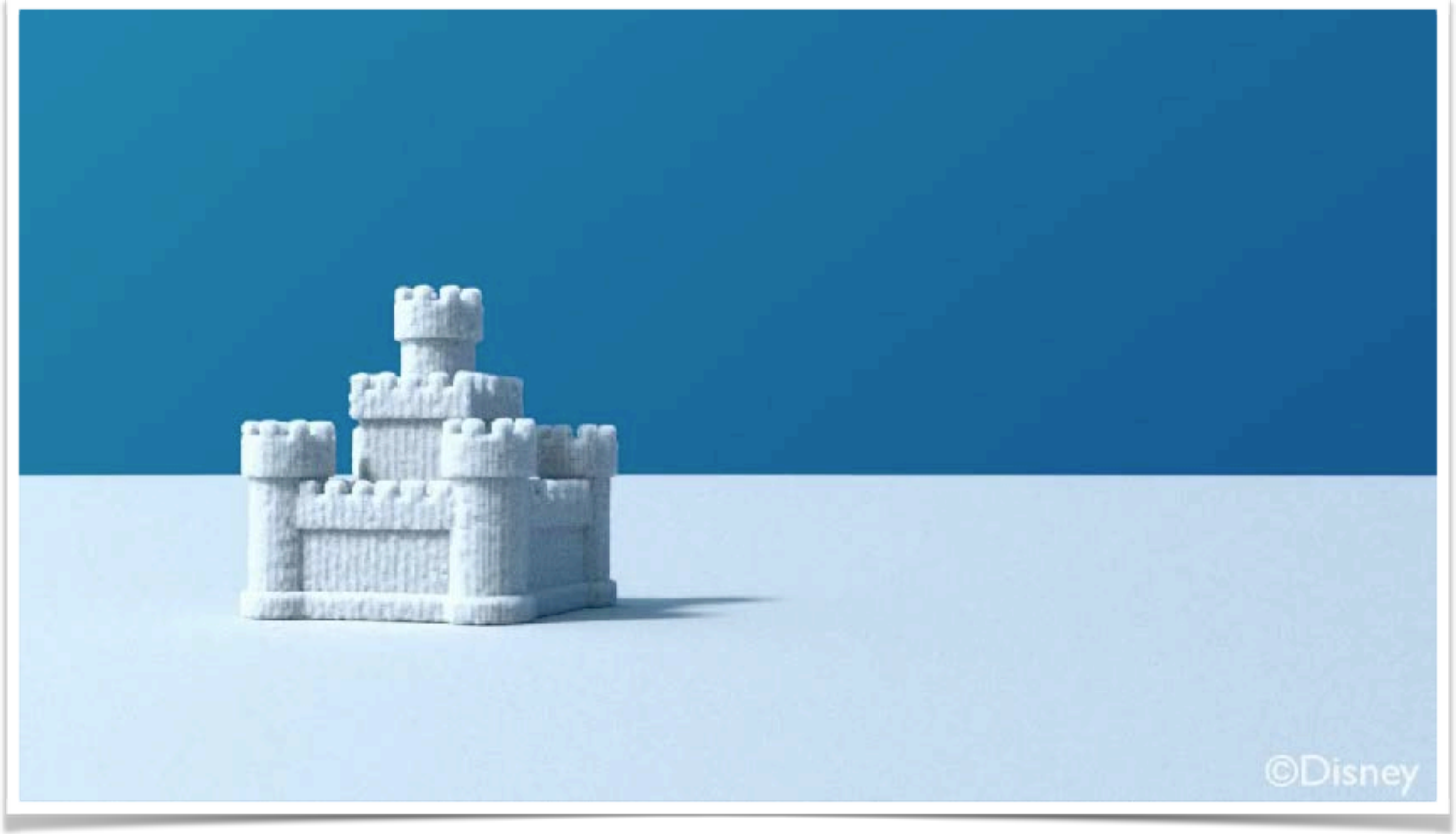
物质点法的最新进展报告

高明
威斯康星大学麦迪逊分校
宾夕法尼亚大学

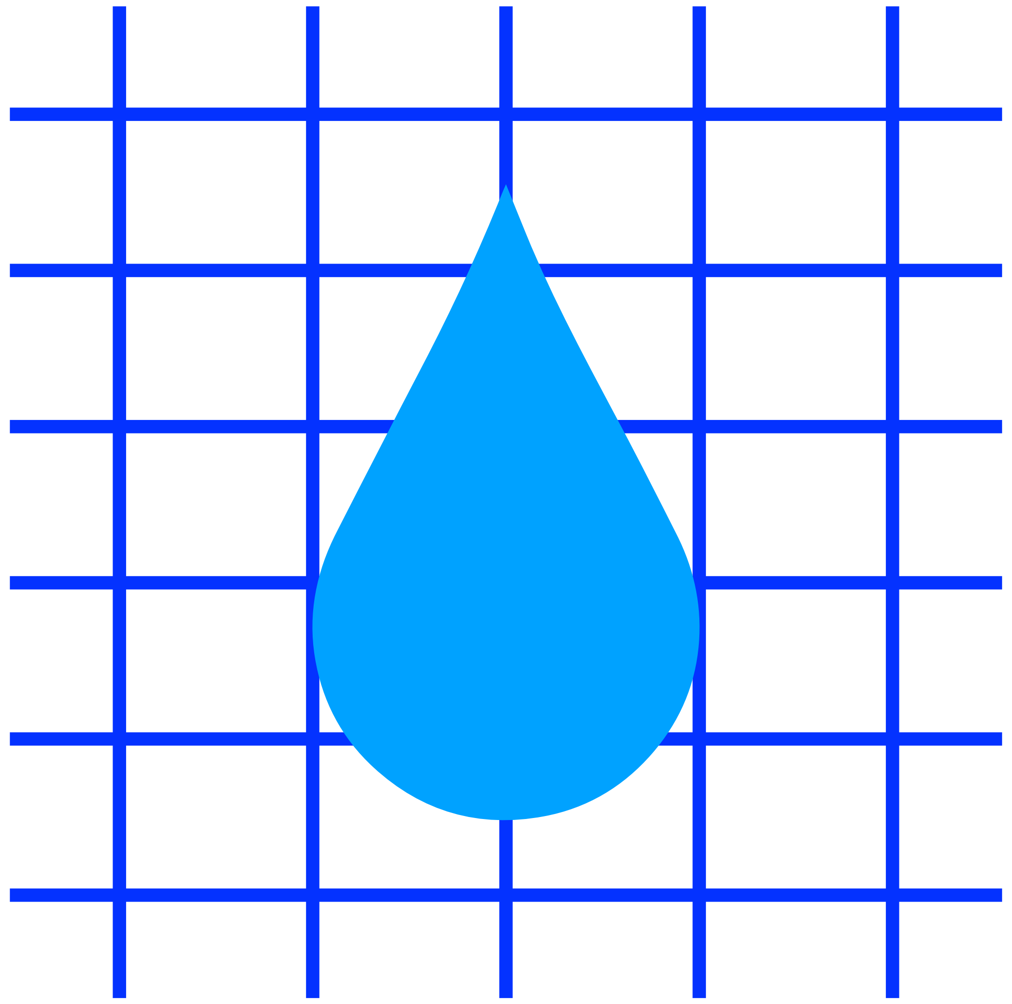
Material point method (物质点法)



Stomakhin et al. 13

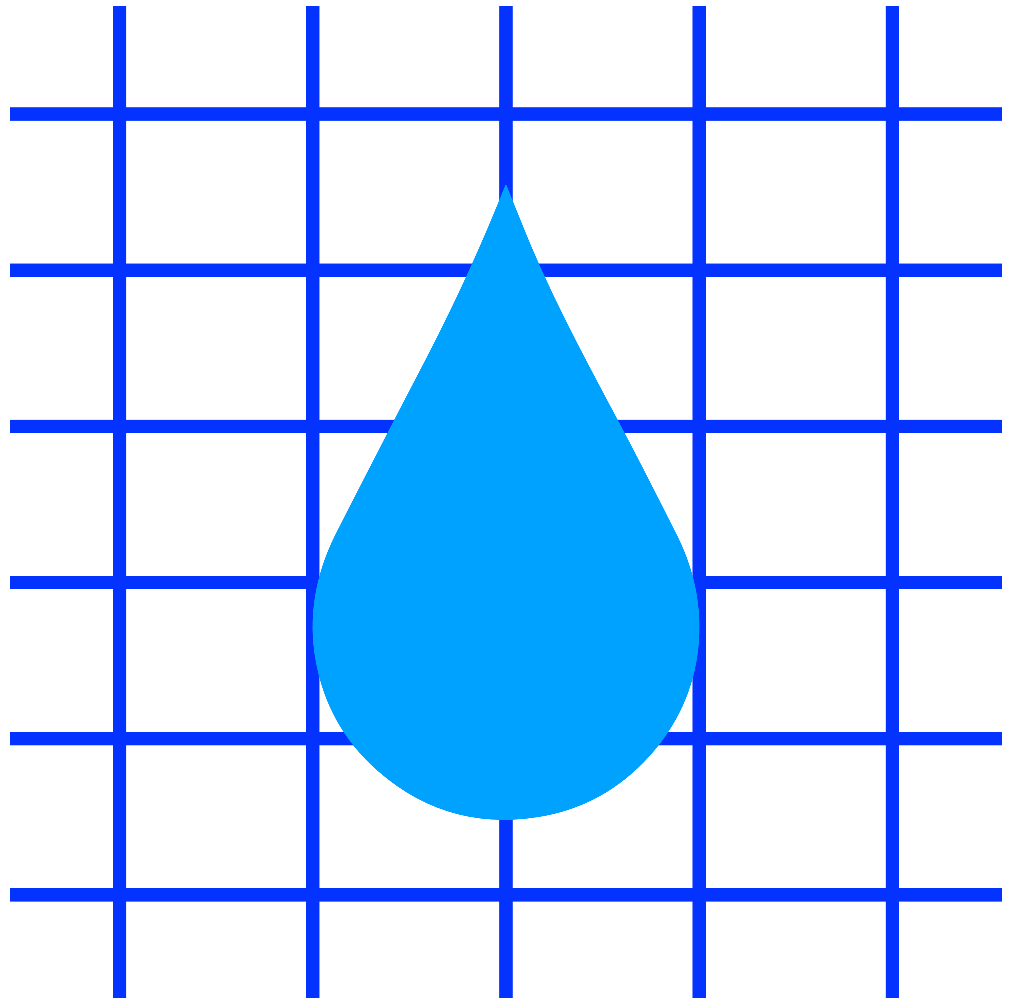
Gao et al. 18

Discretization schemes

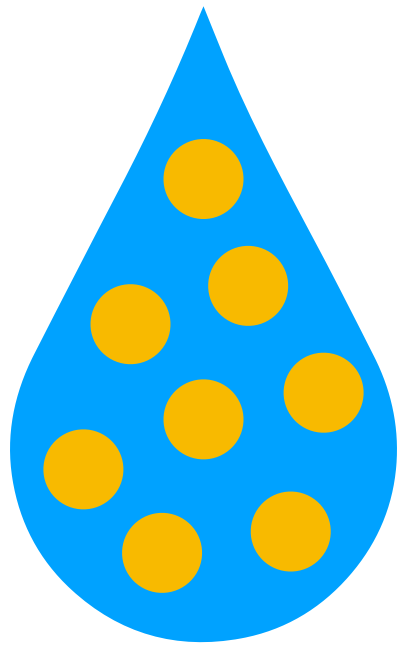


Grid

Discretization schemes

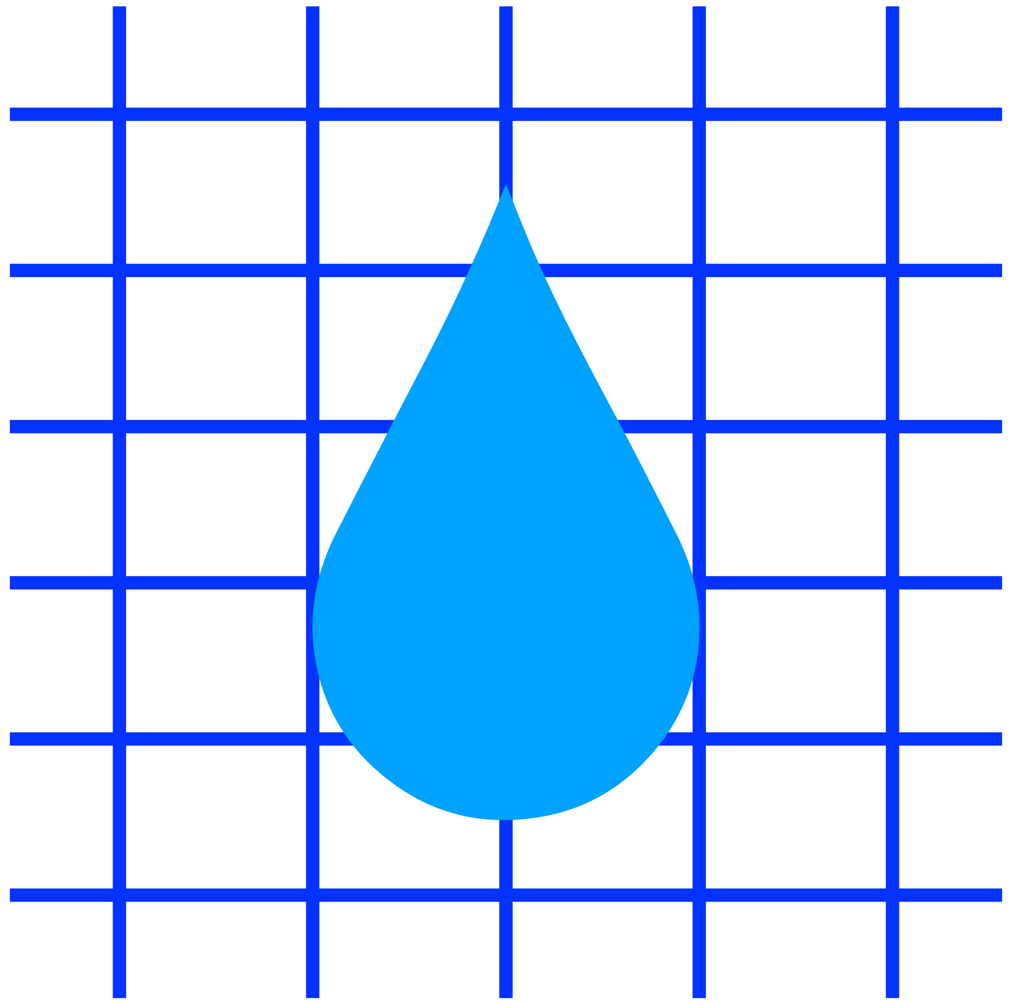


Grid

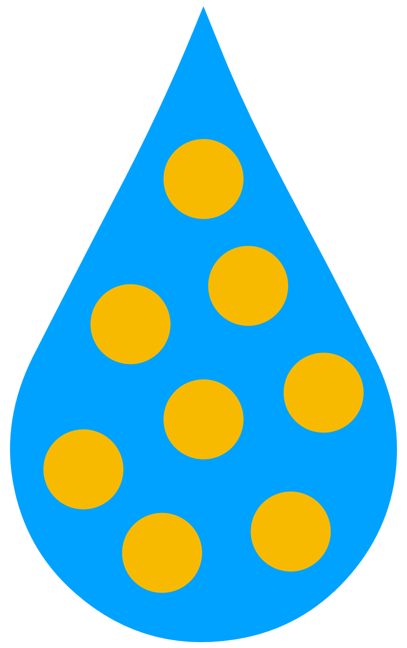


Particle

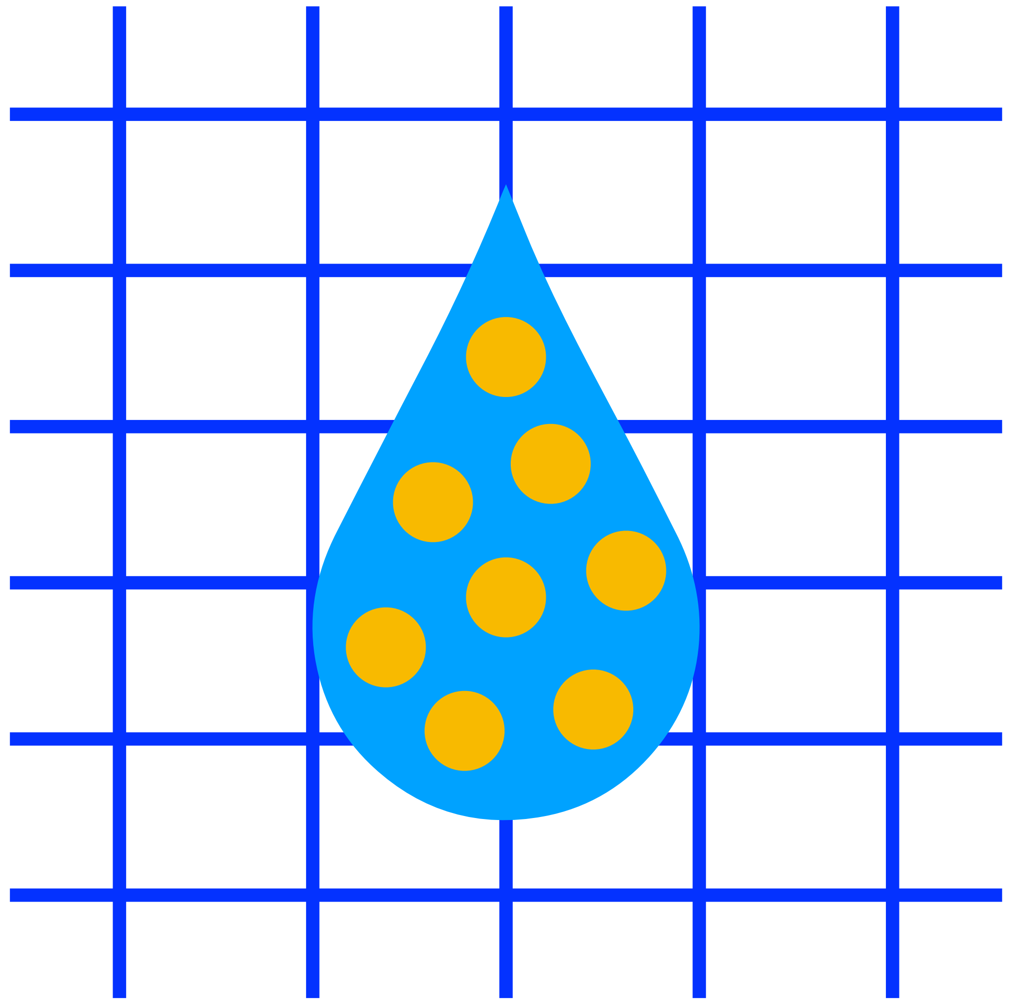
Discretization schemes



Grid

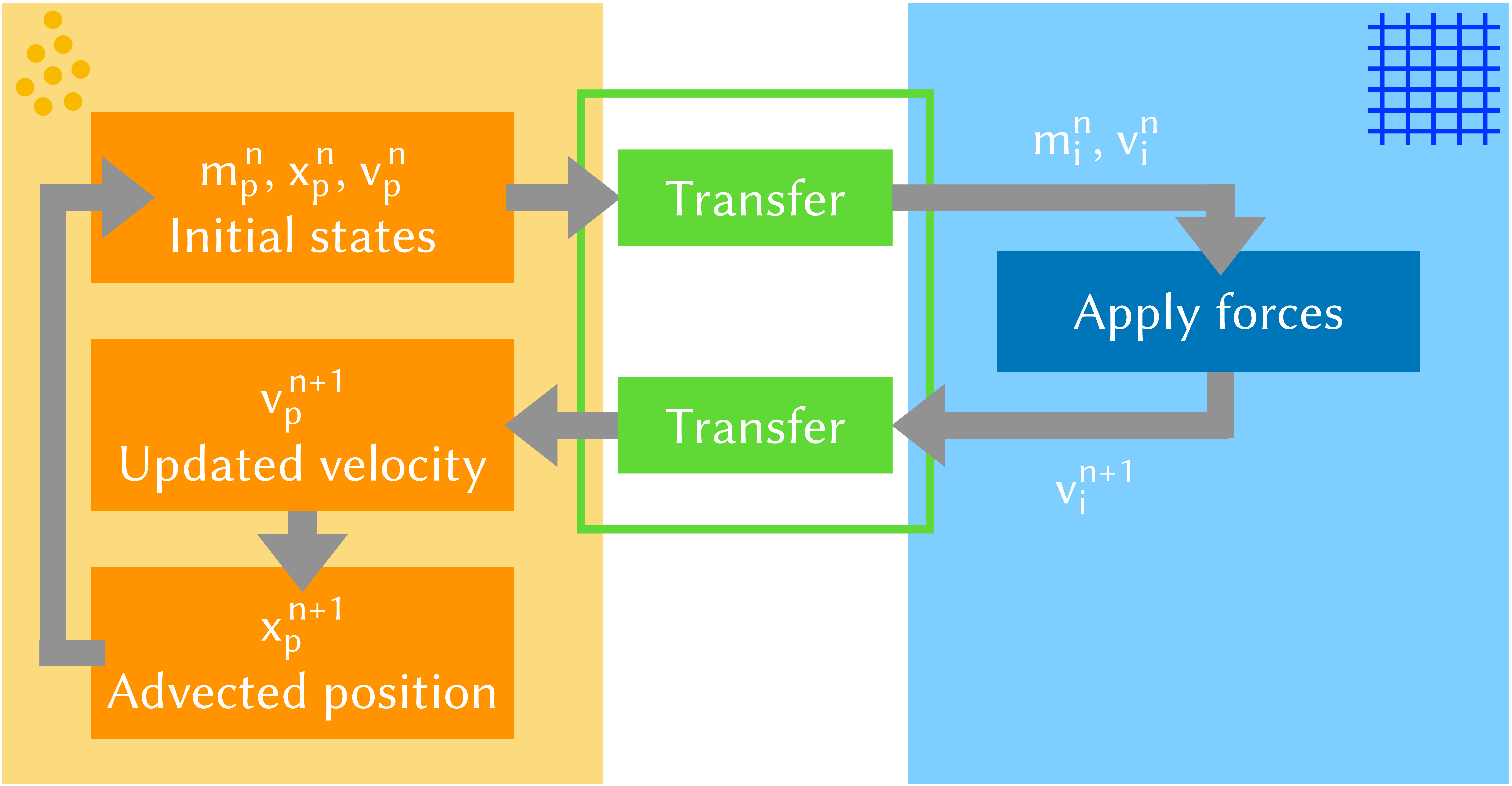


Particle



Hybrid

Data flow



m_p^n, x_p^n, v_p^n
Initial states
v_p^{n+1}
Updated velocity
x_p^{n+1}
Advected position
Transfer
Transfer
m_i^n, v_i^n
Apply forces
v_i^{n+1}

Animating Fluid Sediment Mixture in Particle-Laden Flows

Animating Fluid Sediment Mixture in Particle-Laden Flows

MING GAO, University of Wisconsin, Madison
ANDRE PRADHANA, University of Pennsylvania
XUCHEN HAN, University of California, Los Angeles
QI GUO, University of California, Los Angeles
GRANT KOT, Phosphorus
EFTYCHIOS SIFAKIS, University of Wisconsin, Madison
CHENFANFU JIANG, University of Pennsylvania

Fig. 1. **Sediment transport:** Our method can animate intricate two-way coupled particle-laden flows such as sediment transport in liquid.

In this paper, we present a mixed explicit and semi-implicit Material Point Method for simulating particle-laden flows. We develop a Multigrid Preconditioned fluid solver for the Locally Averaged Navier Stokes equation. This is discretized purely on a semi-staggered standard MPM grid. Sedimentation is modeled with the Drucker-Prager elastoplasticity flow rule, enhanced by a novel particle density estimation method for converting particles between representations of either continuum or discrete points. Fluid and sediment are two-way coupled through a momentum exchange force that can be easily resolved with two MPM background grids. We present various results to demonstrate the efficacy of our method.

CCS Concepts: • **Computing methodologies** → **Physical simulation**;

Additional Key Words and Phrases: Material Point Method (MPM), particle-fluid interaction, multiphase, sedimentation, sediment transport

ACM Reference format:
Ming Gao, Andre Pradhana, Xuchen Han, Qi Guo, Grant Kot, Eftychios Sifakis, and Chenfanfu Jiang. 2018. Animating Fluid Sediment Mixture in Particle-Laden Flows. *ACM Trans. Graph.* 37, 4, Article 1 (August 2018), 11 pages.
DOI: 10.1145/3197517.3201309

DOI: 10.1145/3197517.3201309

1 INTRODUCTION

Recently, multi-phase multi-material simulations are increasingly gaining attention from computer graphics researchers. Simulating various phases or materials in a unified framework is particularly favored. Existing work includes coupled Lagrangian particle simulation with Position Based Dynamics (PBD) [Macklin et al. 2014], water-gas mixtures [Nielsen and Østerby 2013] with an Eulerian method, solid-fluid phase-change [Stomakhin et al. 2014] and porous granular media [Pradhana-Tampubolon et al. 2017] with Material Point Method (MPM), as well as interactive solids and fluids based on the mixture model with Smoothed Particle Hydrodynamics (SPH) [Yan et al. 2016].

