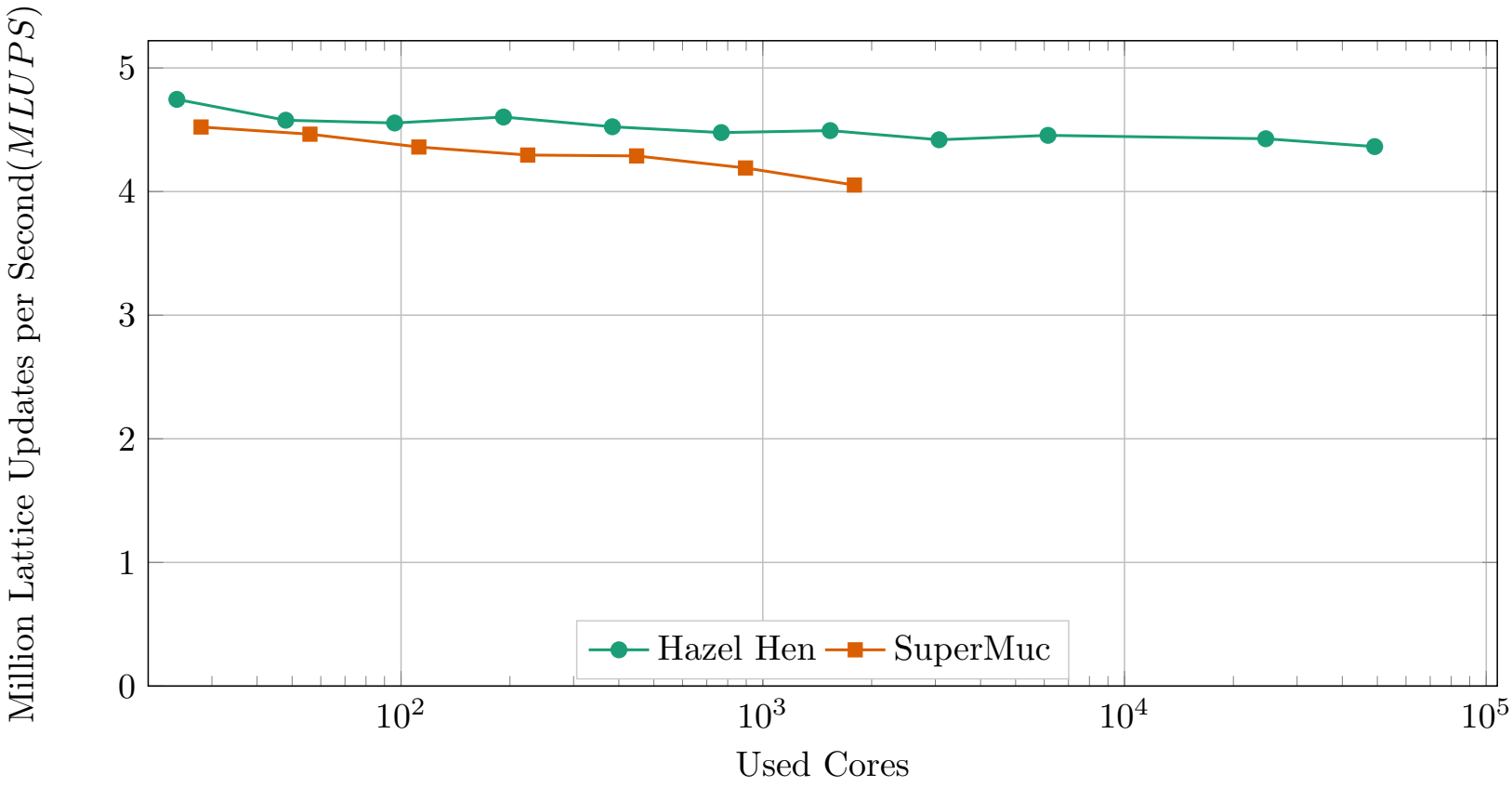
Single node scaling



⇒ linear weak scaling behavior

⇒ full exploitation of nodes on different systems

Weak scaling results

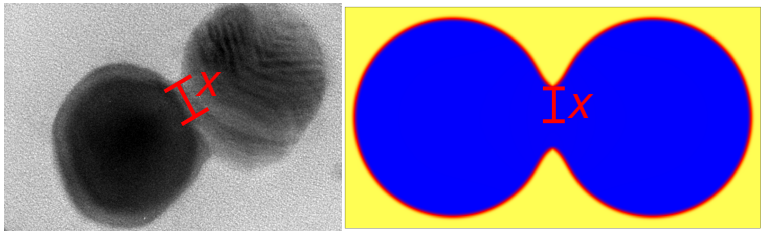


- proper scaling behavior
- wide halo does not improve performance for considered number of cores