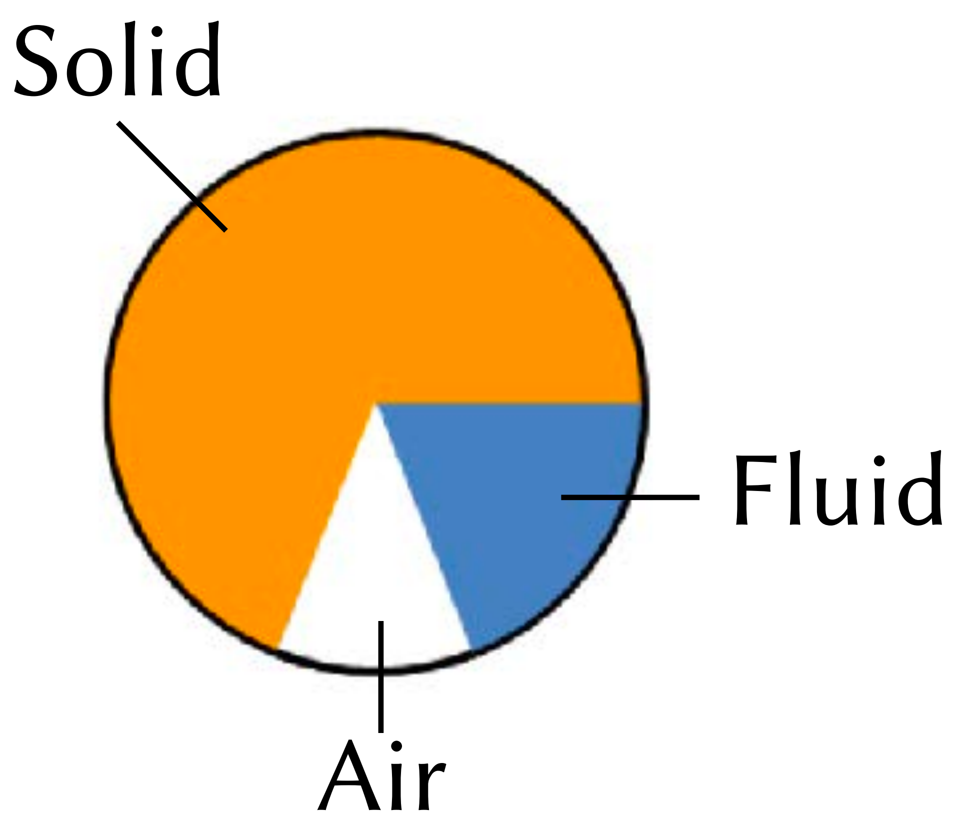
Most of the existing approaches are based on *continuum* mixture theory [Manninen et al. 1996]. The continuum assumption for each material phase is essential for simulations of macroscopic porous media (e.g., landslides and liquid blending). However, it may fail to capture the correct behavior of particle-laden flows where the solid phase is on a relatively small scale. Note that particle-laden sediment flow is ubiquitous in natural systems. Typical examples include sediment transport, sedimentation, volcano eruption, dune migration by erosion with ripples, and dust storms. The significance of understanding and simulating these phenomena is also recognized in many engineering applications, such as granular material fluidization [van der Hoef et al. 2006] and coastal erosion prediction [Sun and Xiao 2016a].

M. Gao, A. Tampubolon, X. Han, Q. Guo, G. Kot, E. Sifakis, C. Jiang
ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH), 2018

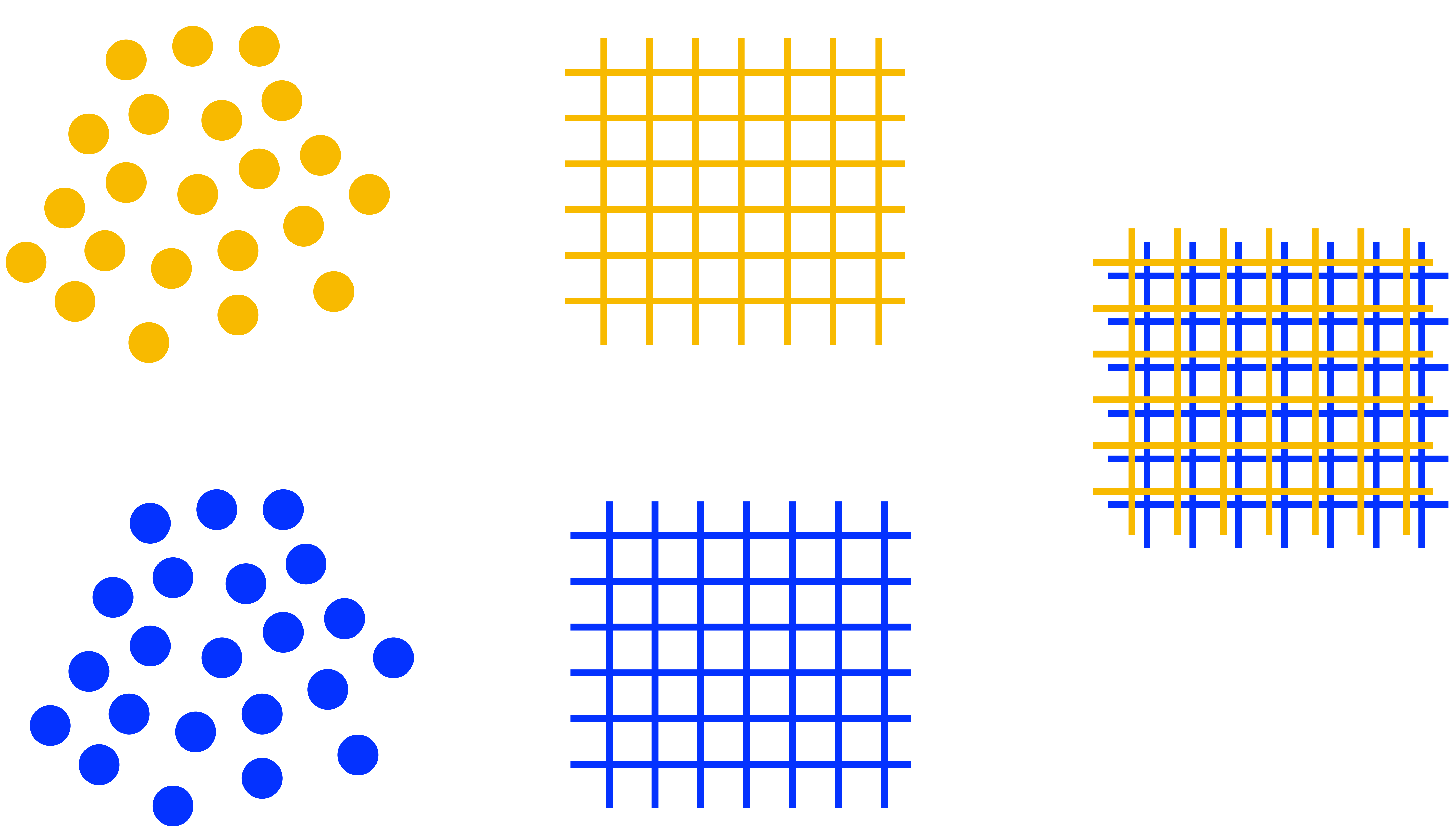
GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018


fluid
sediment

Approach: mixture in particles vs. grid



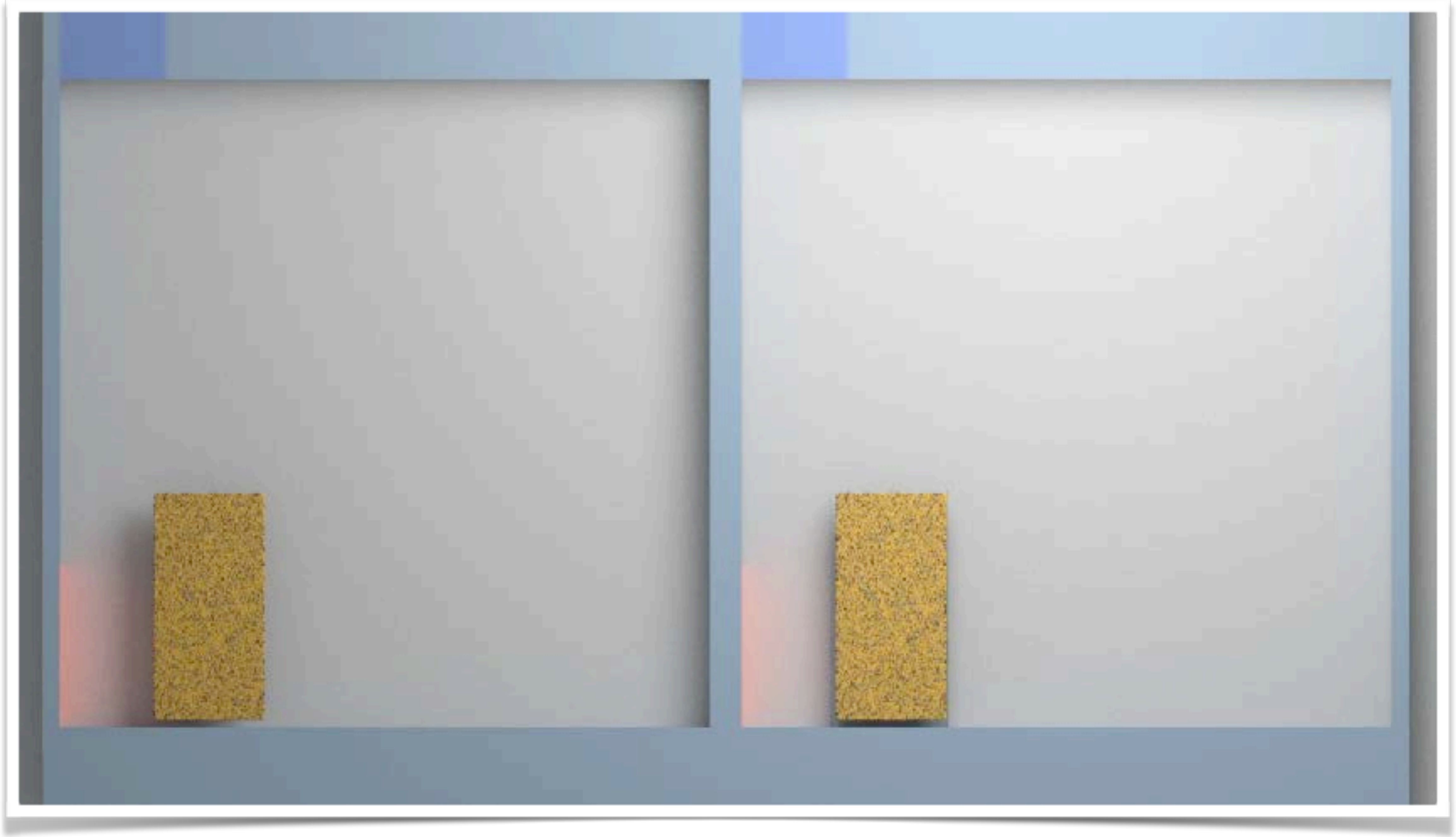
Colom et al. 15



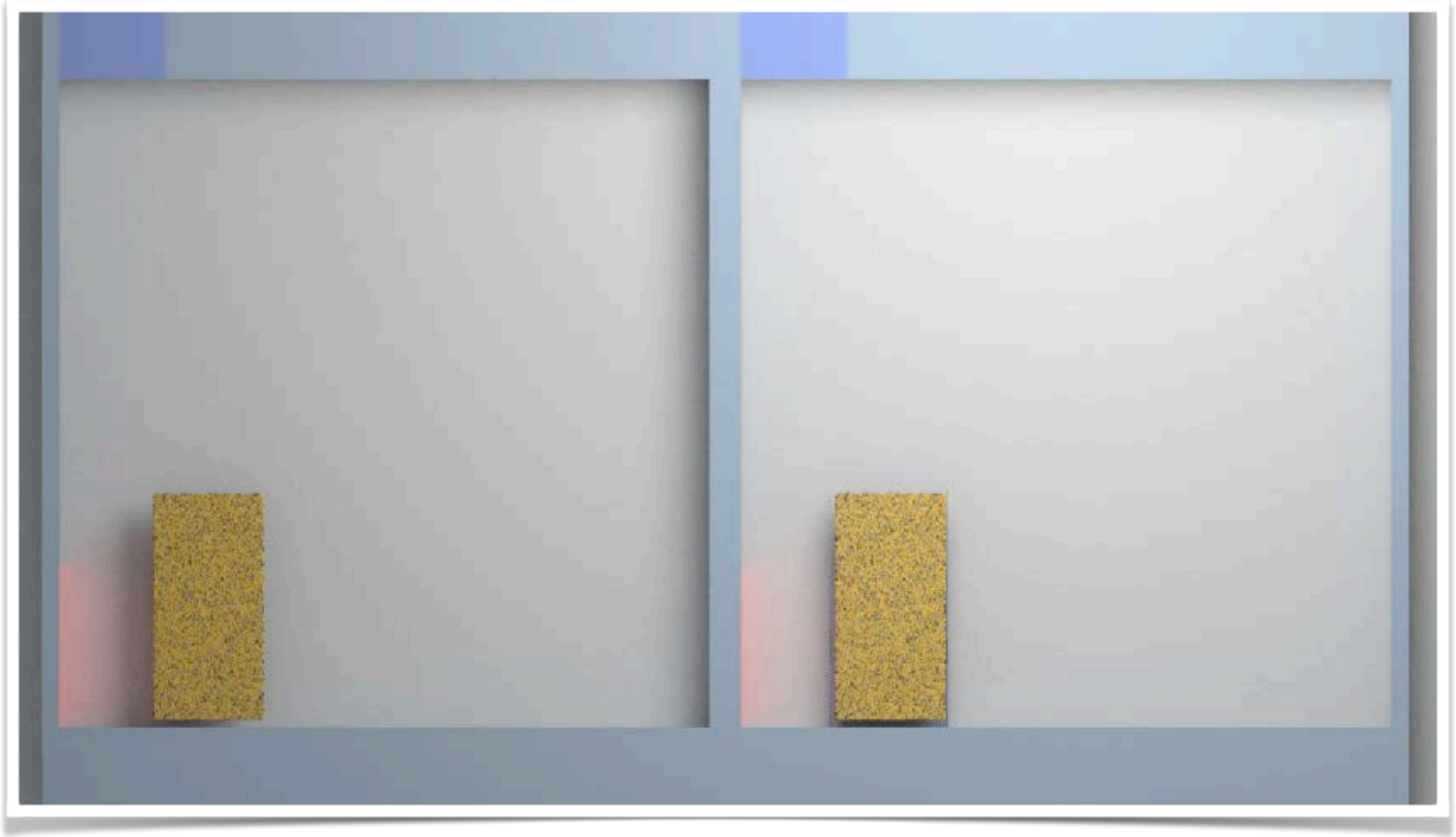
3 phases 1 point

2 phases 2 point

Approach: one-way coupling vs. two-way

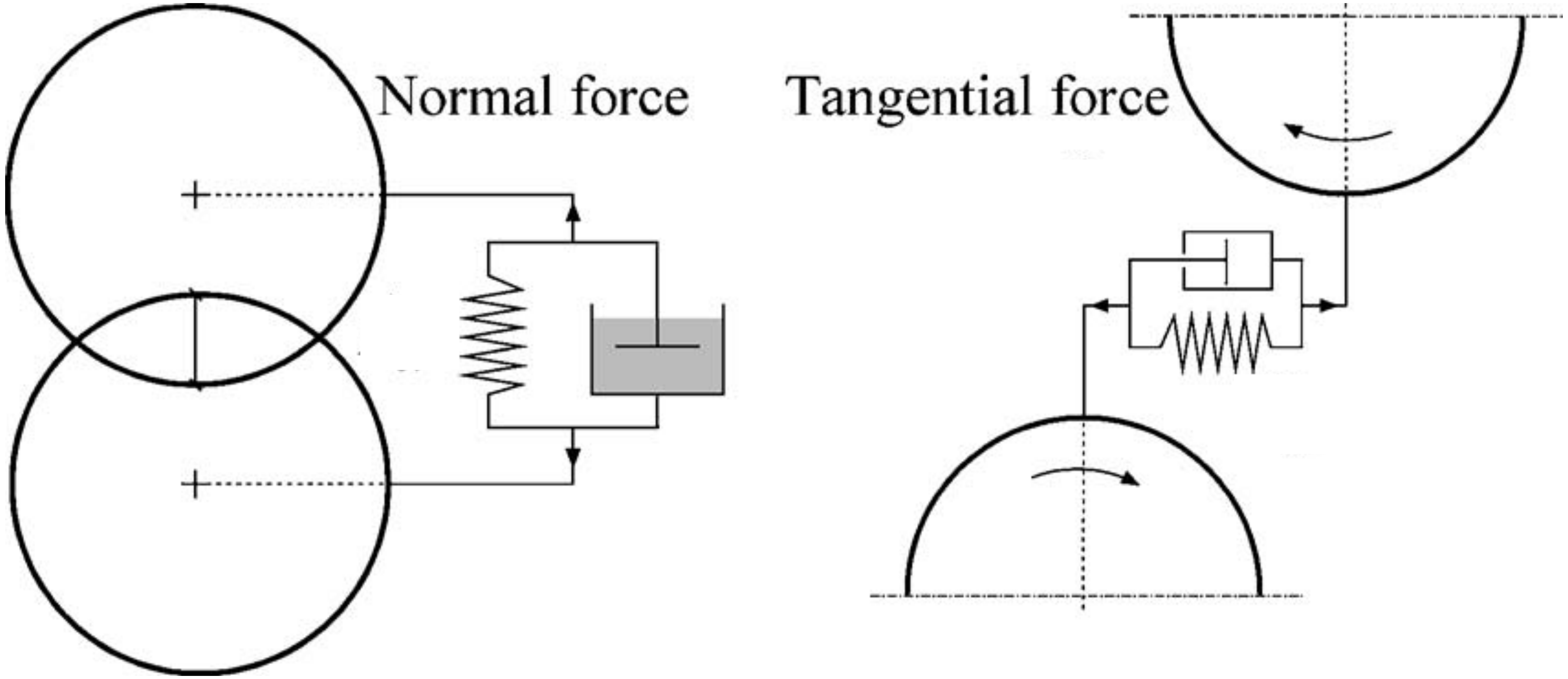


One way coupling



Two way coupling

Sediment: DEM vs. MPM

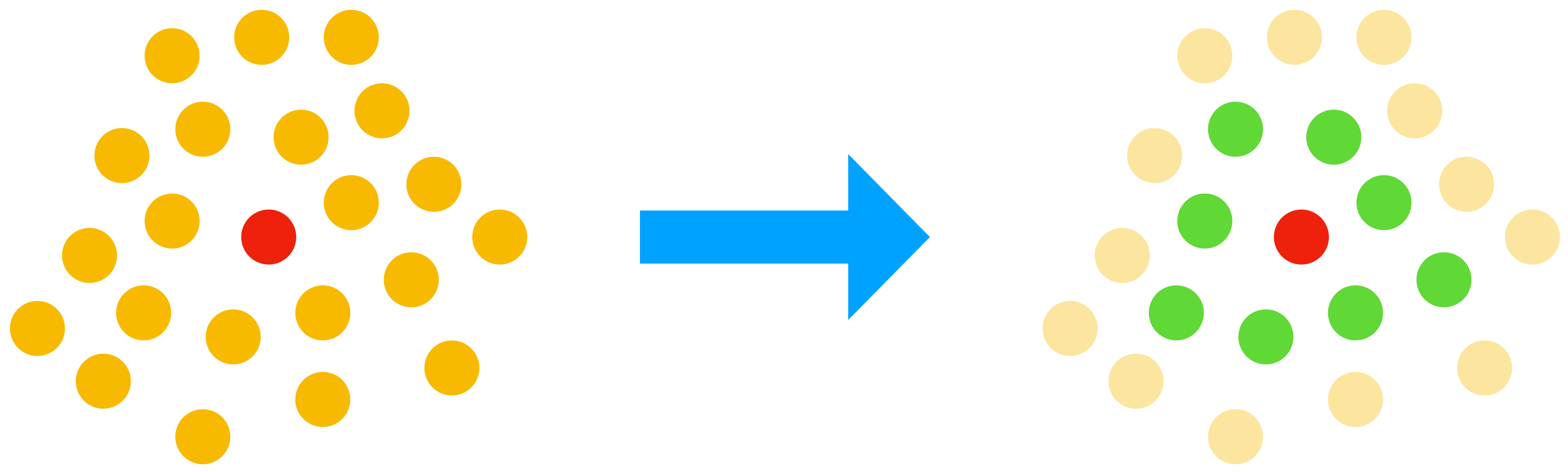


Krugger-Emden et al. 05

Translational interaction

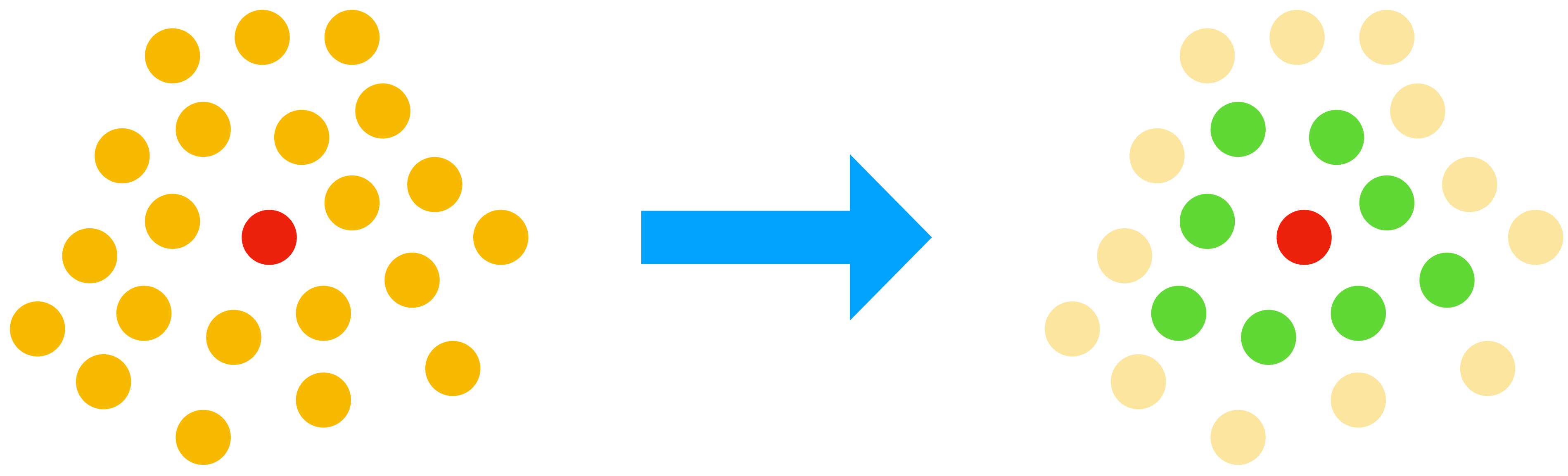
Rotational interaction

Sediment: DEM vs. MPM

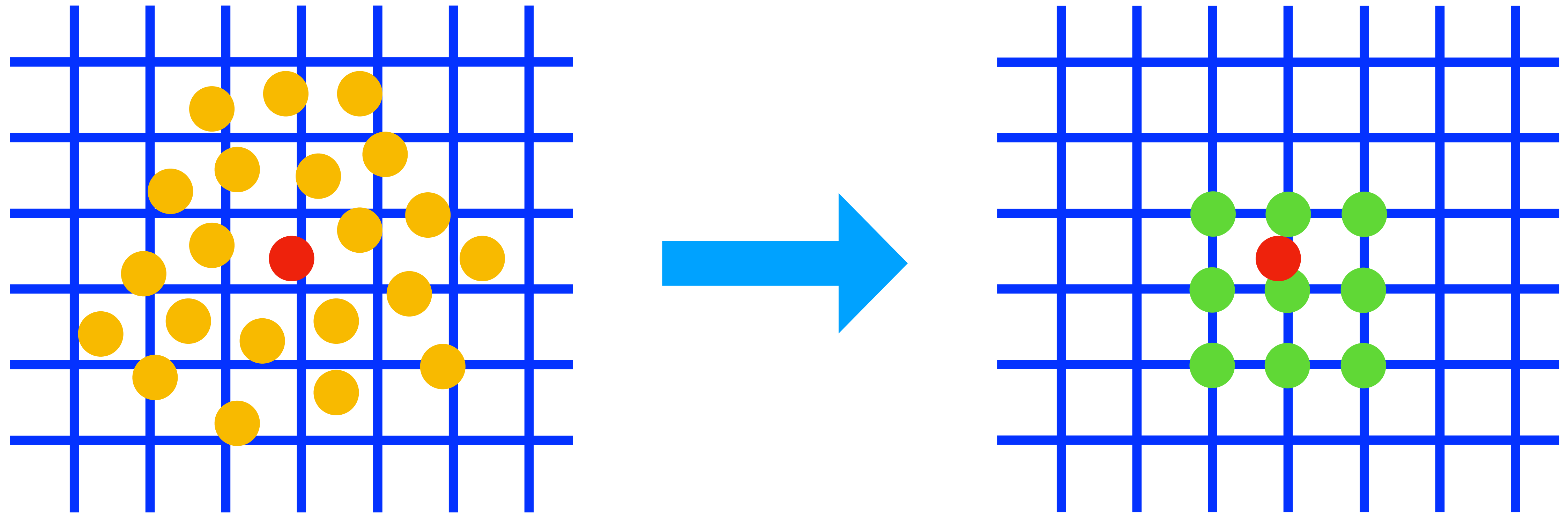


DEM - discrete view

Sediment: DEM vs. MPM

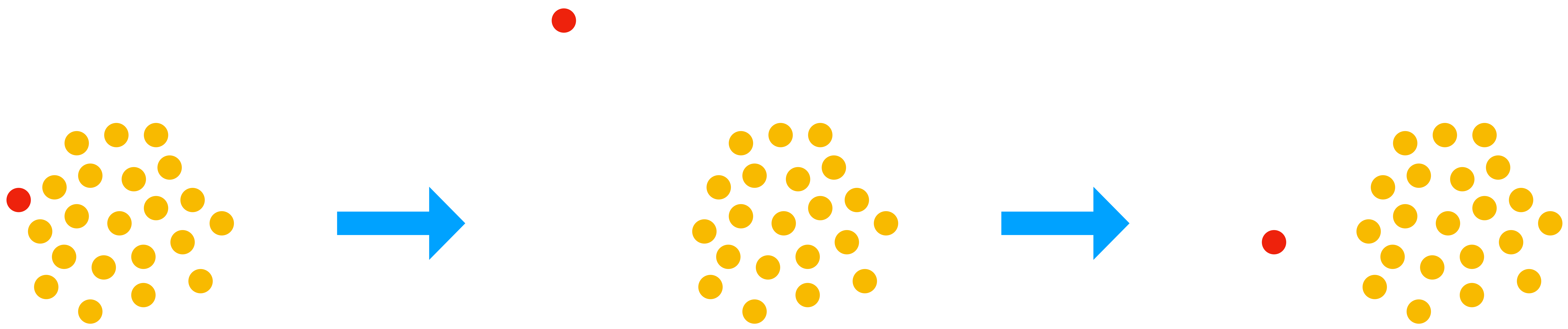


DEM - discrete view

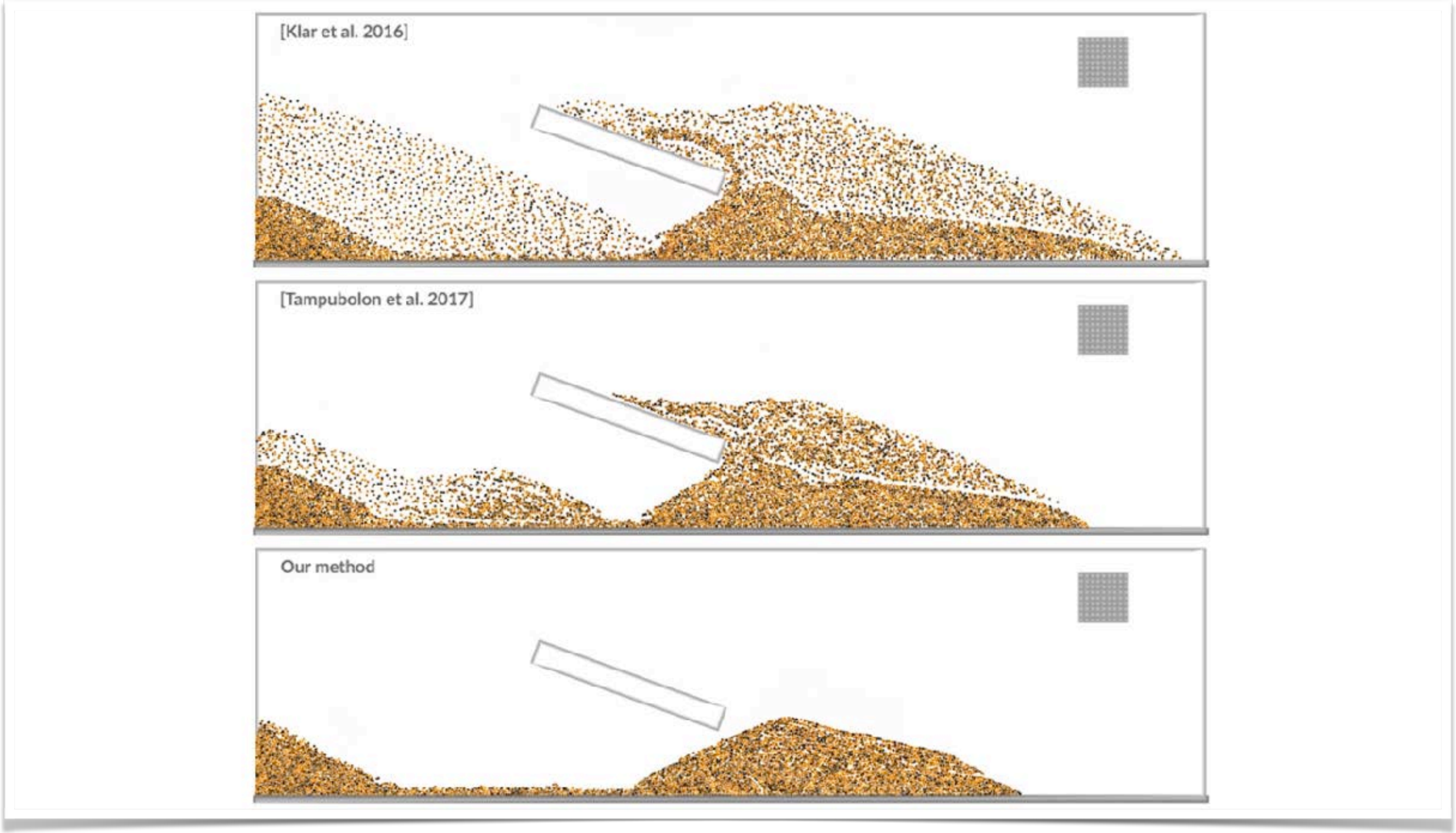


MPM - continuum view

Challenge of MPM: handle discrete particles



Challenge of MPM: handle discrete particles



Volume gain problem

Fluid - previous methods

Tampubolon et al. 17

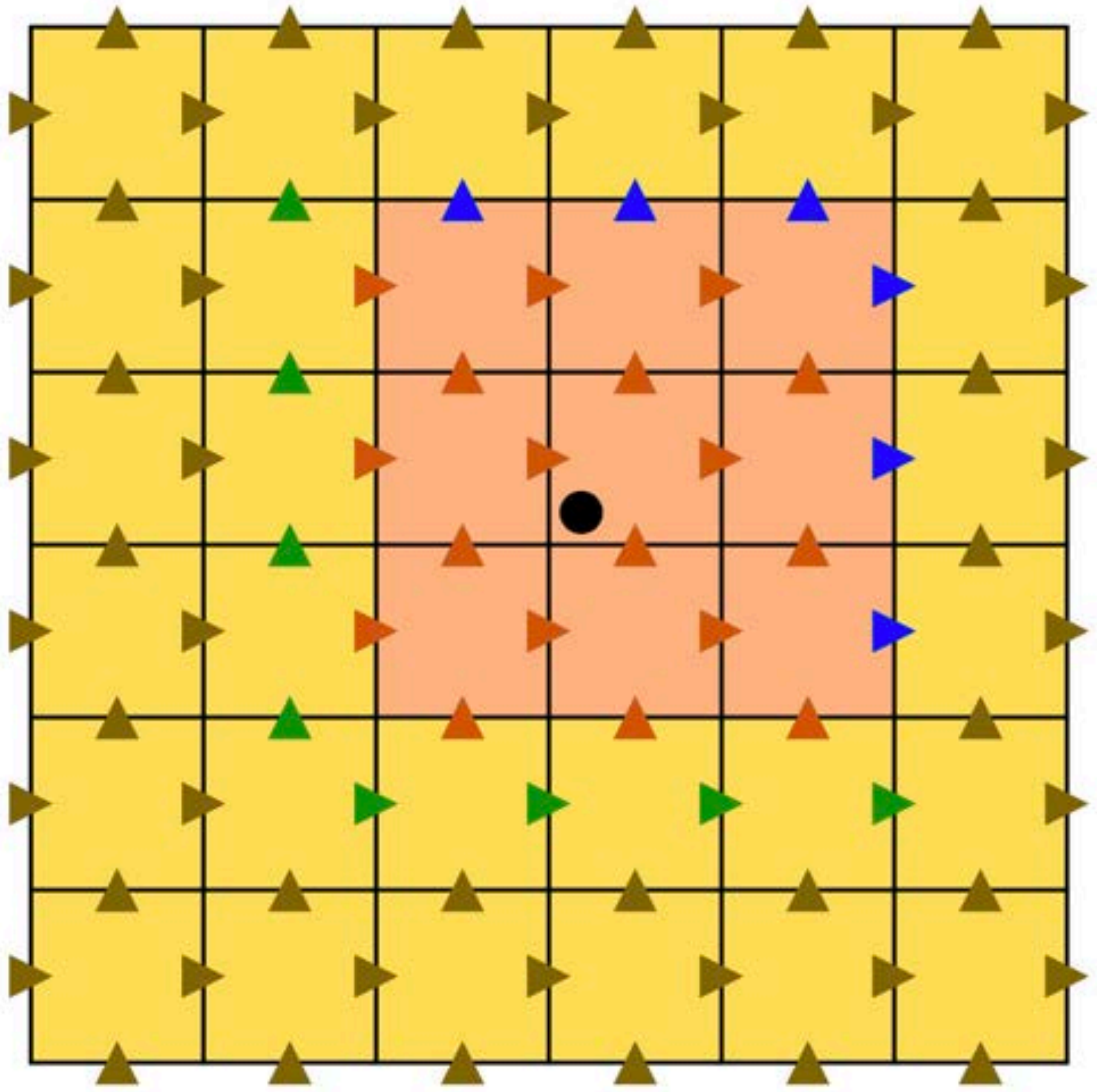
Weakly compressible

Stringent time step

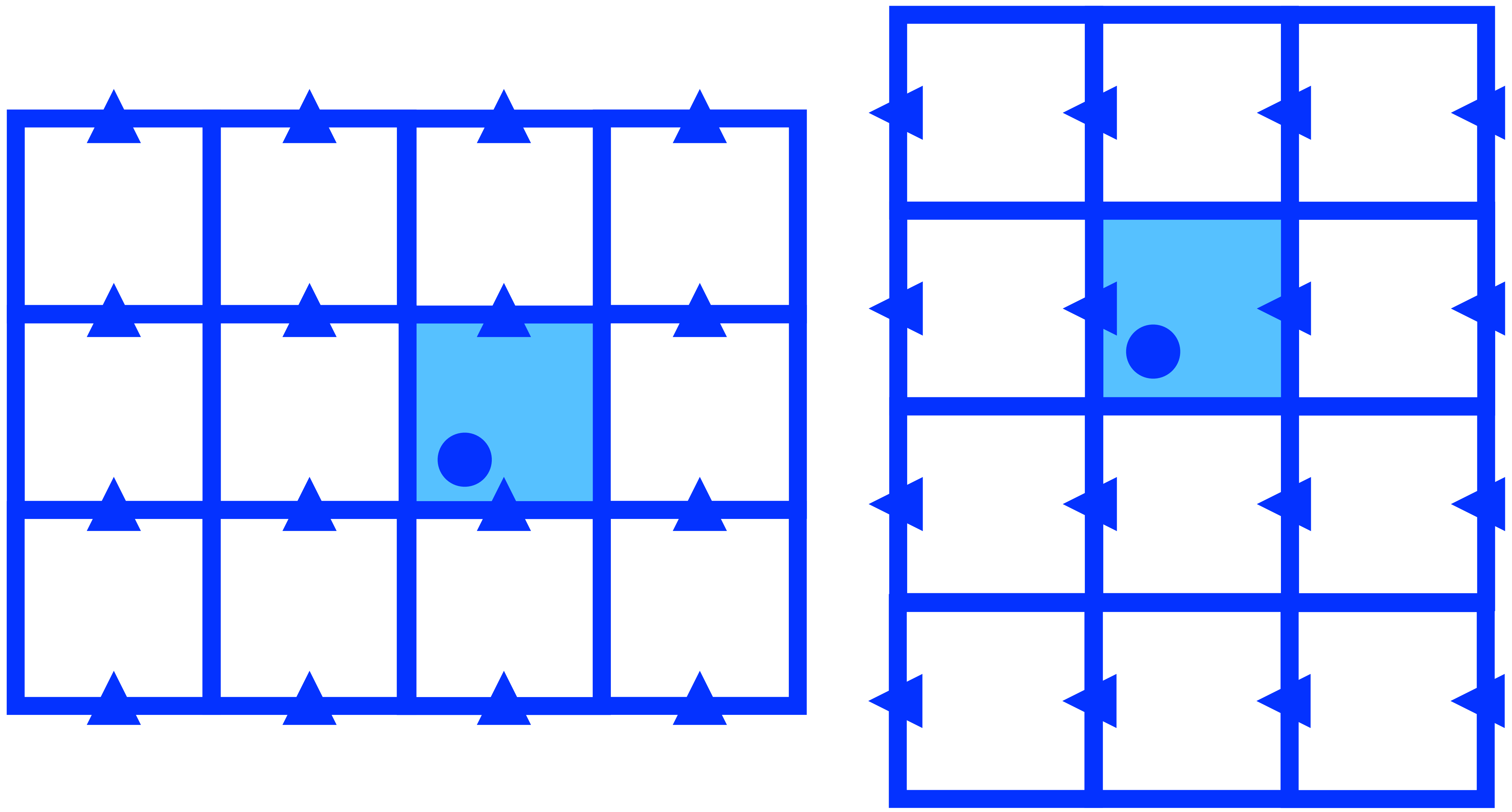
Modified sand model

Fluid - previous methods

Horizontal component velocity ◀
Vertical component velocity ▲

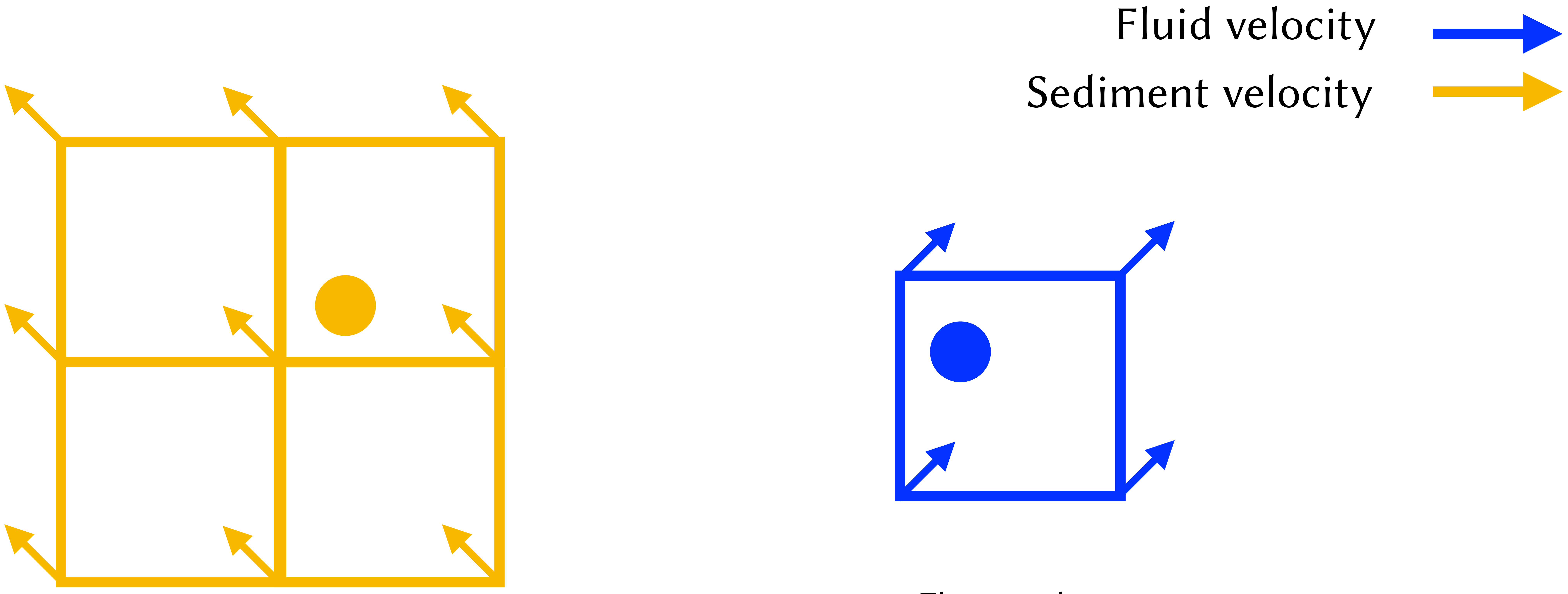


Stomakhin et al. 14



Cubic kernels and staggered grid

Our discretization

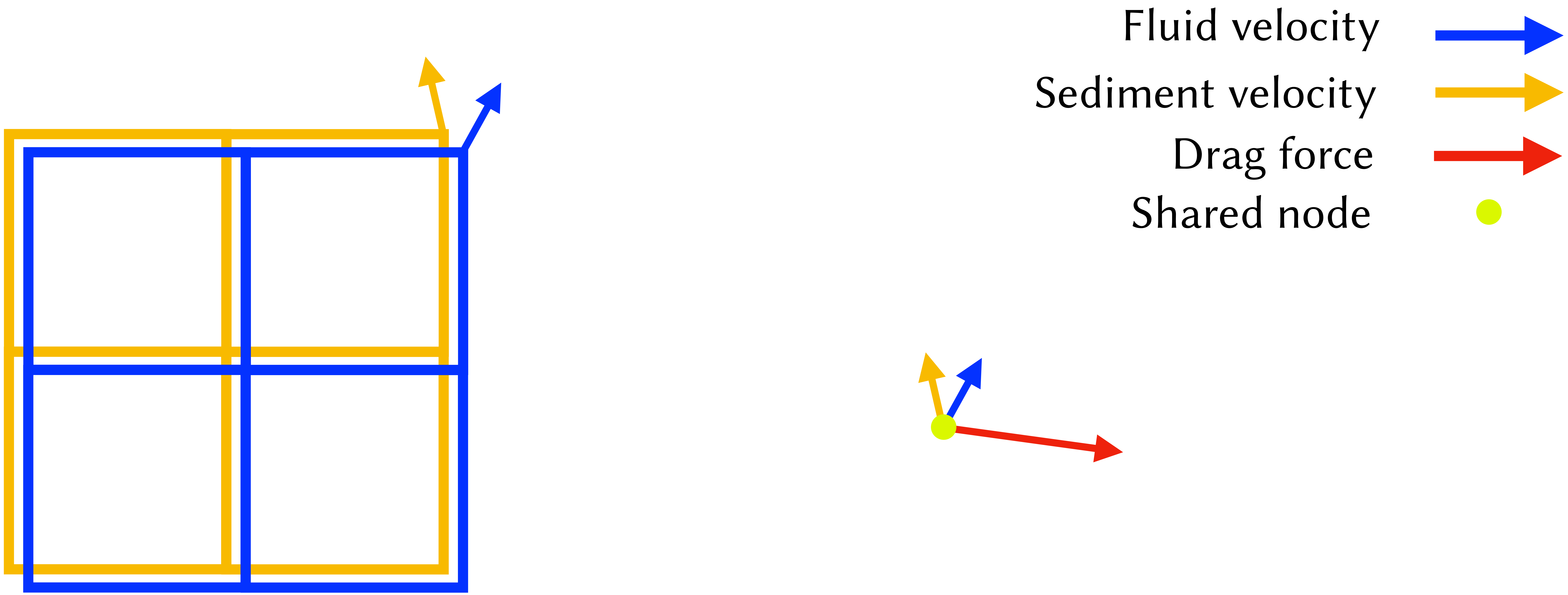


Zhang et al. 17

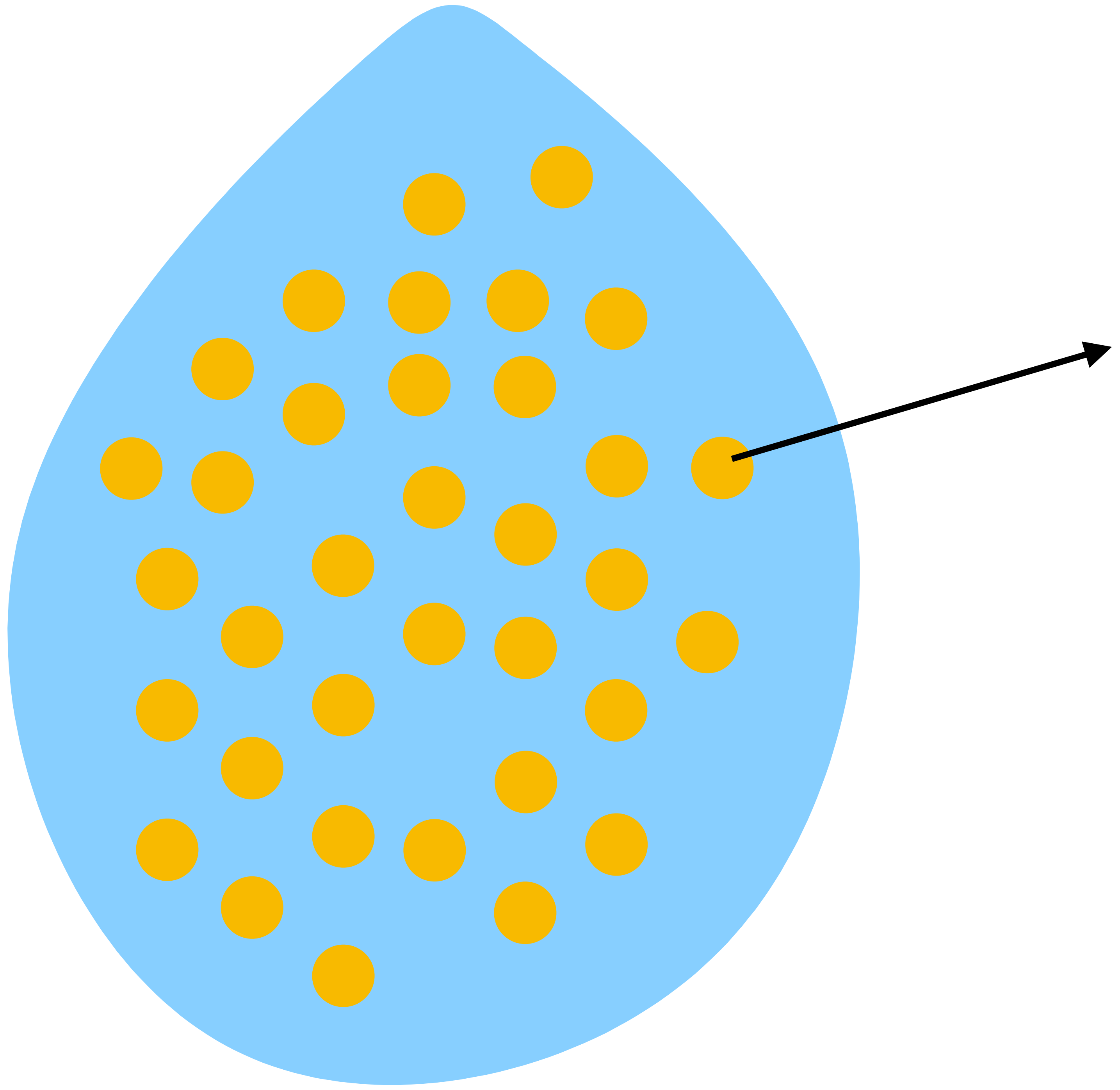
Solid

Fluid

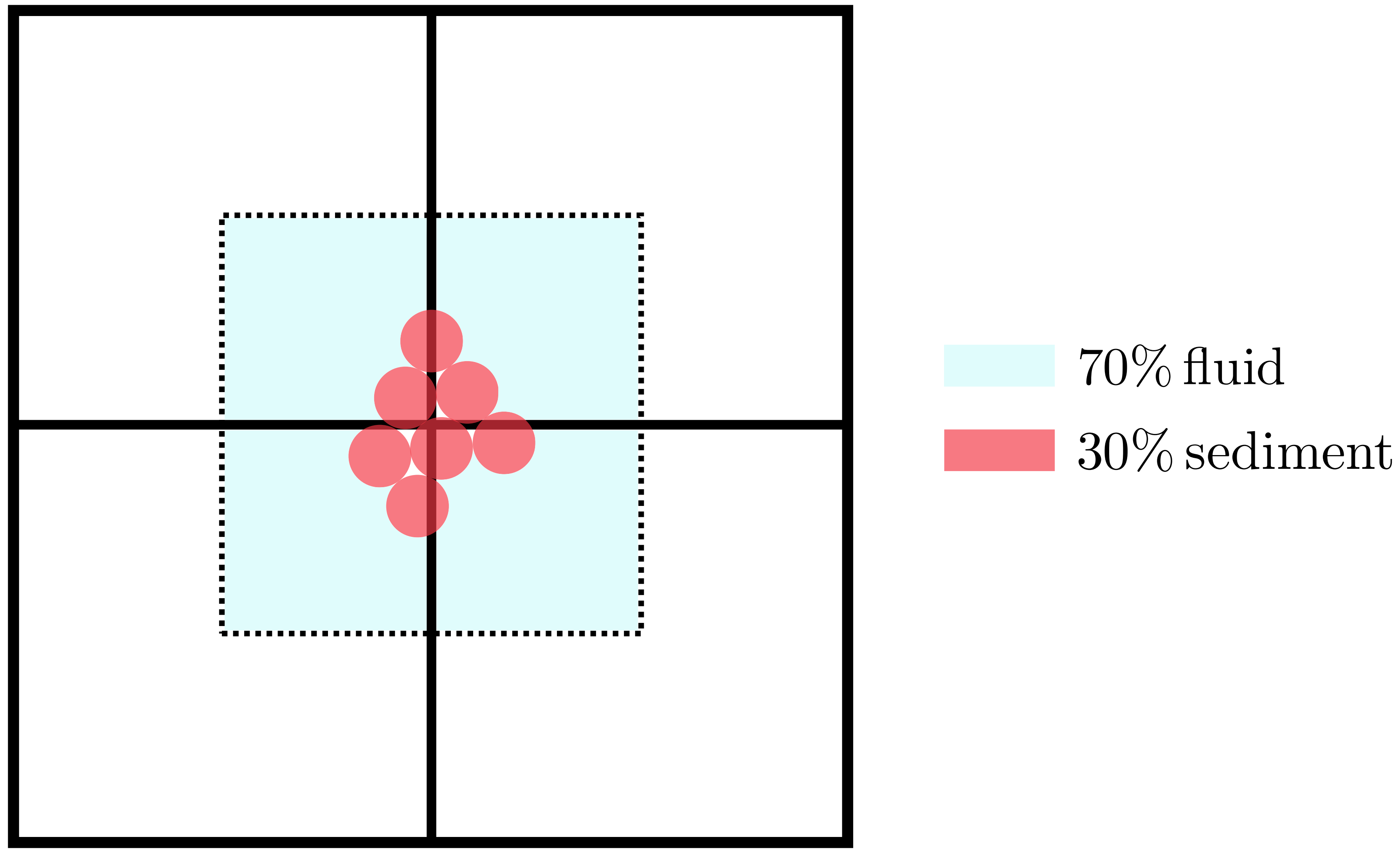
Coupling - drag force



Coupling - mixture theory