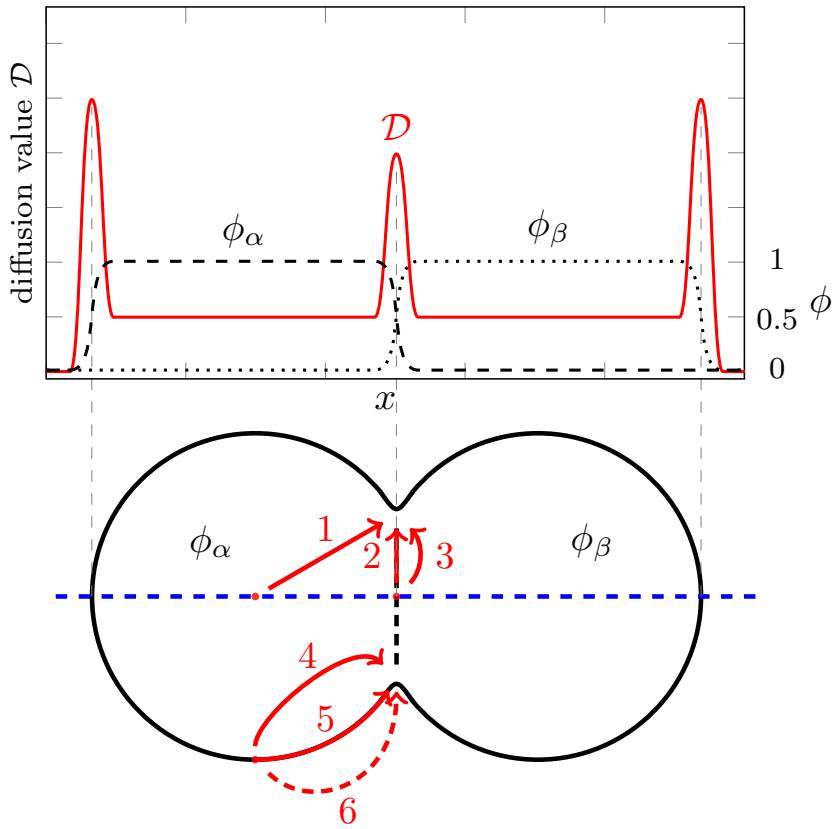
Part VI: Simulation results of the sintering process