Coupling - volume fraction



$$\frac{V_i}{\Delta x^3}$$

Coupling - incompressibility

Fluid volume fraction ϵ

Fluid velocity $\vec{u}$

Solid volume fraction δ

Solid velocity $\vec{v}$

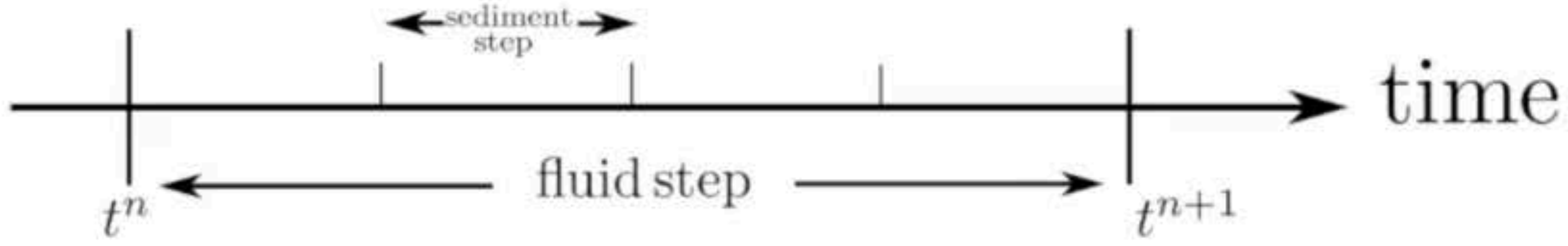
$$\nabla \cdot (\epsilon \vec{u} + \delta \vec{v}) = 0$$

$$\epsilon = 1, \delta = 0$$

$$\nabla \cdot \vec{u} = 0$$

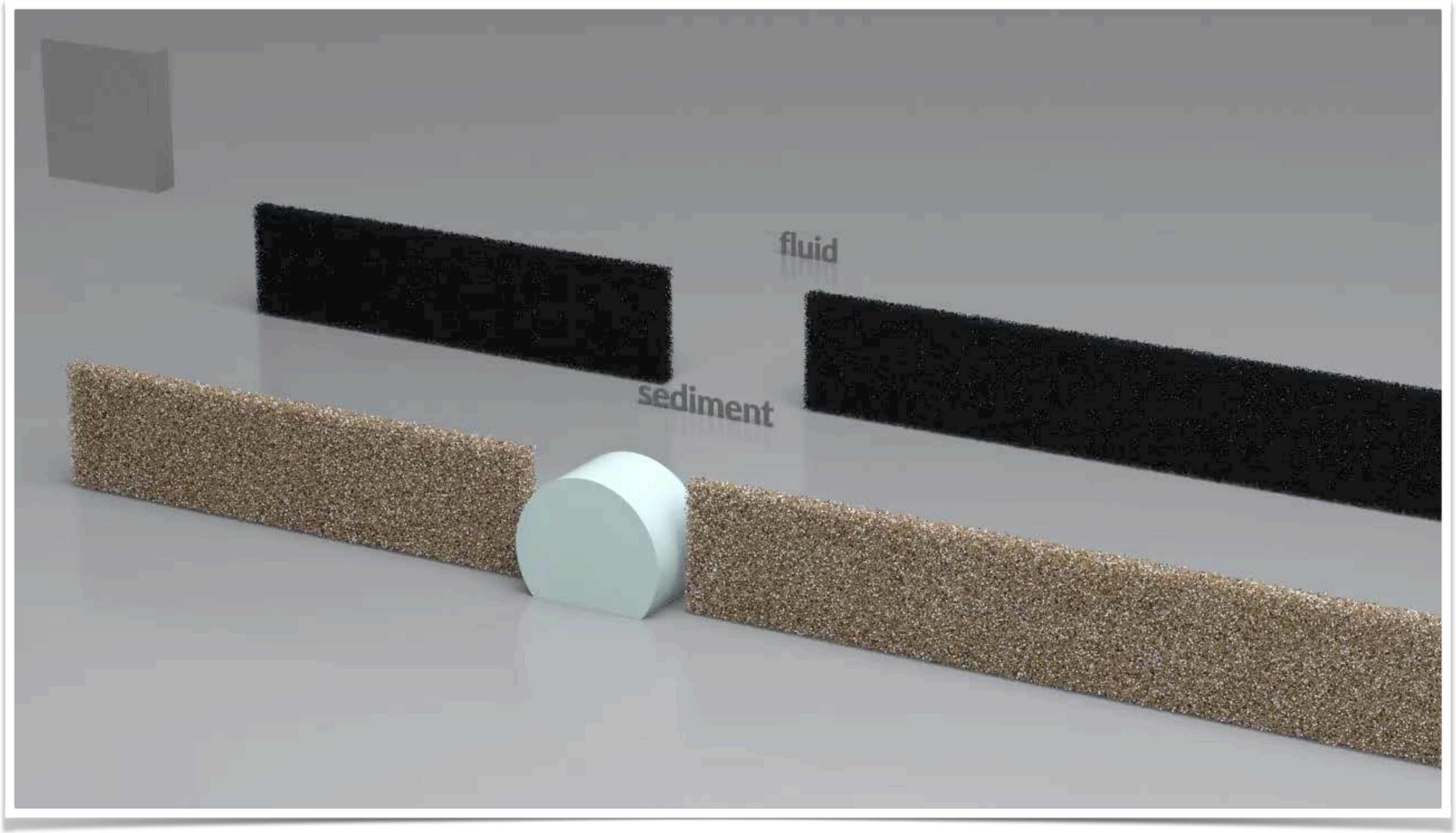
Challenges

- Variable coefficient multigrid preconditioner
- Sub-stepping

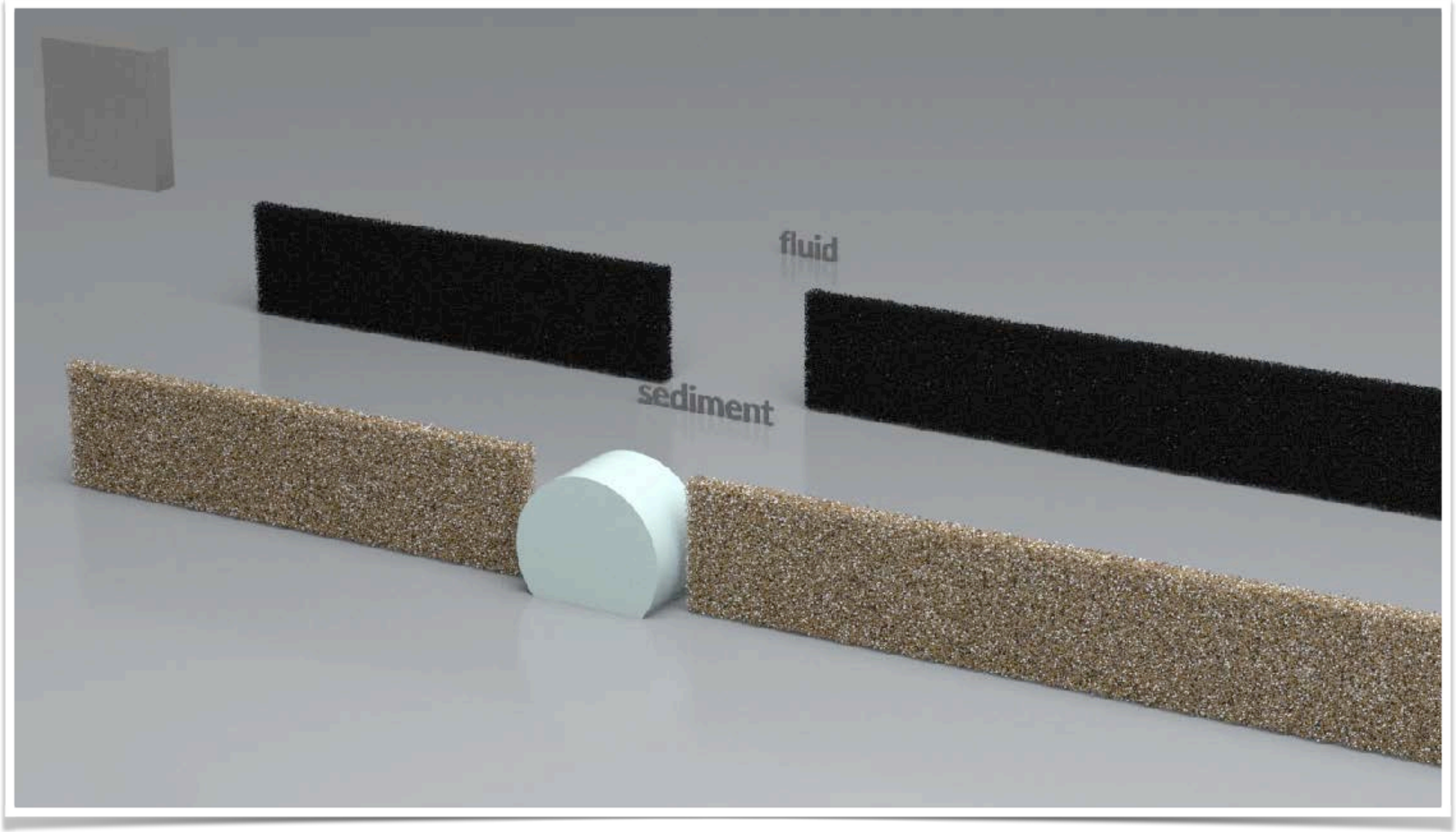


- Momentum conservation

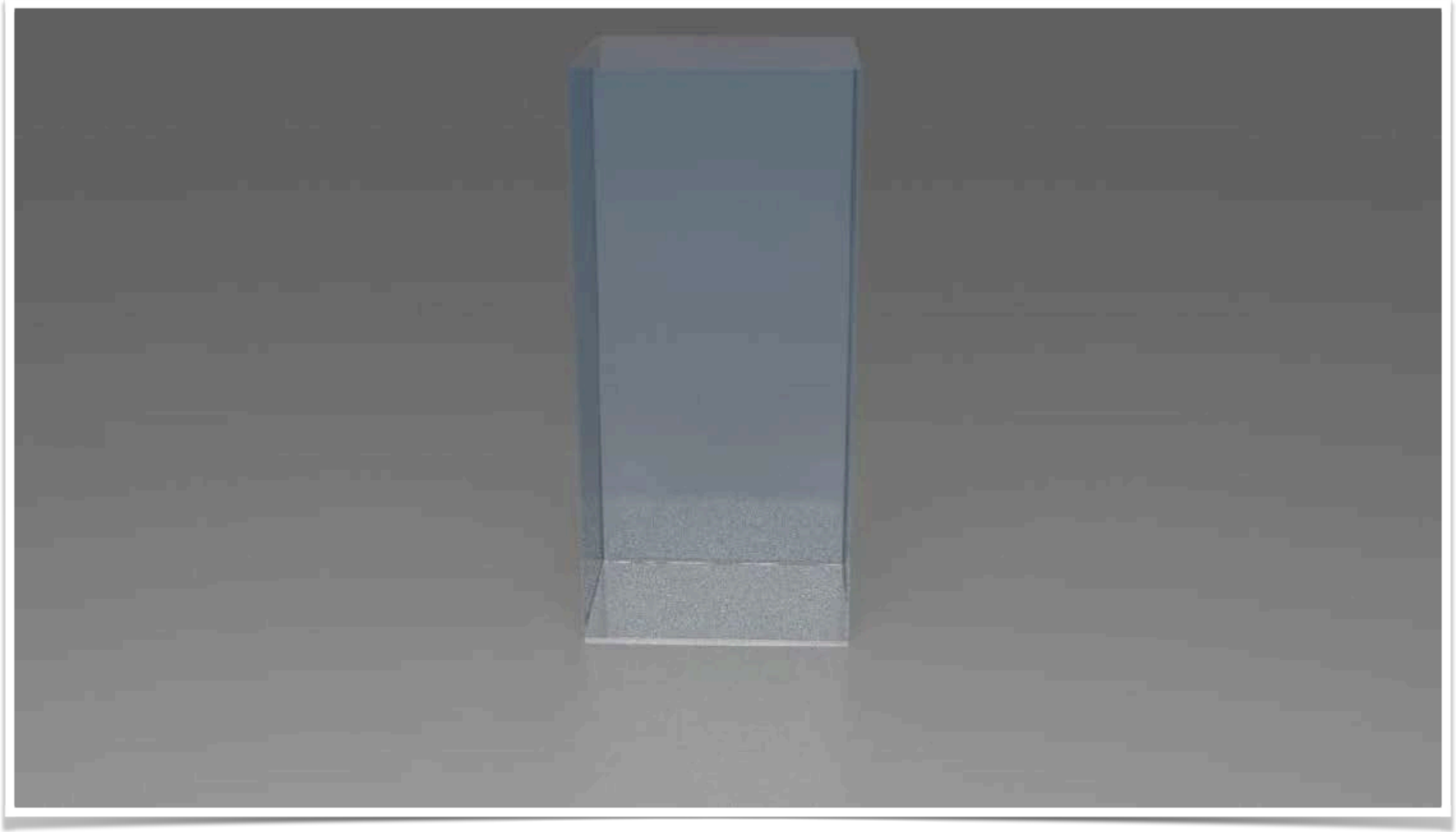
Results



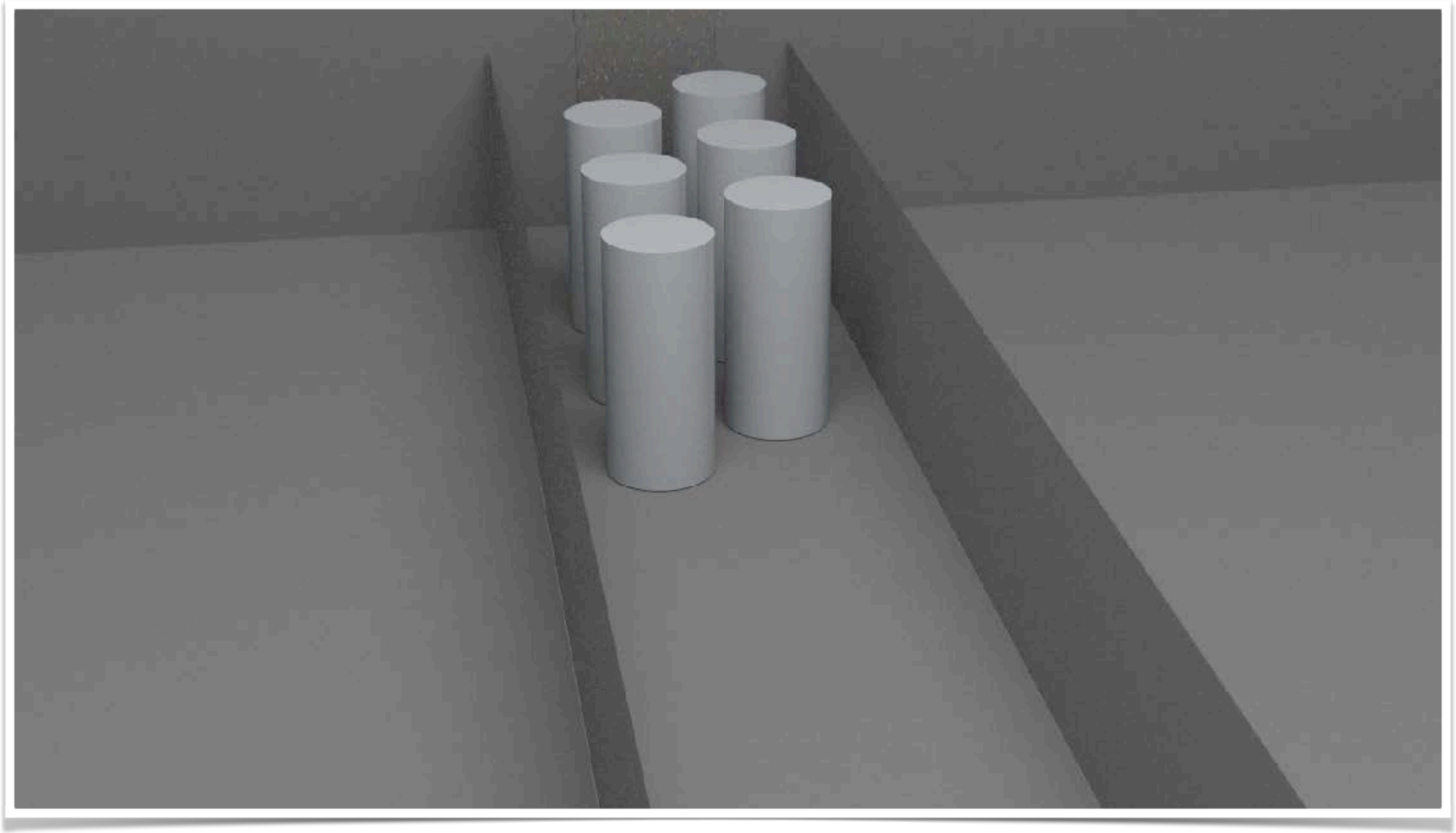
Results



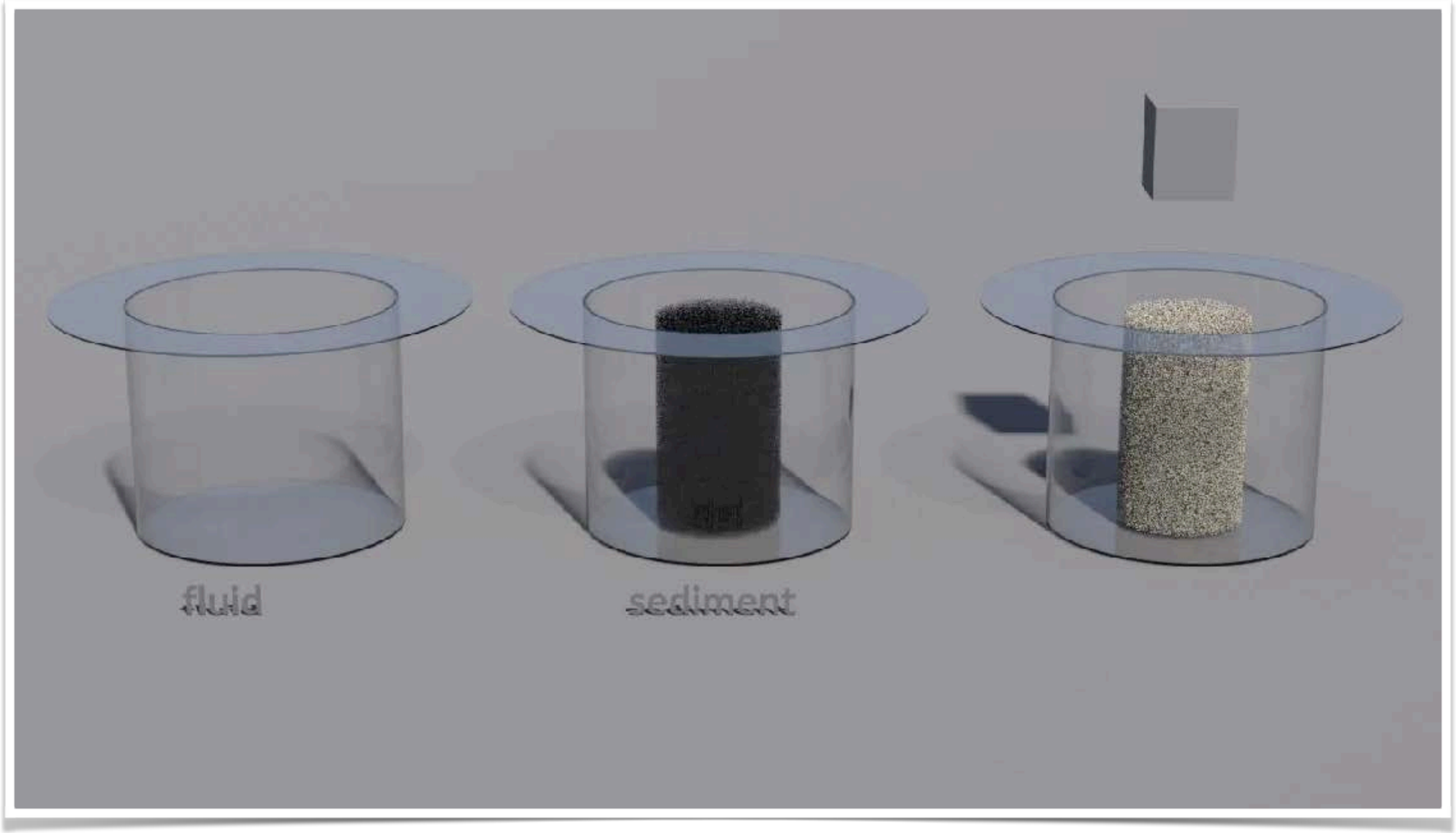
Results



Results



Results



An Adaptive Generalized Interpolation Material Point Method for Simulating Elastoplastic Materials

An Adaptive Generalized Interpolation Material Point Method for Simulating Elastoplastic Materials

MING GAO, University of Wisconsin Madison
ANDRE PRADHANA TAMPUBOLON, University of Pennsylvania
CHENFANFU JIANG, University of Pennsylvania
EFTYCHIOS SIFAKIS, University of Wisconsin Madison

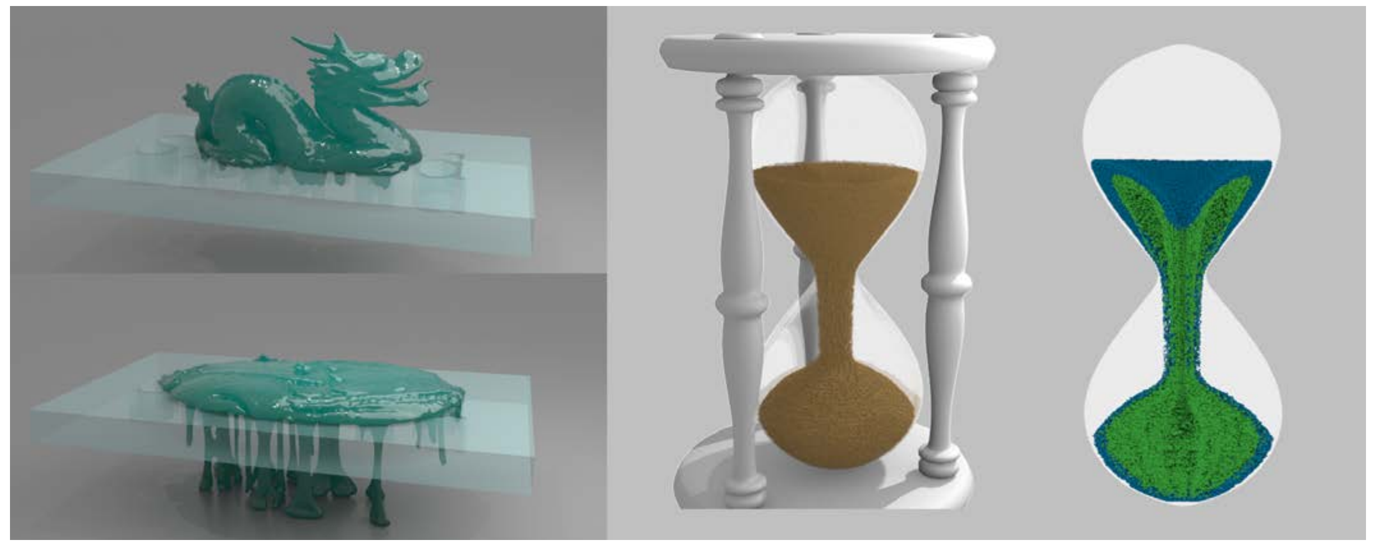


Fig. 1. Left: An elastoplastic model is dropped into a plane with a thin perforation pattern; our adaptive discretization allows the material to drip through. Right: Adaptive sand simulation with a visualization of the underlying grid refinement. We color refined particles with blue and coarse ones with green.

We present an adaptive Generalized Interpolation Material Point (GIMP) method for simulating elastoplastic materials. Our approach allows adaptive refining and coarsening of different regions of the material, leading to an efficient MPM solver that concentrates most of the computation resources in specific regions of interest. We propose a C^1 continuous adaptive basis function that satisfies the partition of unity property and remains non-negative throughout the computational domain. We develop a practical strategy for particle-grid transfers that leverages the recently introduced SPGrid data structure for storing sparse multi-layered grids. We demonstrate the robustness and efficiency of our method on the simulation of various elastic and plastic materials. We also compare key kernel components to uniform grid MPM solvers to highlight performance benefits of our method.

CCS Concepts: • **Computing methodologies → Physical simulation**;

Additional Key Words and Phrases: Material Point Method (MPM), Generalized Interpolation Material Point (GIMP), Adaptive grids, Elastoplasticity

DOI: 10.1145/3130800.3130879

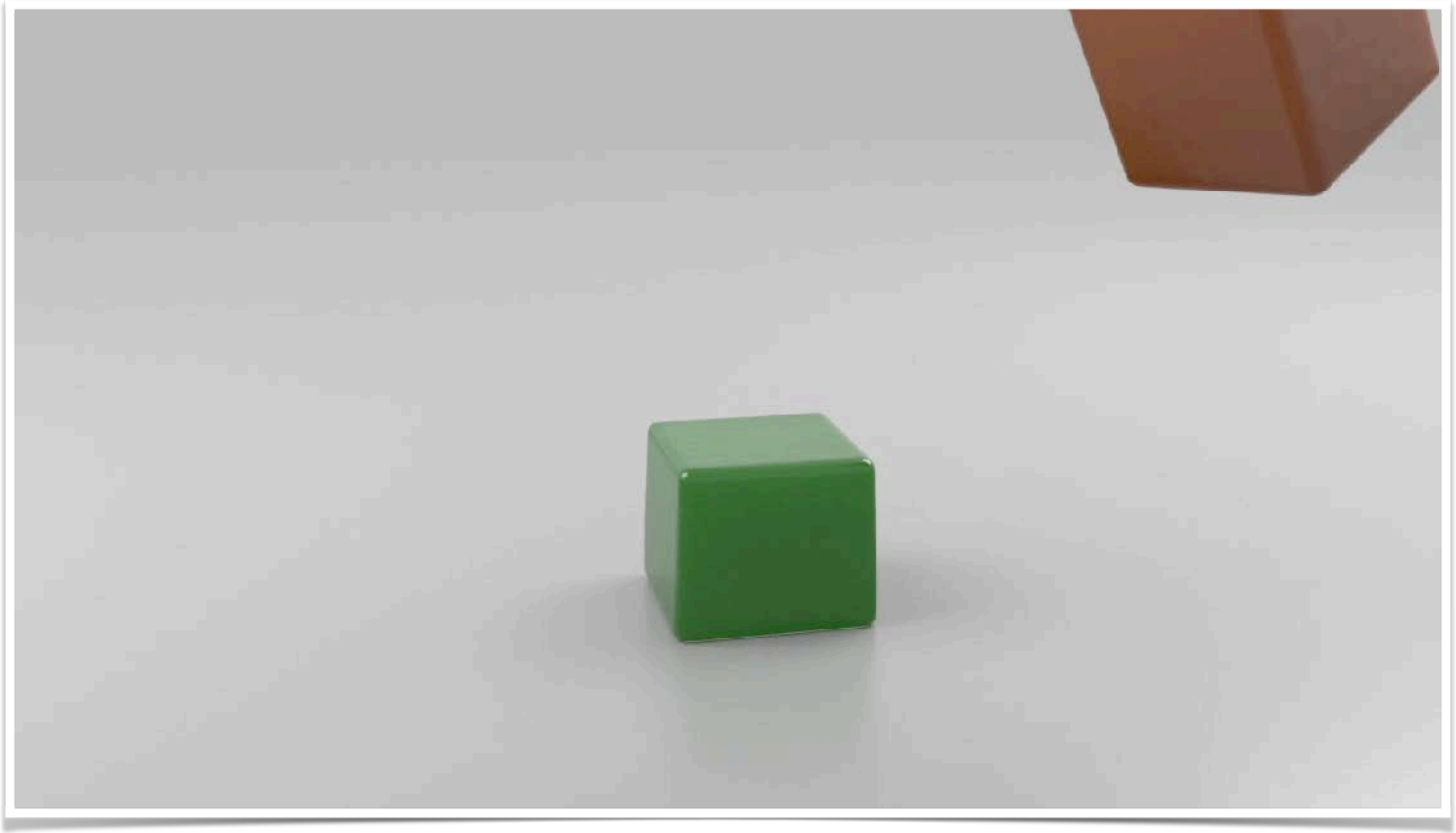
ACM Reference format:
Ming Gao, Andre Pradhana Tampubolon, Chenfanfu Jiang, and Eftychios Sifakis. 2017. An Adaptive Generalized Interpolation Material Point Method for Simulating Elastoplastic Materials. *ACM Trans. Graph.* 36, 6, Article 223 (November 2017), 12 pages.
DOI: 10.1145/3130800.3130879

1 INTRODUCTION

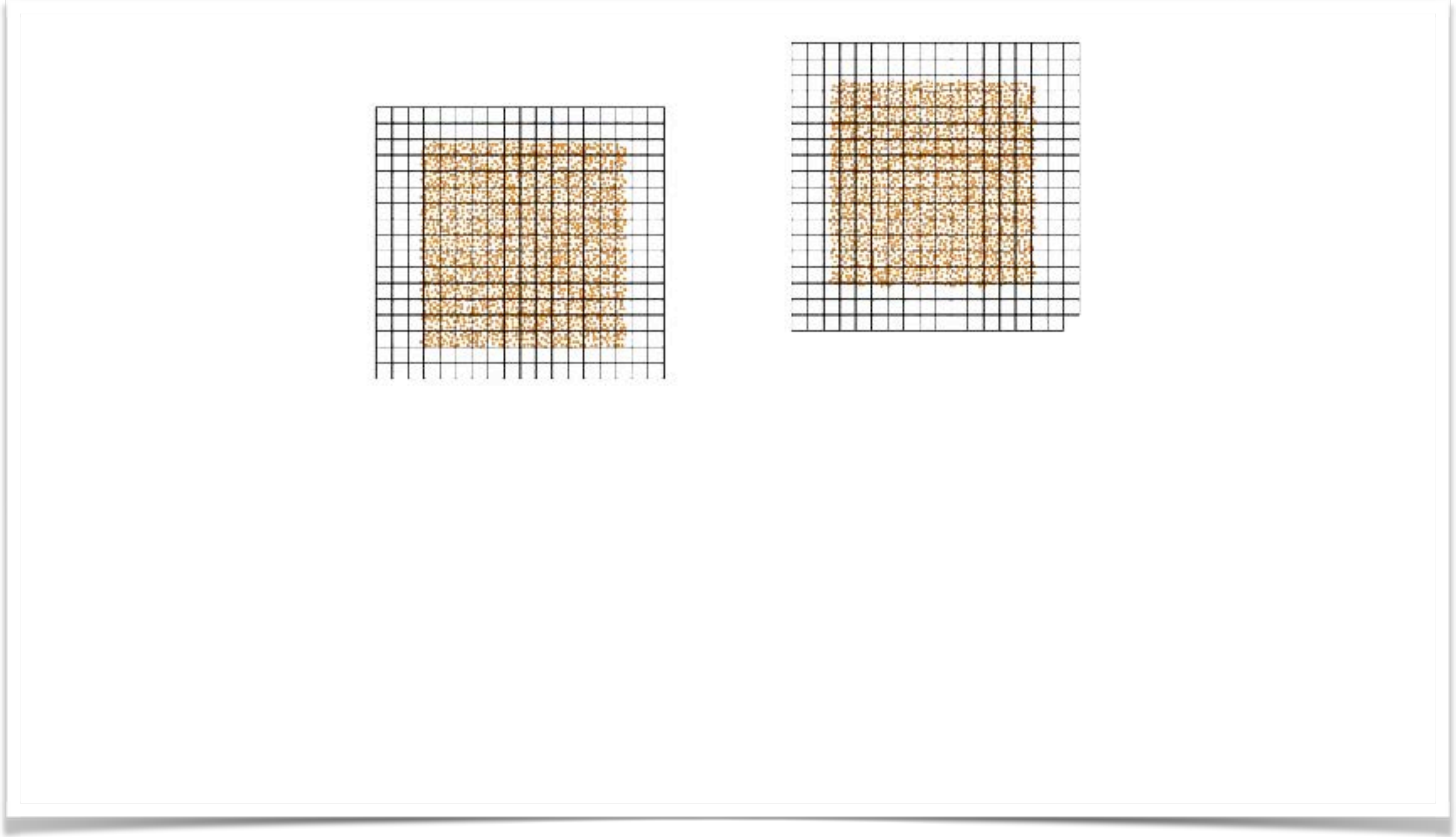
The Material Point Method (MPM) has been attracting considerable interest since it was introduced to the field of computer graphics by Stomakhin et al. [2013]. Combining advantages from both Lagrangian particle representation and Eulerian grid representation, MPM proves to be especially effective for animating elastoplastic materials undergoing large deformation or topology change [Jiang et al. 2016]. Despite its physical realism and geometrical convenience, a traditional MPM solver has several disadvantages. First, it is more computationally expensive than mesh-based Lagrangian approaches such as those based on Finite Element Methods (FEM) [Sifakis and Barbic 2012]. The bottleneck of MPM is usually the costly transfer operations between the particles and the grid. The cost of such transfer operations is particularly evident when we realize that MPM has to maintain the same grid resolution and a sufficient particle count throughout the simulation domain. The overhead of this process is highlighted in scenarios such as the example of drawing in a

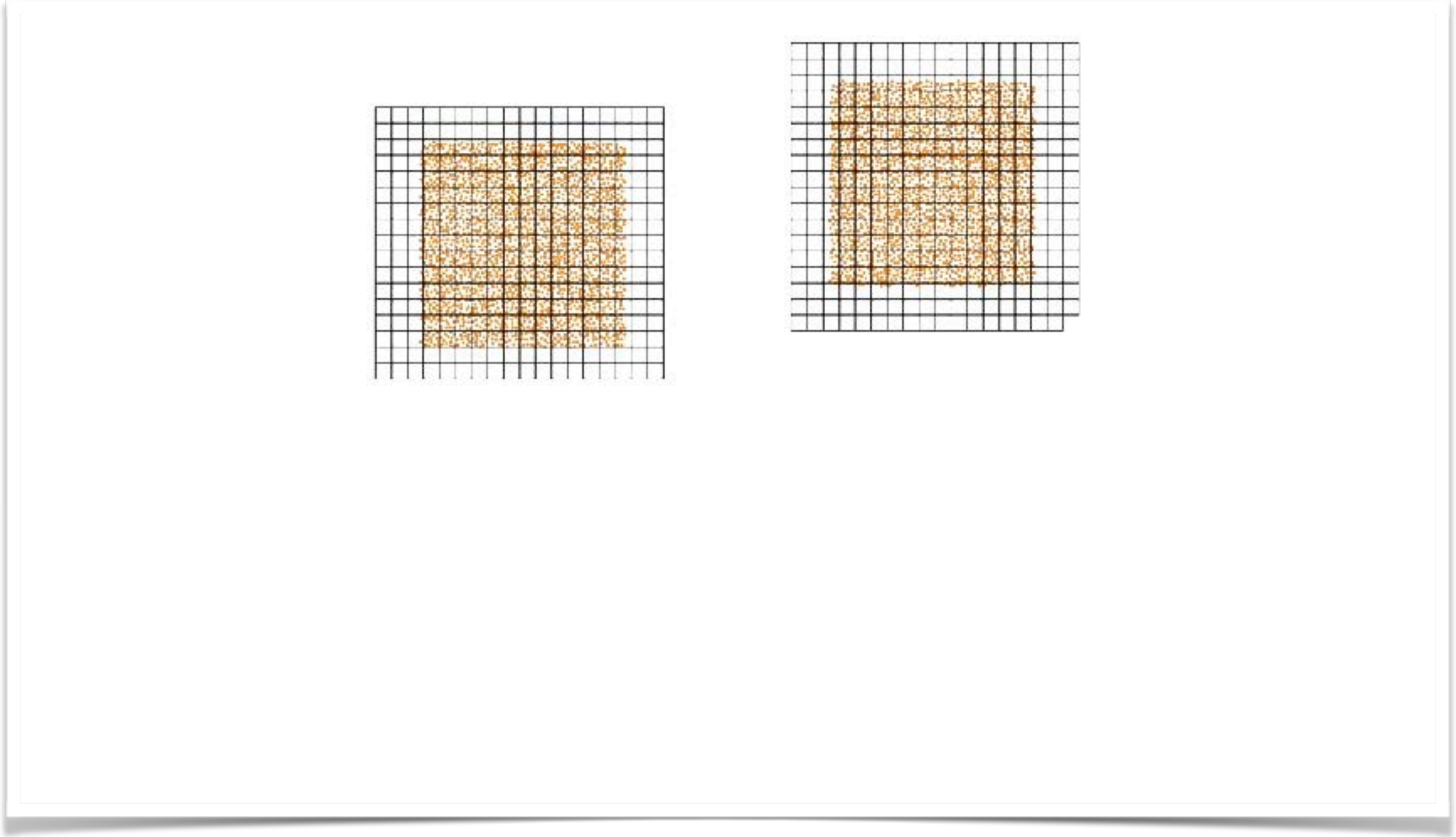
M. Gao, A. Tampubolon, C. Jiang, E. Sifakis
ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia), 2017