Initial and intermediate sintering stage model - Validation

- classical two particle system
- studied parameter:
  - neck radius X



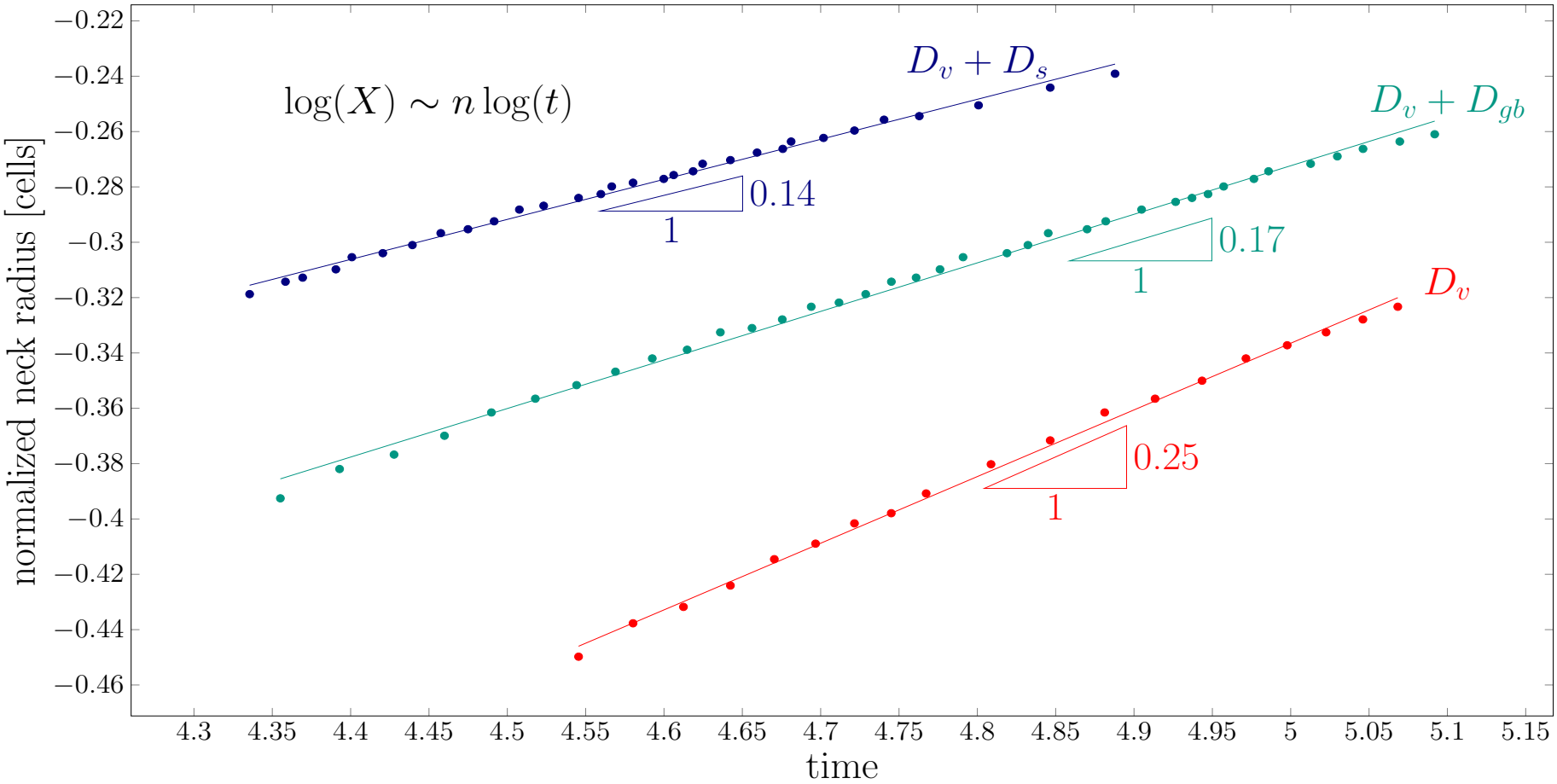
- analytical description of neck radius:
  $X = At^n$, $n \in [0.14, 0.33]$
- depending on dominant diffusion mechanisms
  - volume diffusion D_v (1,3,4)
  - surface diffusion D_s (5)
  - grain boundary diffusion D_{gb} (2)