Motivation

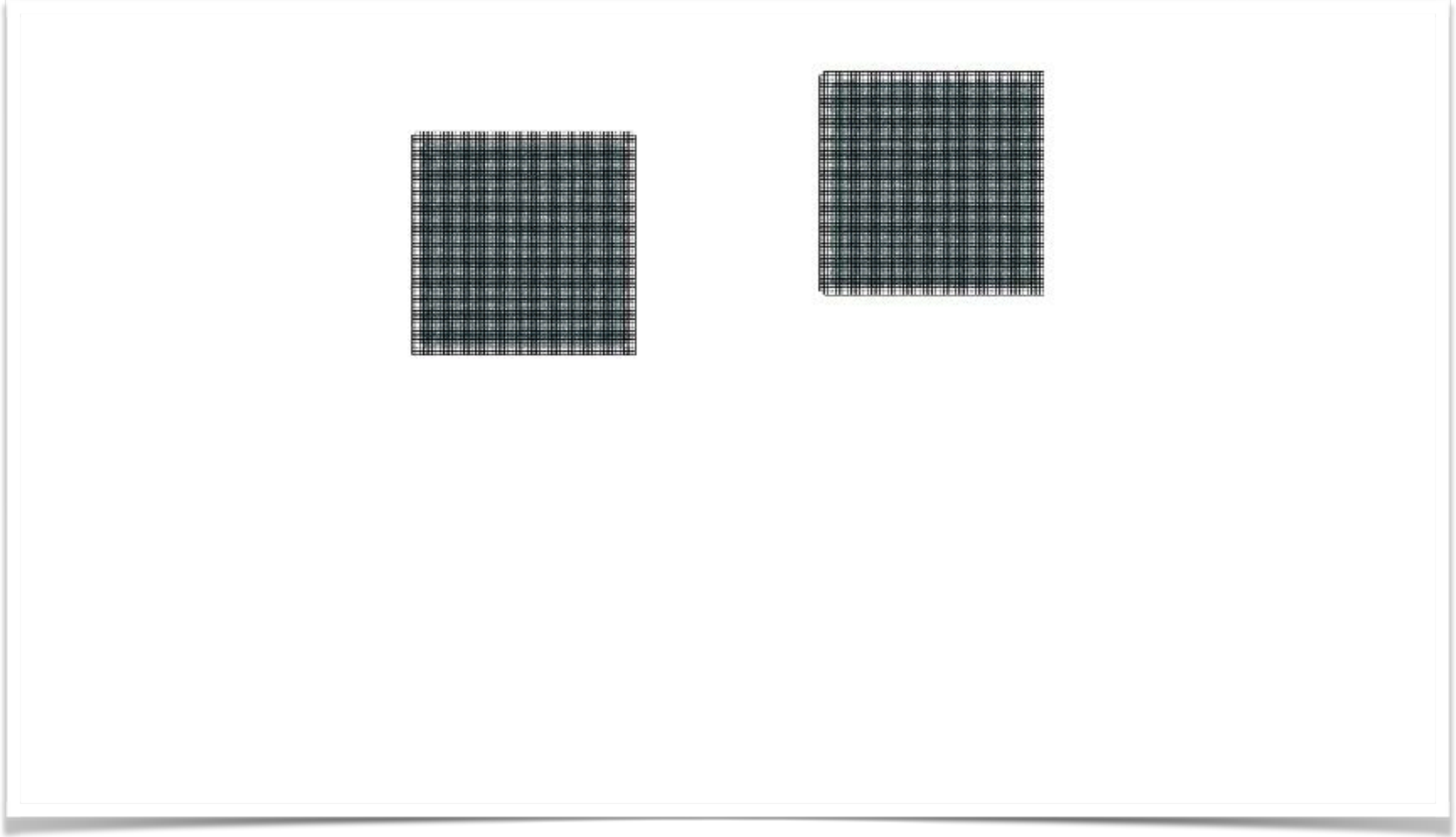


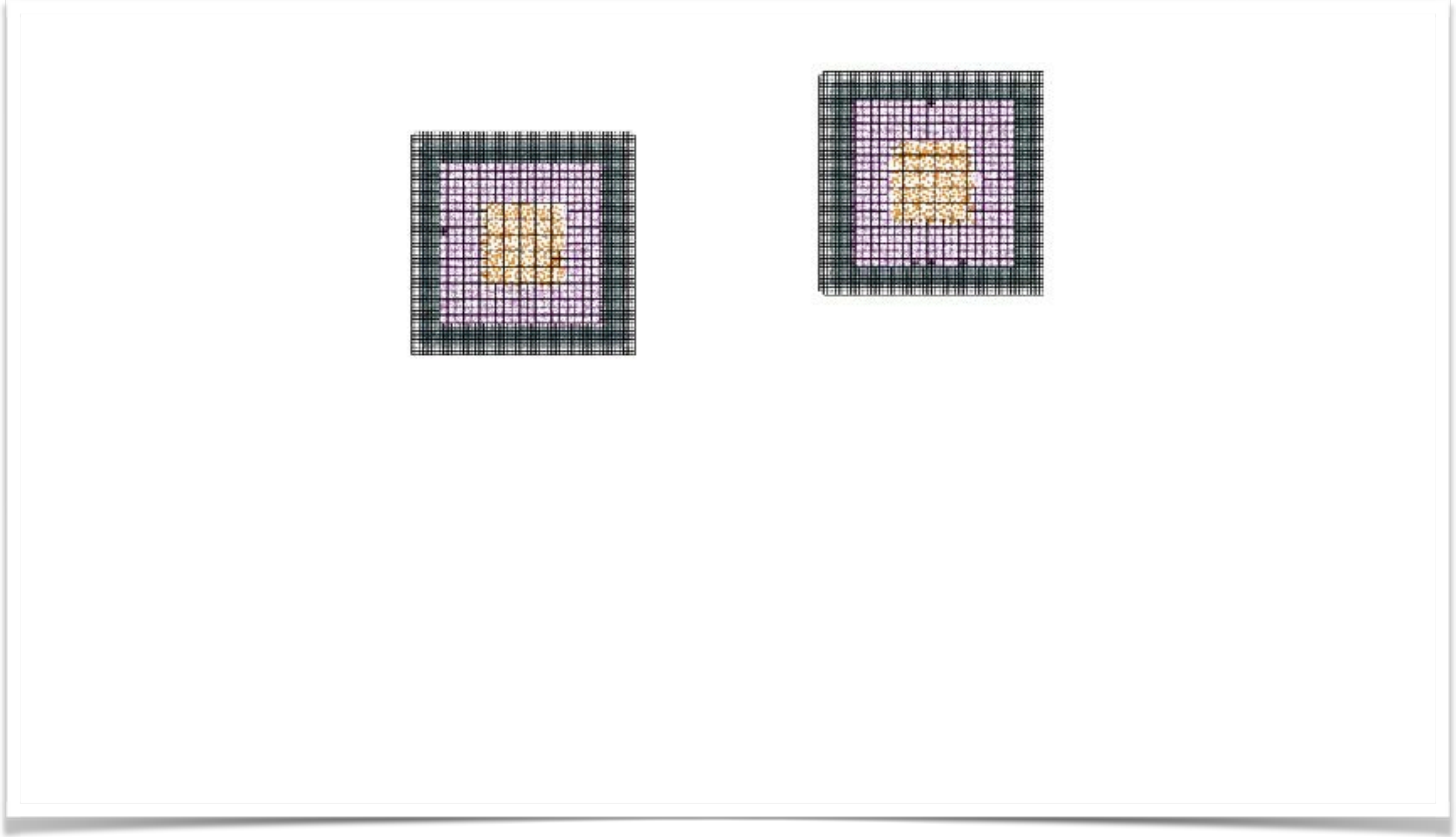
Motivation



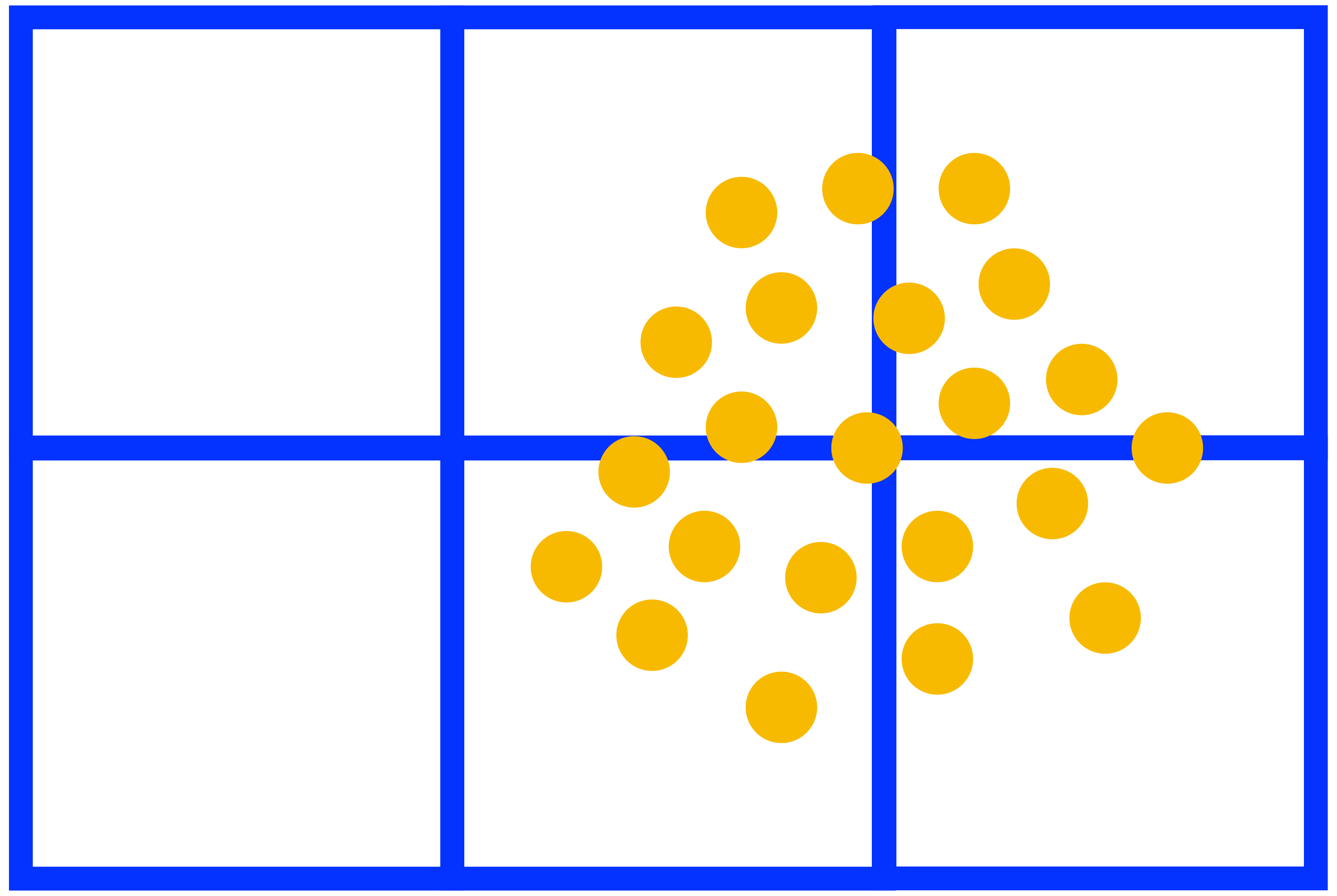


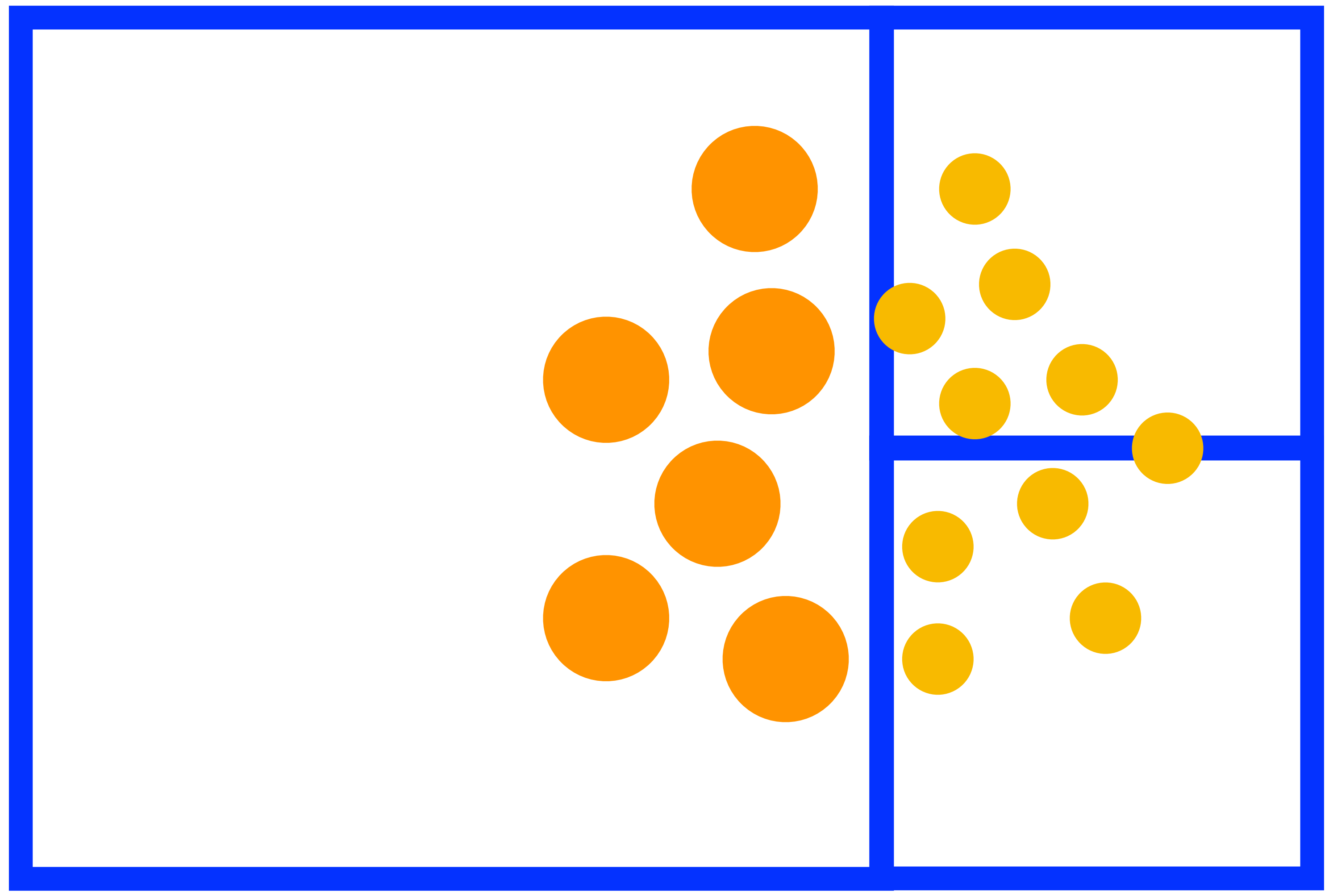
Motivation



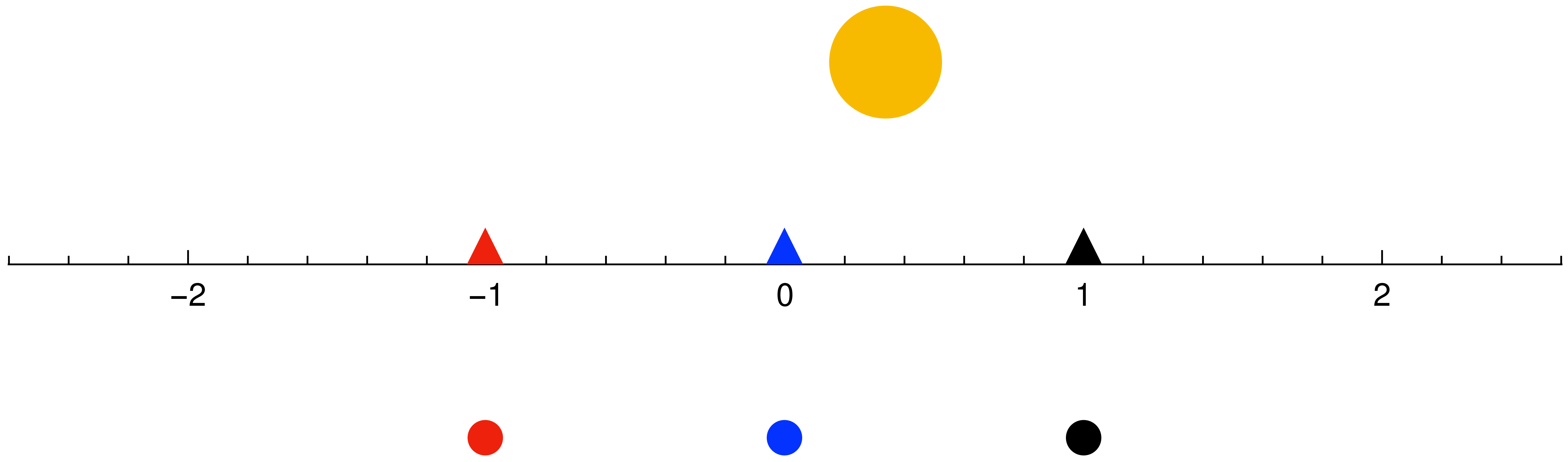


MPM adaptivity

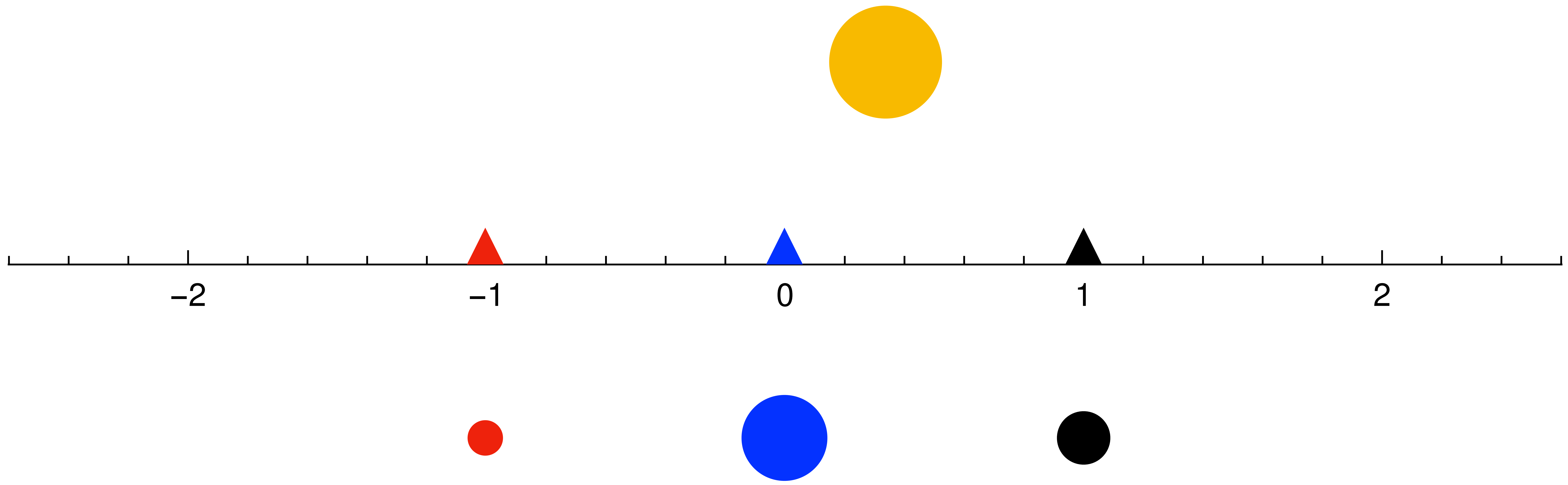




Transfer of mass



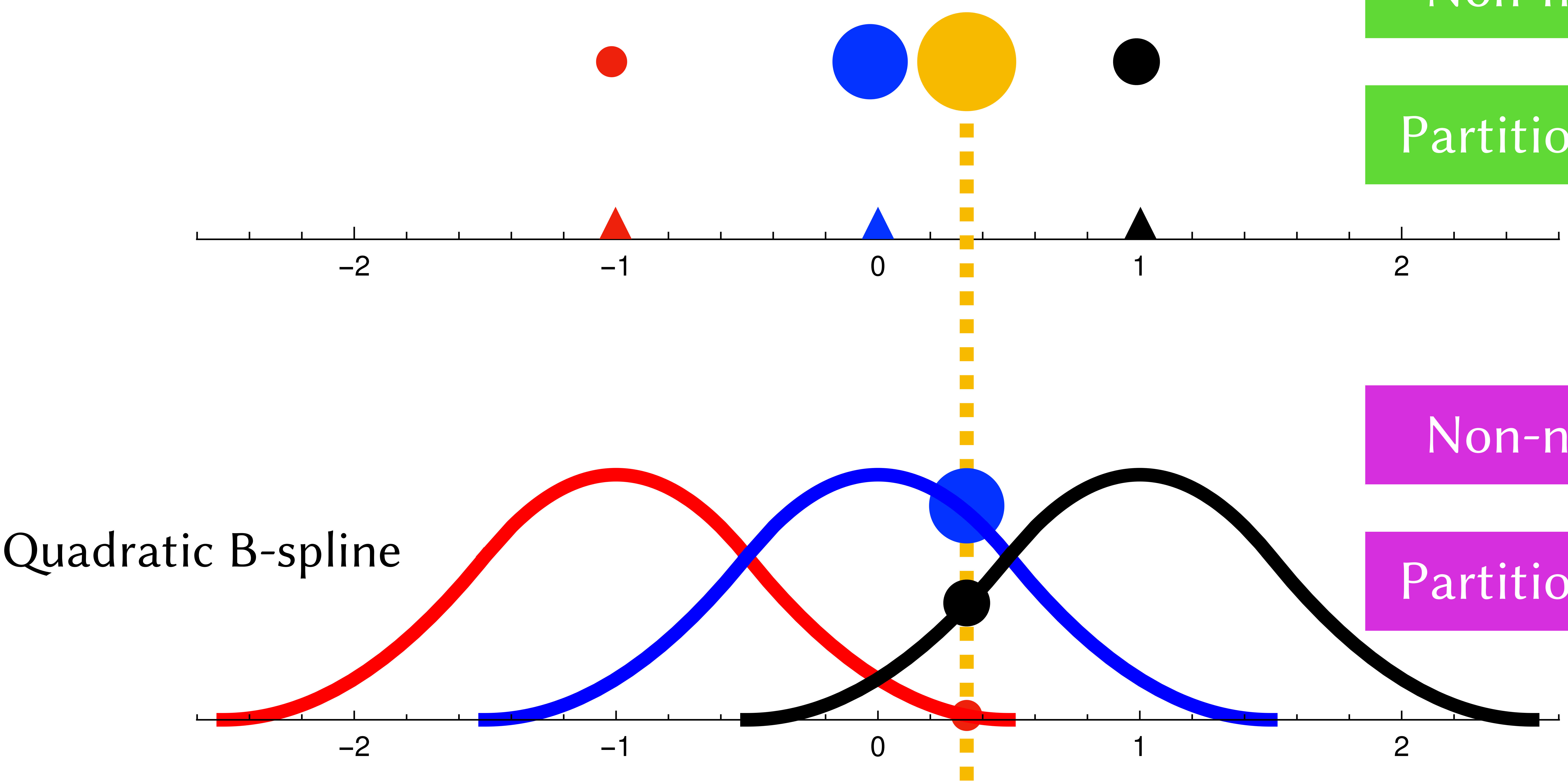
Transfer of mass



Non-negativity

Partition of unity

Transfer of mass



Non-negativity

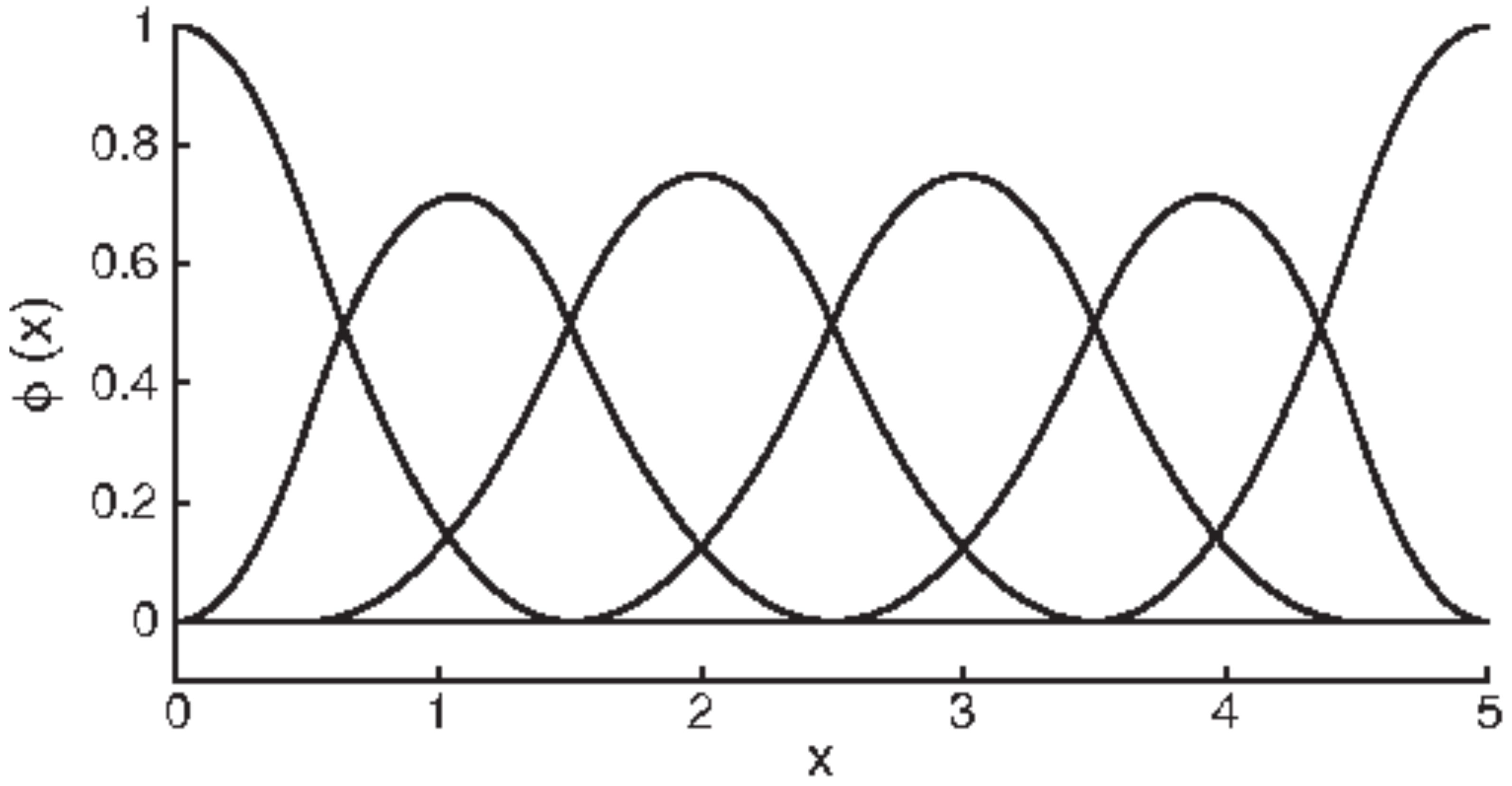
Partition of unity

Non-negativity

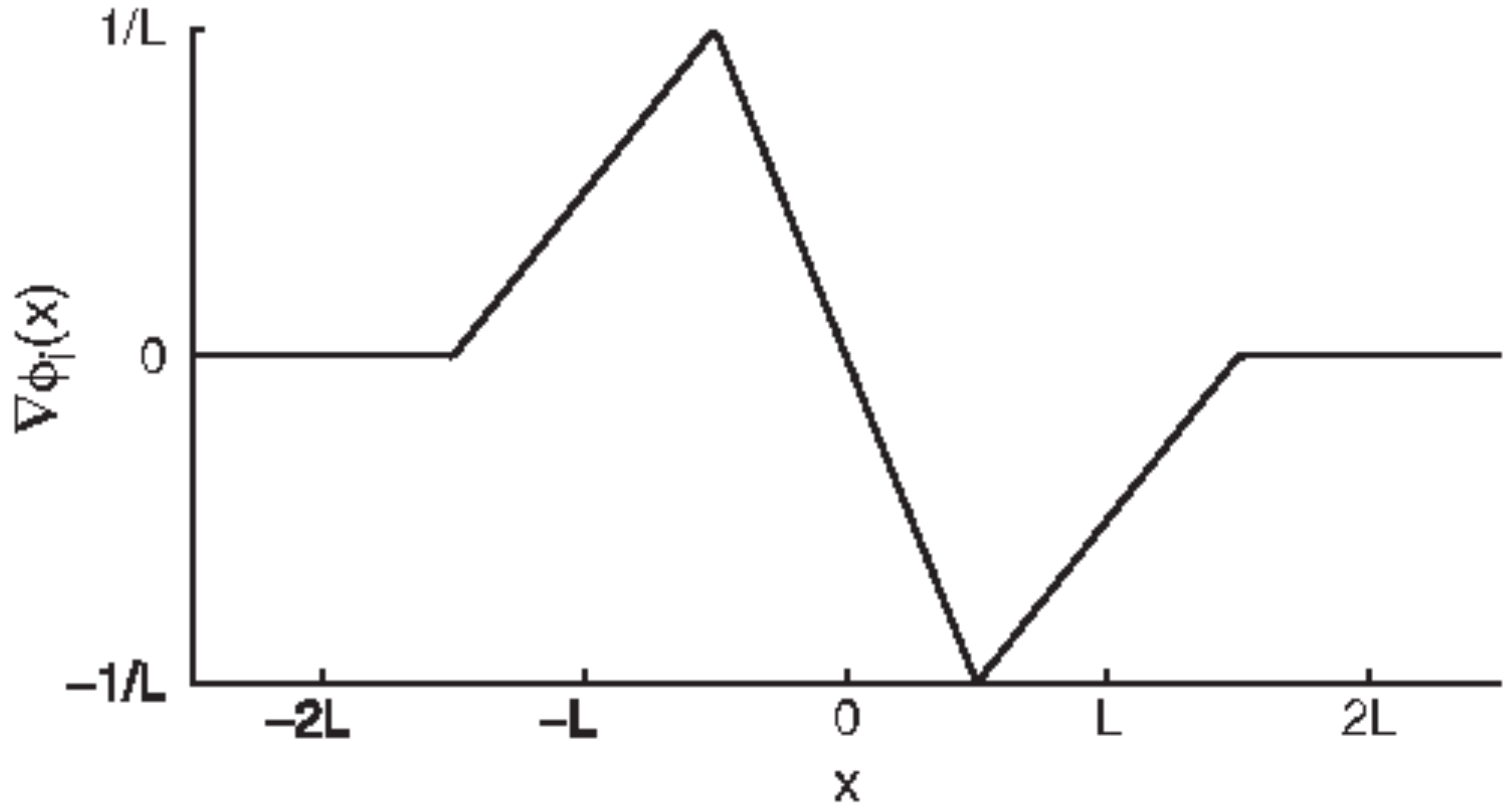
Partition of unity

C^1 continuity

Weight / mass



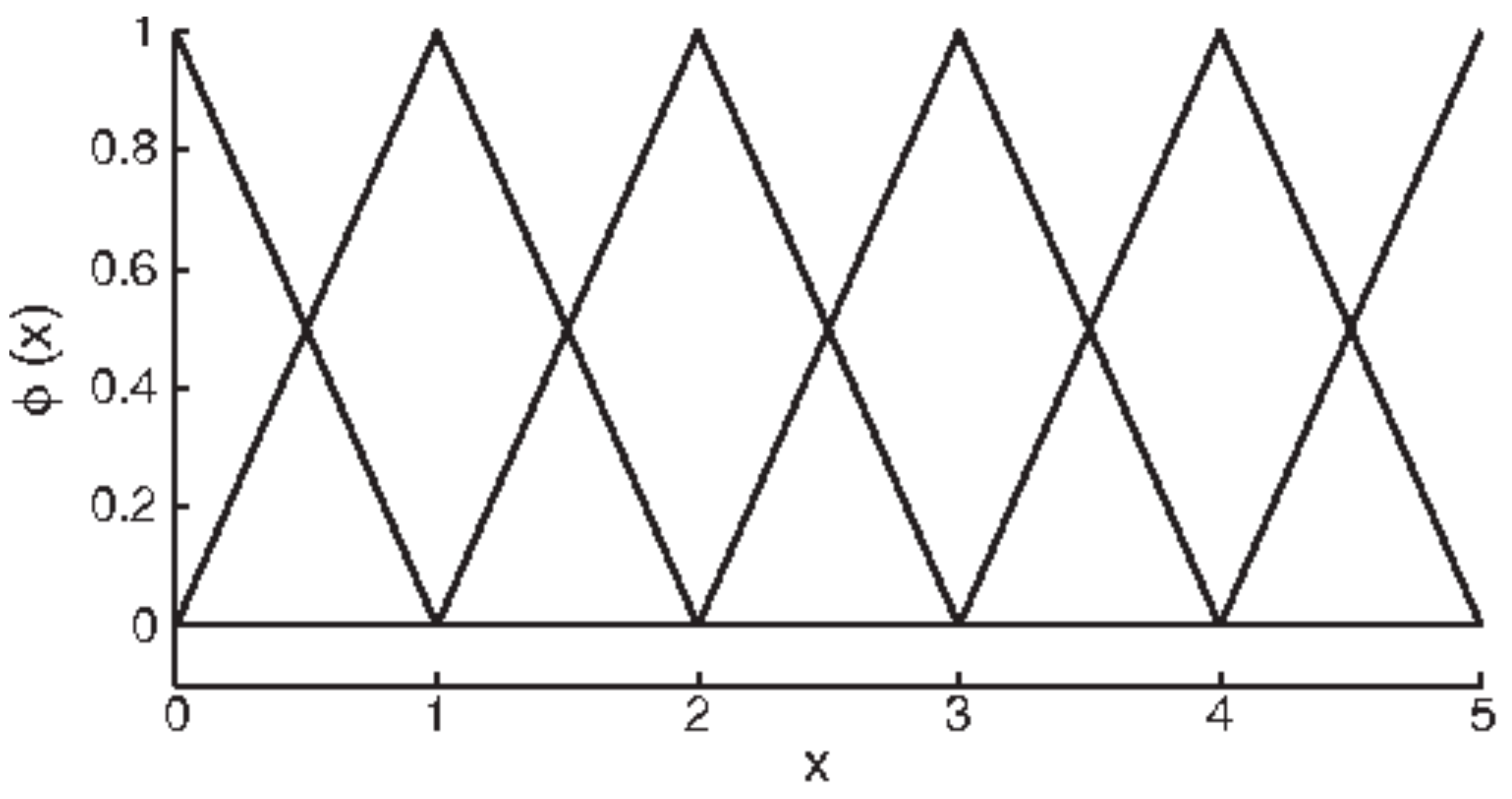
Weight gradient / force



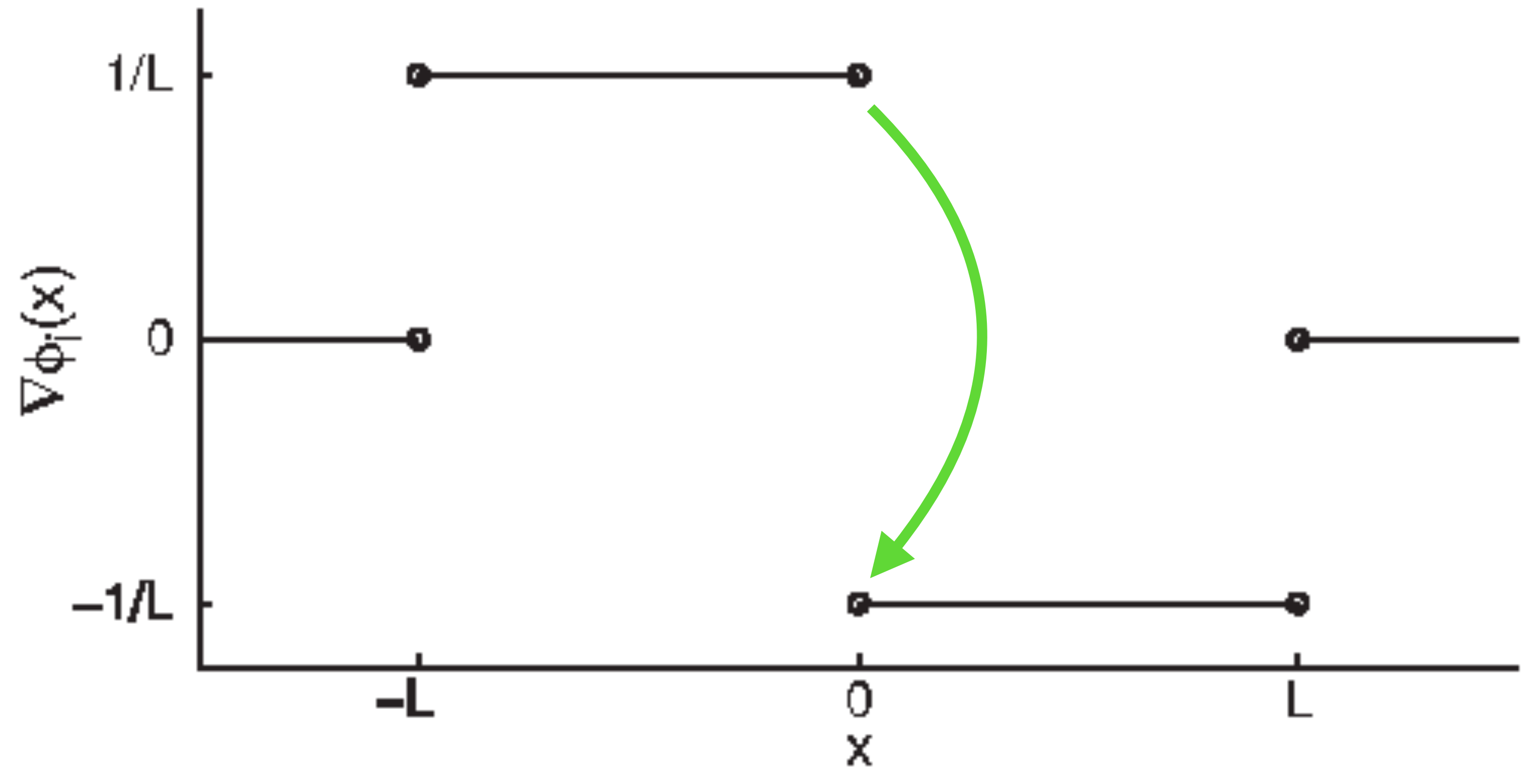
Steffen et al. 08

C^0 continuity

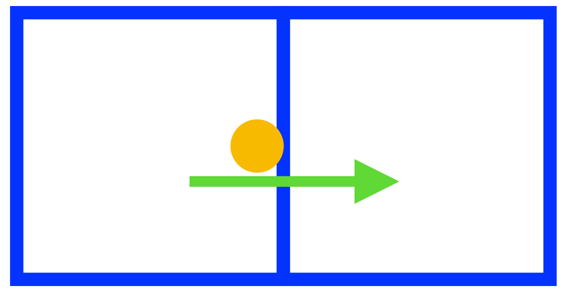
Weight / mass



Weight gradient / force

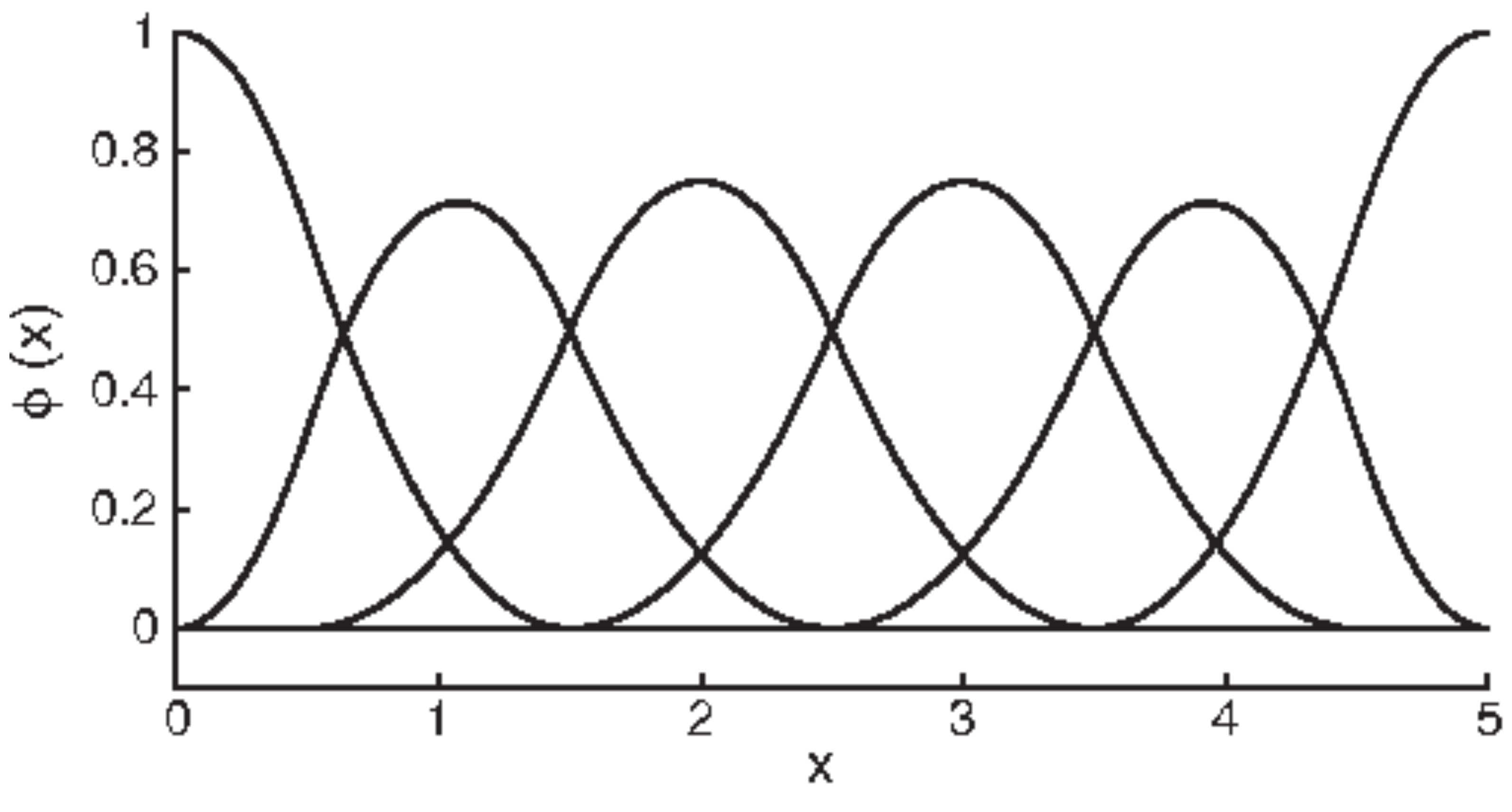


Steffen et al. 08

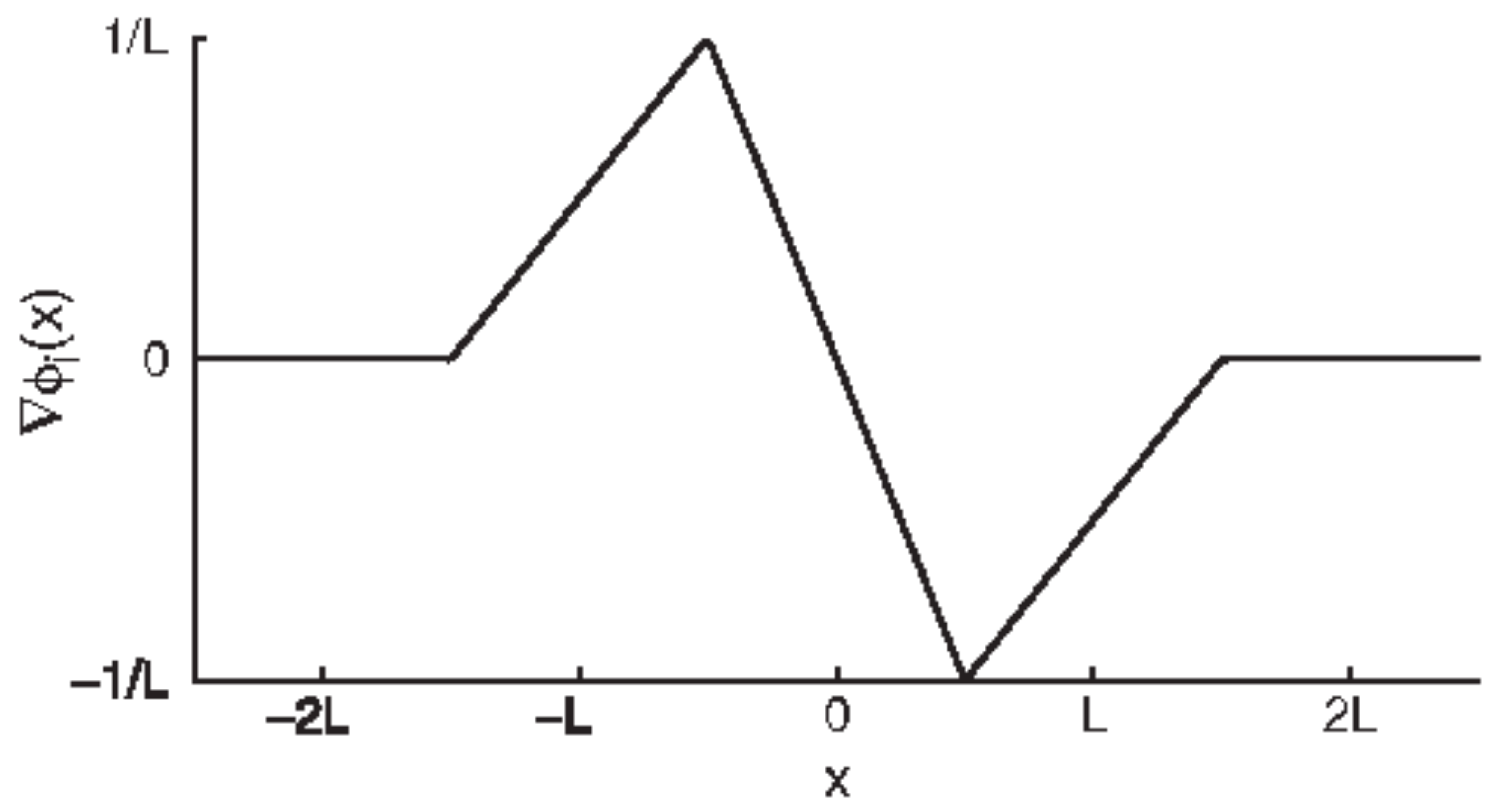


C^1 continuity in octree ?