Wang et al. 2006 Acta Materialia

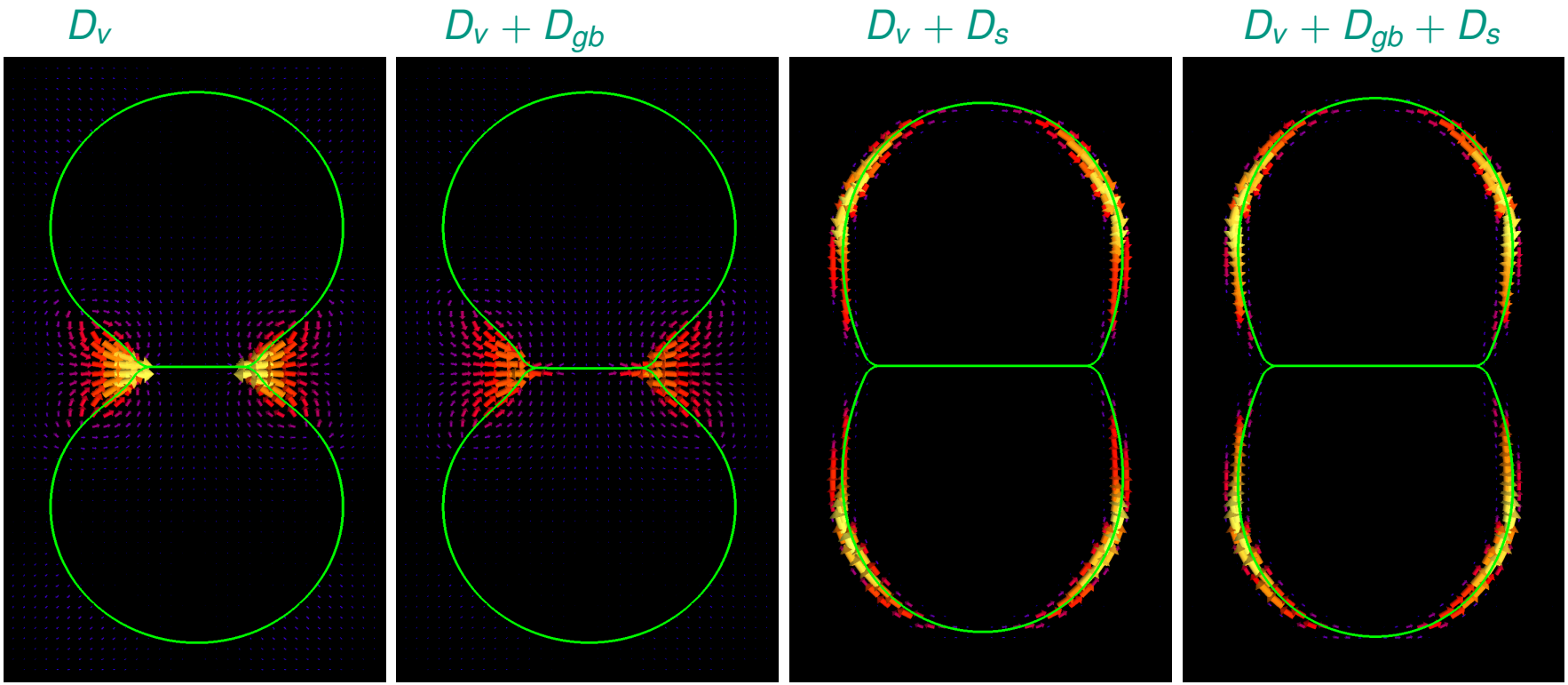
Benjamin Ehreiser, IAM, KIT/Ceramic Processing and Sintering

Initial and intermediate sintering stage model - Validation: neck radius



- good accordance with analytically predicted values of the exponent $\boldsymbol{n} \in [0.14, 0.33]$)

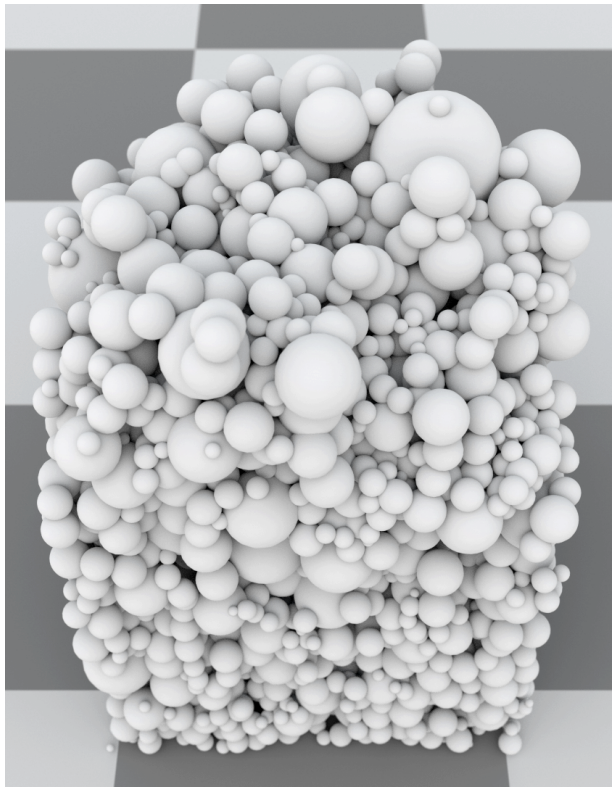
Mass flow for different active diffusion mechanisms



Initial and intermediate sintering stage model - Green body generator

- generation of packings with defined
  - ⟶ density
  - ⟶ particle size distribution
  - ⟶ particle shapes





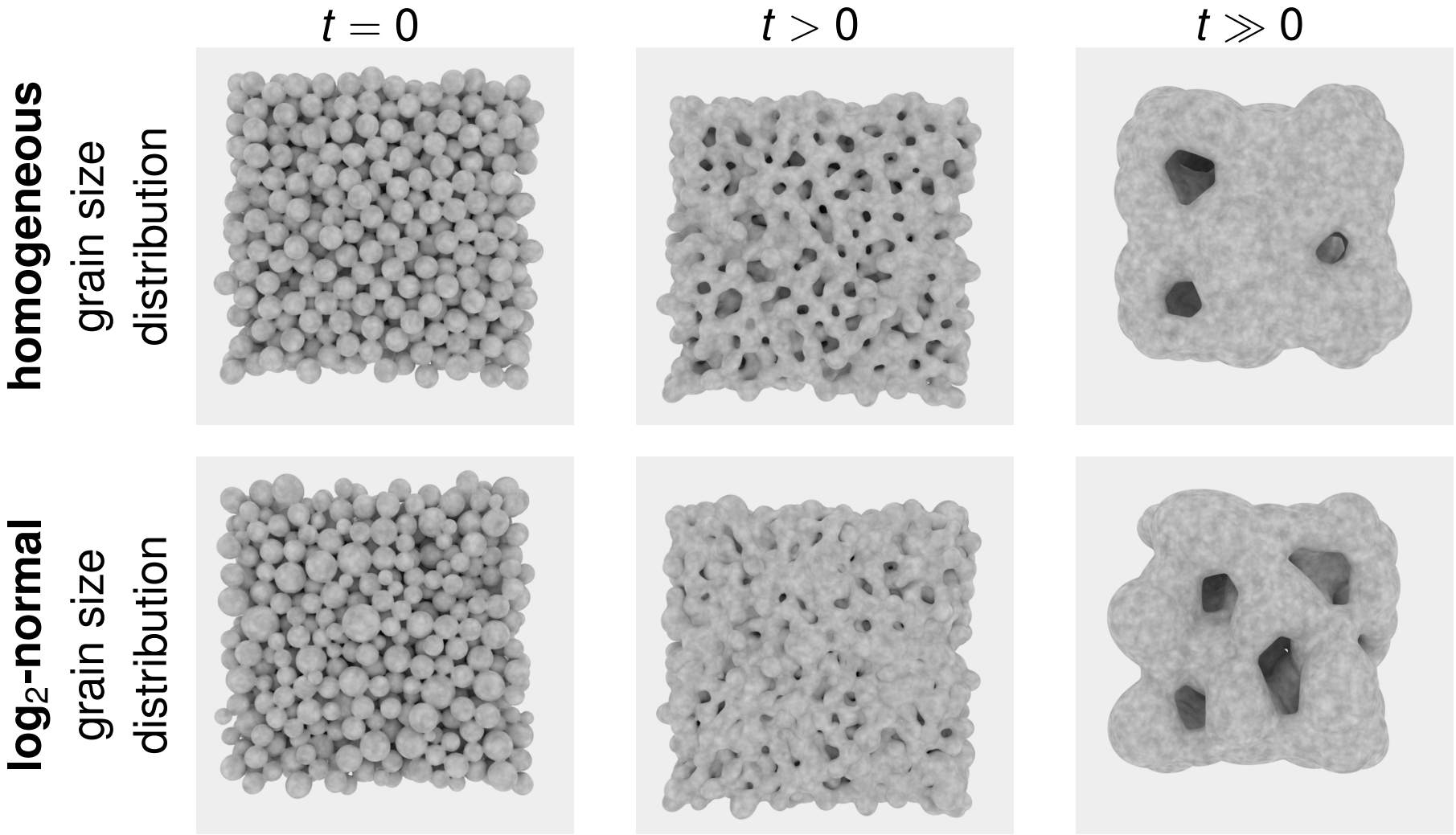
Hötzer et al., Forschung Aktuell, Hochschule Karlsruhe, 2016

Initial and intermediate sintering stage model - 3D simulation

Video: Simulation

- heterogeneous grain size distribution
- **400^3** voxel cells, **1332** CPUs, first **24** hours

Initial and intermediate sintering stage model - First 3D simulation



- both simulations: **400^3** voxel cells on **1332** CPUs for **72** hours

Conclusions

Summary

- efficient calculation of multi phase-field model for sintering processes
- validated model for soild state sintering
- connecting of highly optimized and vectorized kernels
- optimized mapping of LROP cells to vector cells
- massive parallel code

Future work

- adaptive time stepping
- optimizing for NEC SX-ACE architecture
- investigation of green body densification
- investigation of abnormal grain growth with pore interaction

Open questions? Ideas? Improvements?

- Johannes Hötzer: `johannes.hoetzer@kit.edu`
- Britta Nestler: `britta.nestler@kit.edu`