Weight / mass



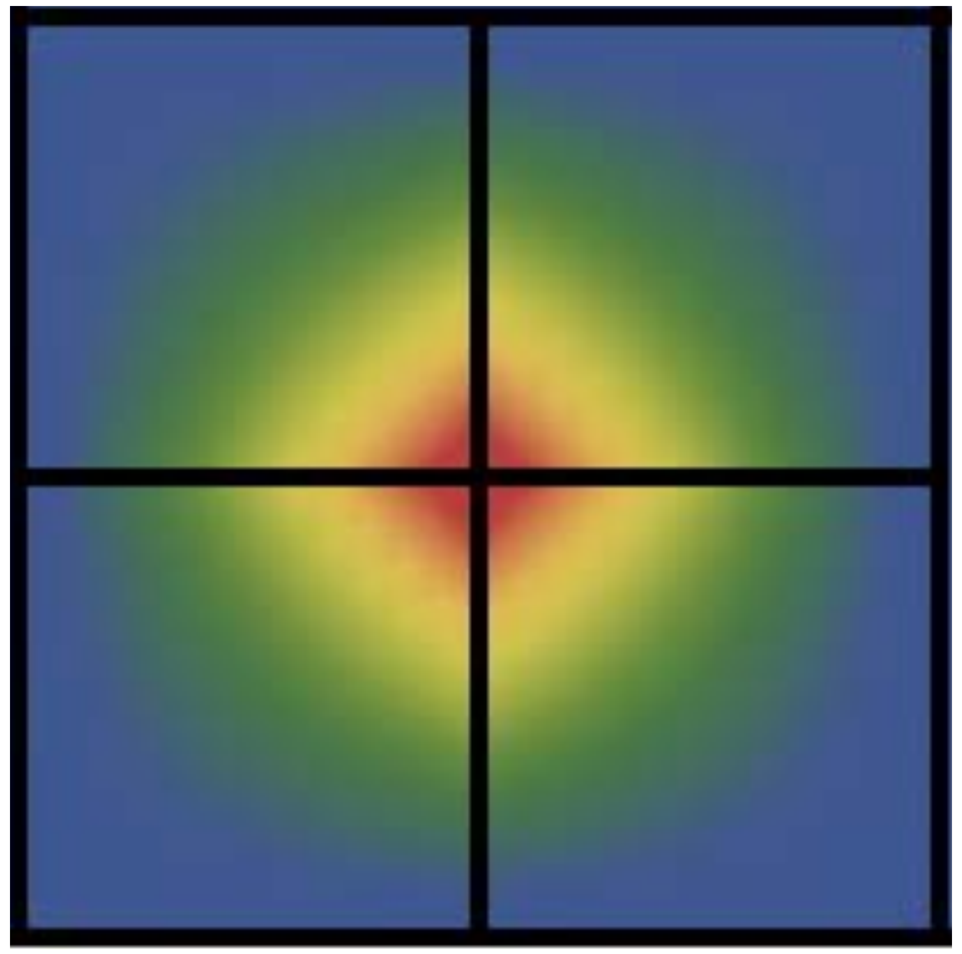
Weight gradient / force



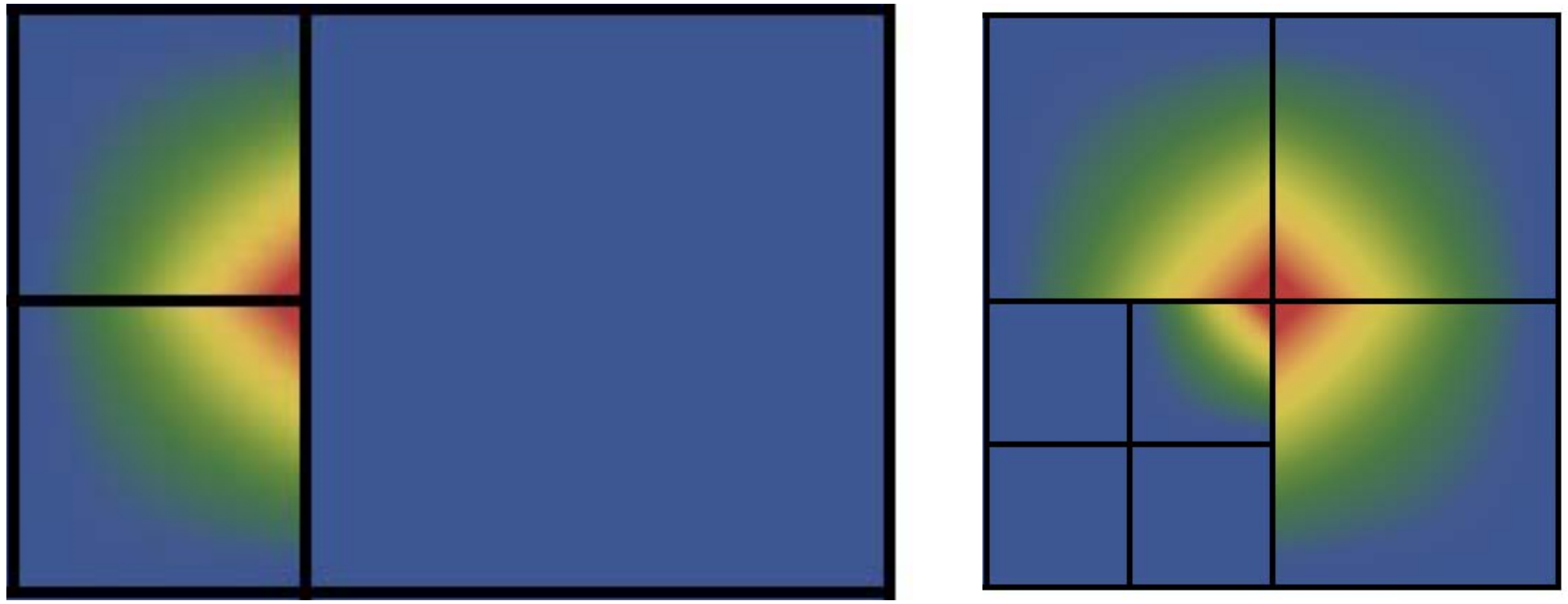
Steffen et al. 08

$C^0 \rightarrow C^1$

C^0 from uniform to quadtree



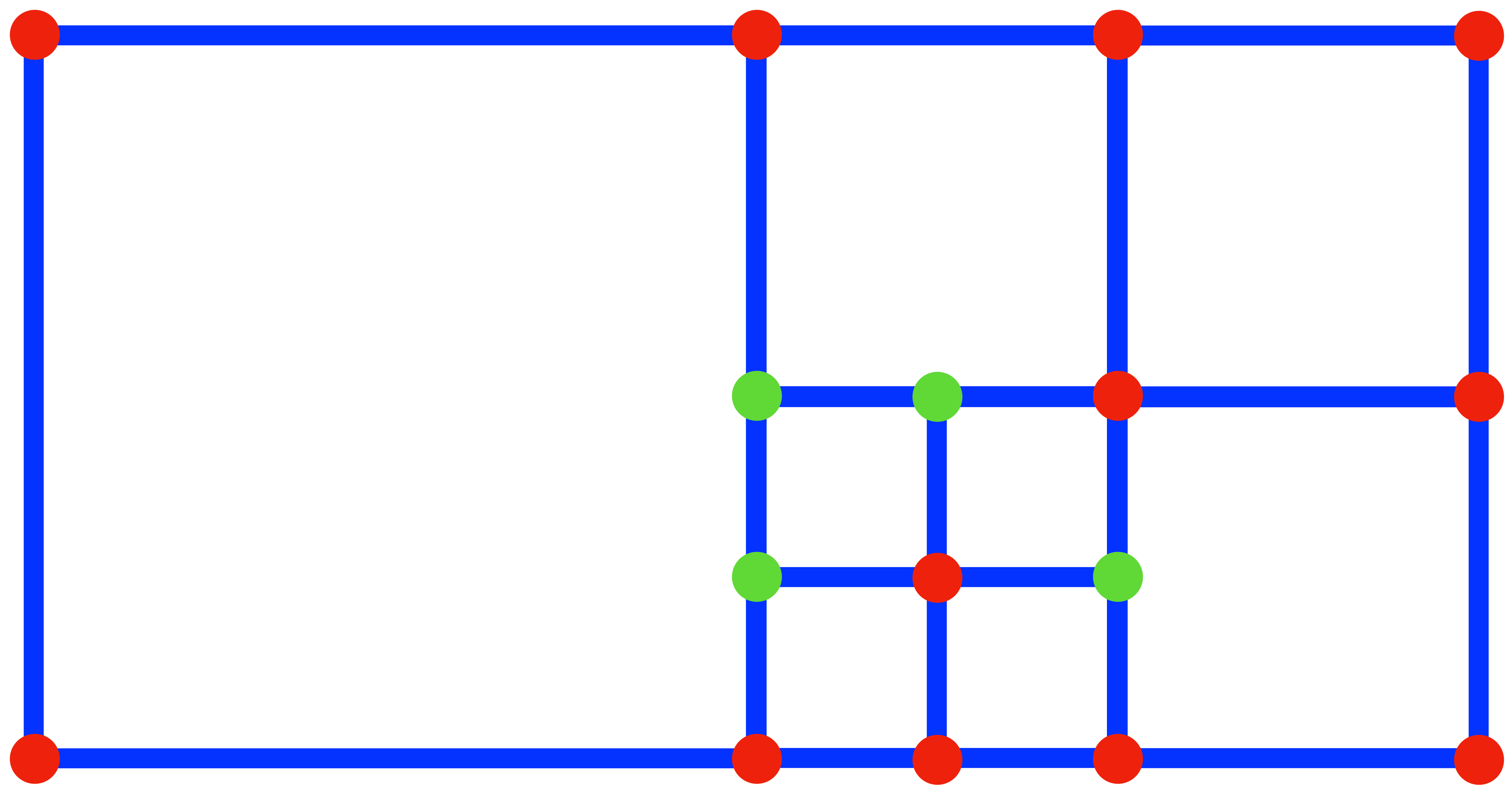
Uniform



Quadtree

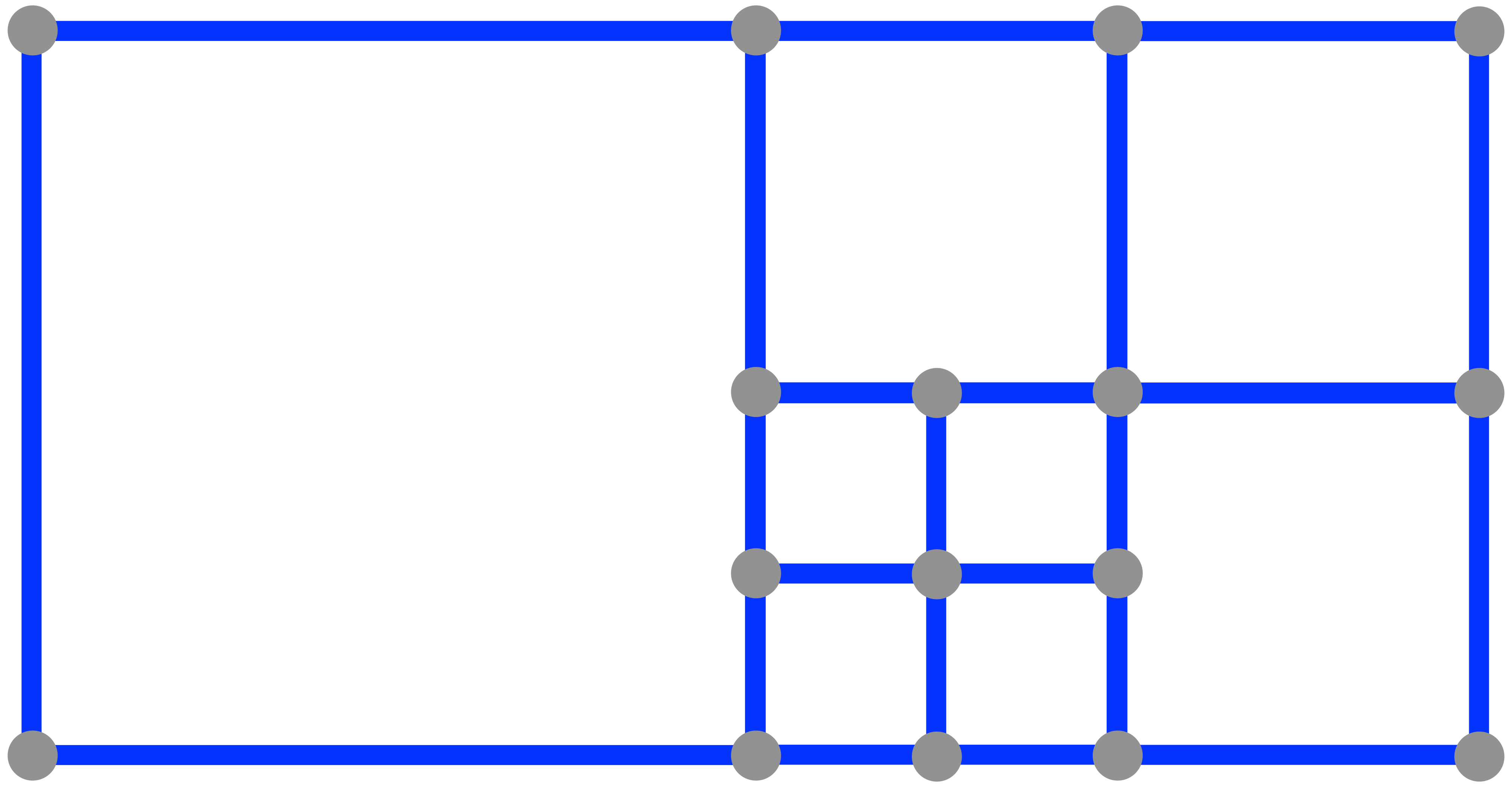
Embedding T-junctions

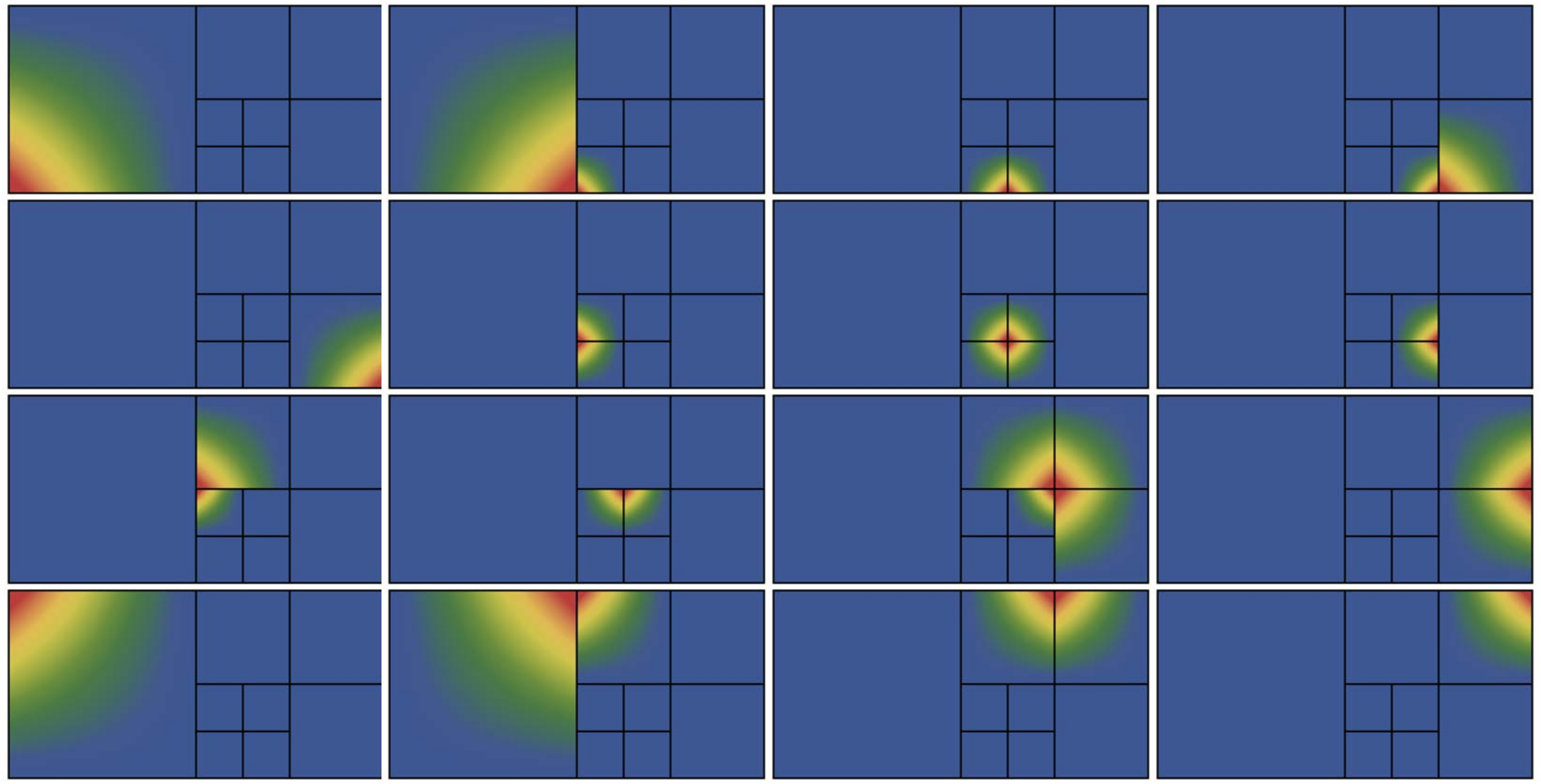
- DOF node
- Embedded node / T-junction node



Step1 - set all nodes free

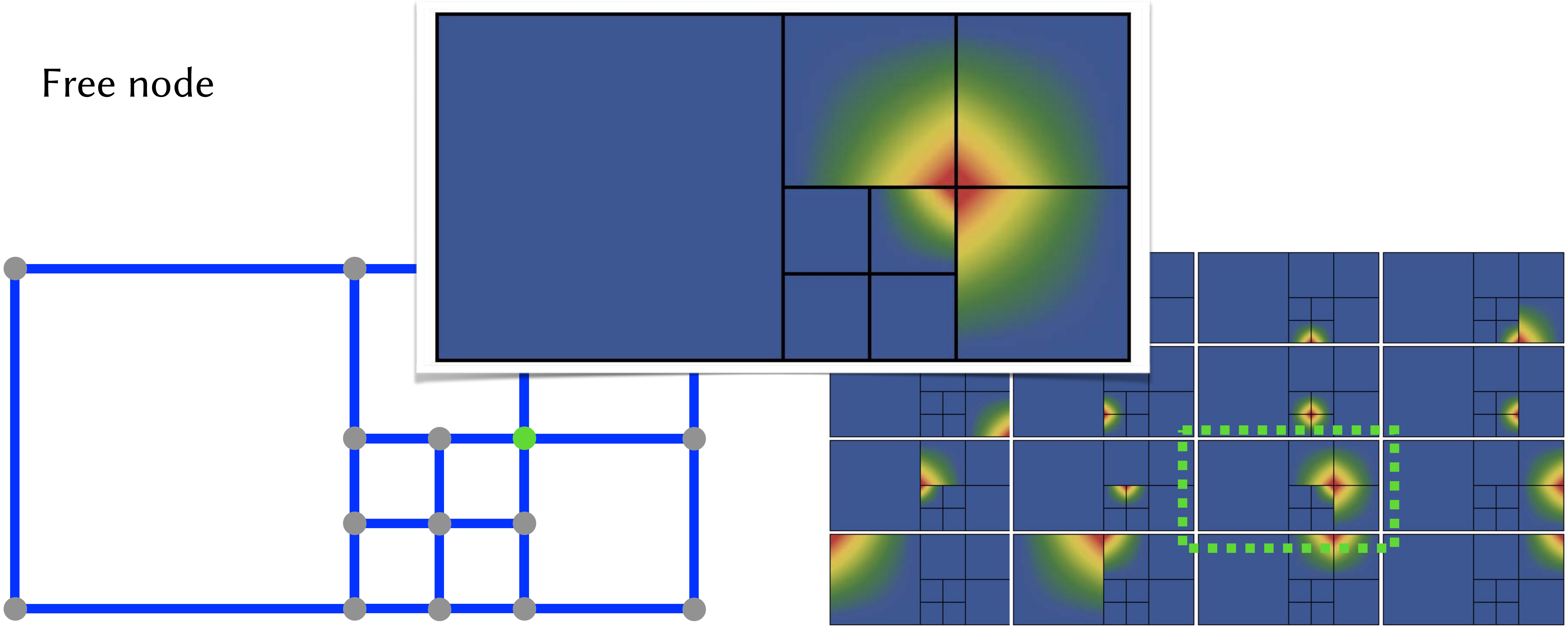
Free node





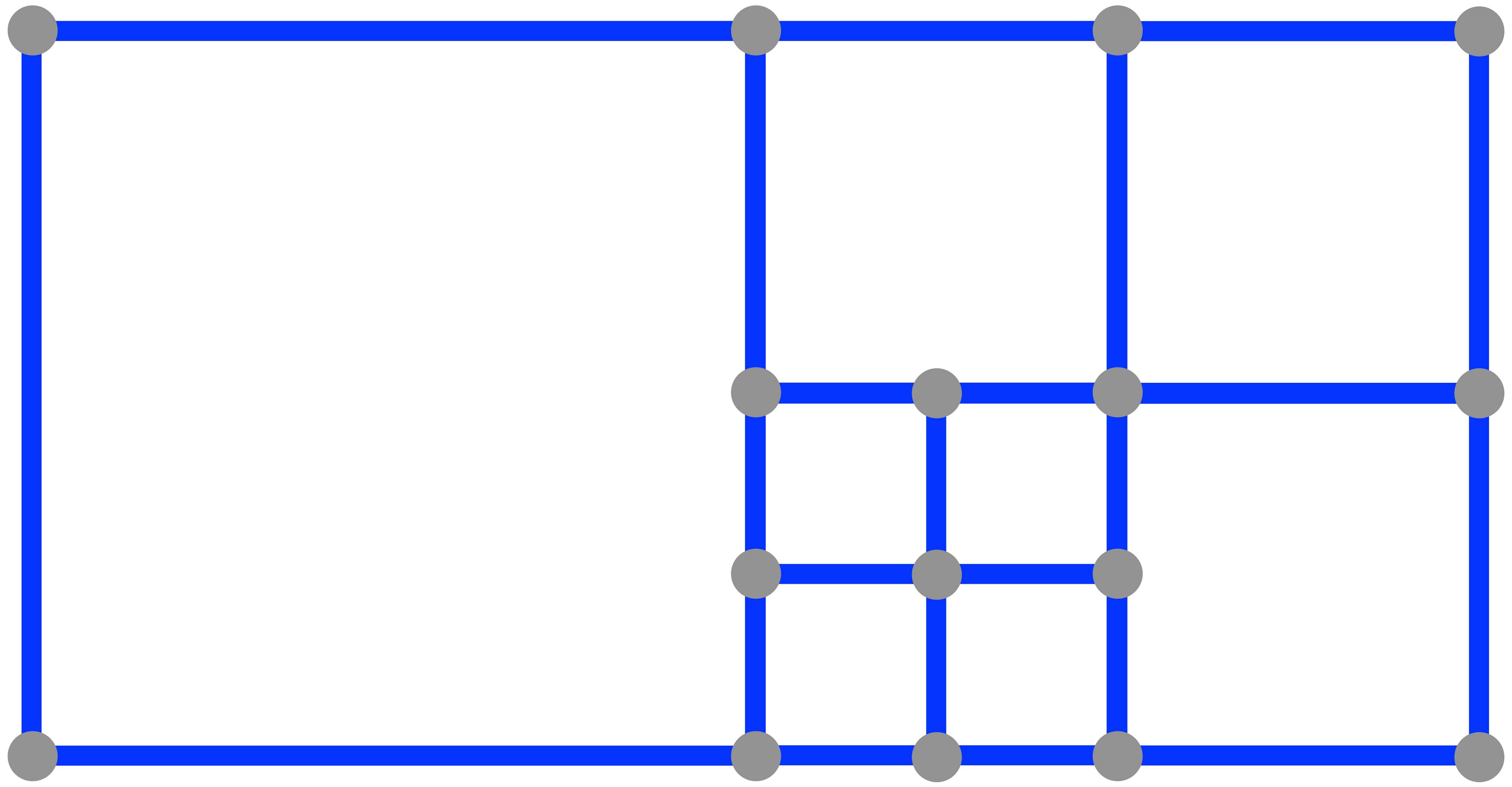
Step1 - set all nodes free

● Free node



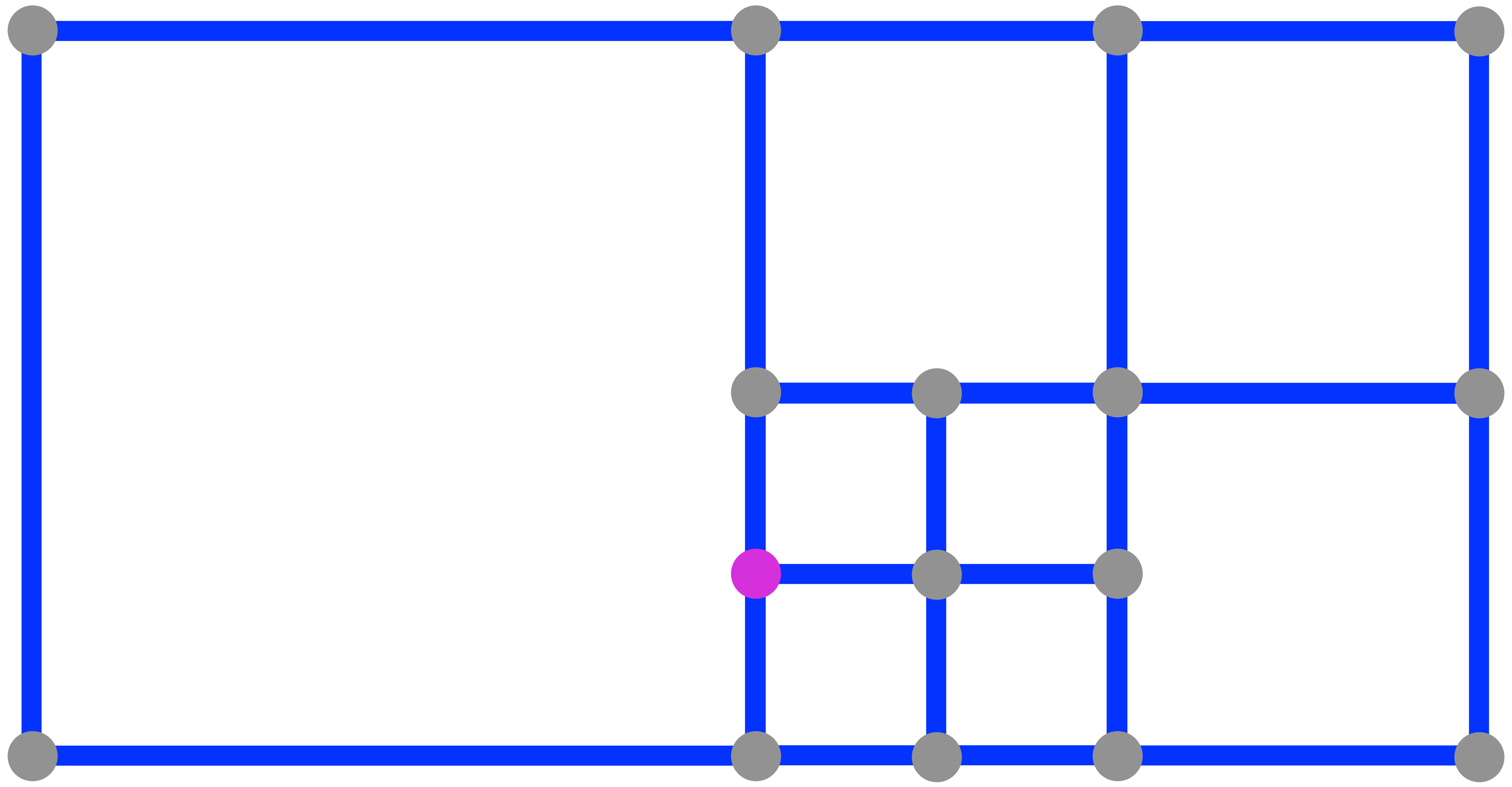
Step 2 - constrain T-junctions

- Free node



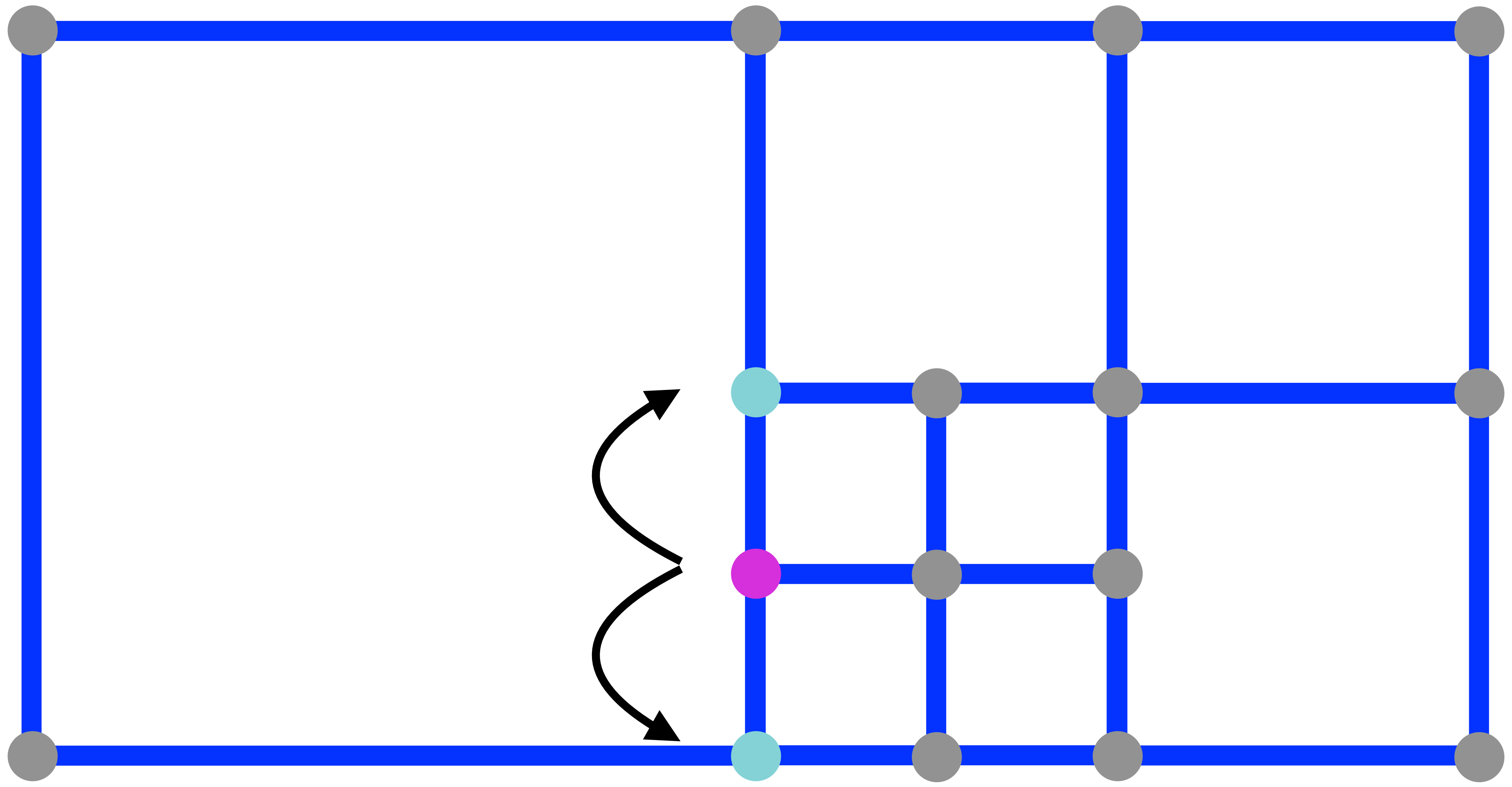
Step 2 - constrain T-junctions

- Free node
- T-junction node



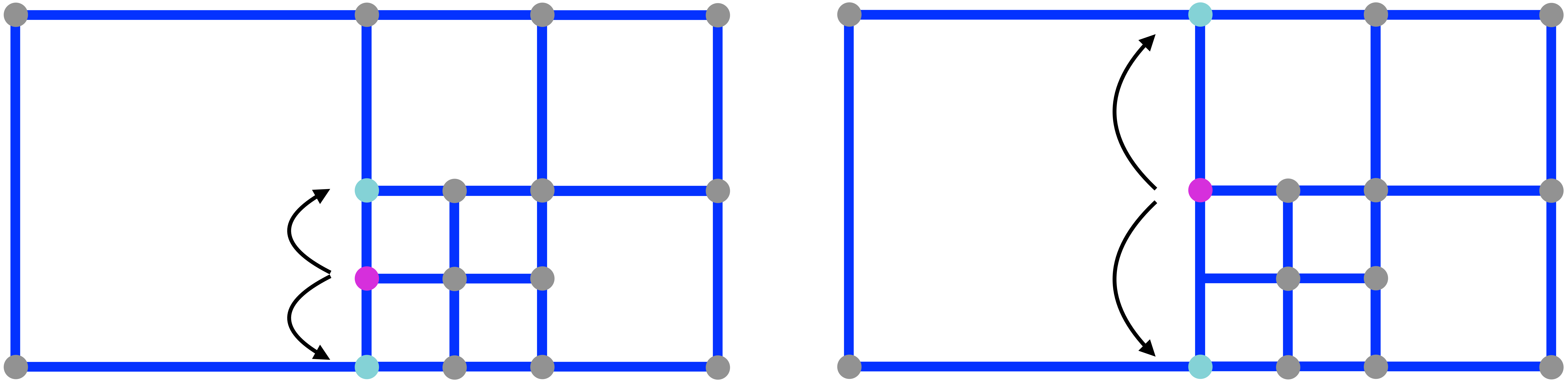
Step 2 - constrain T-junctions

- Free node
- T-junction node
- Parent node



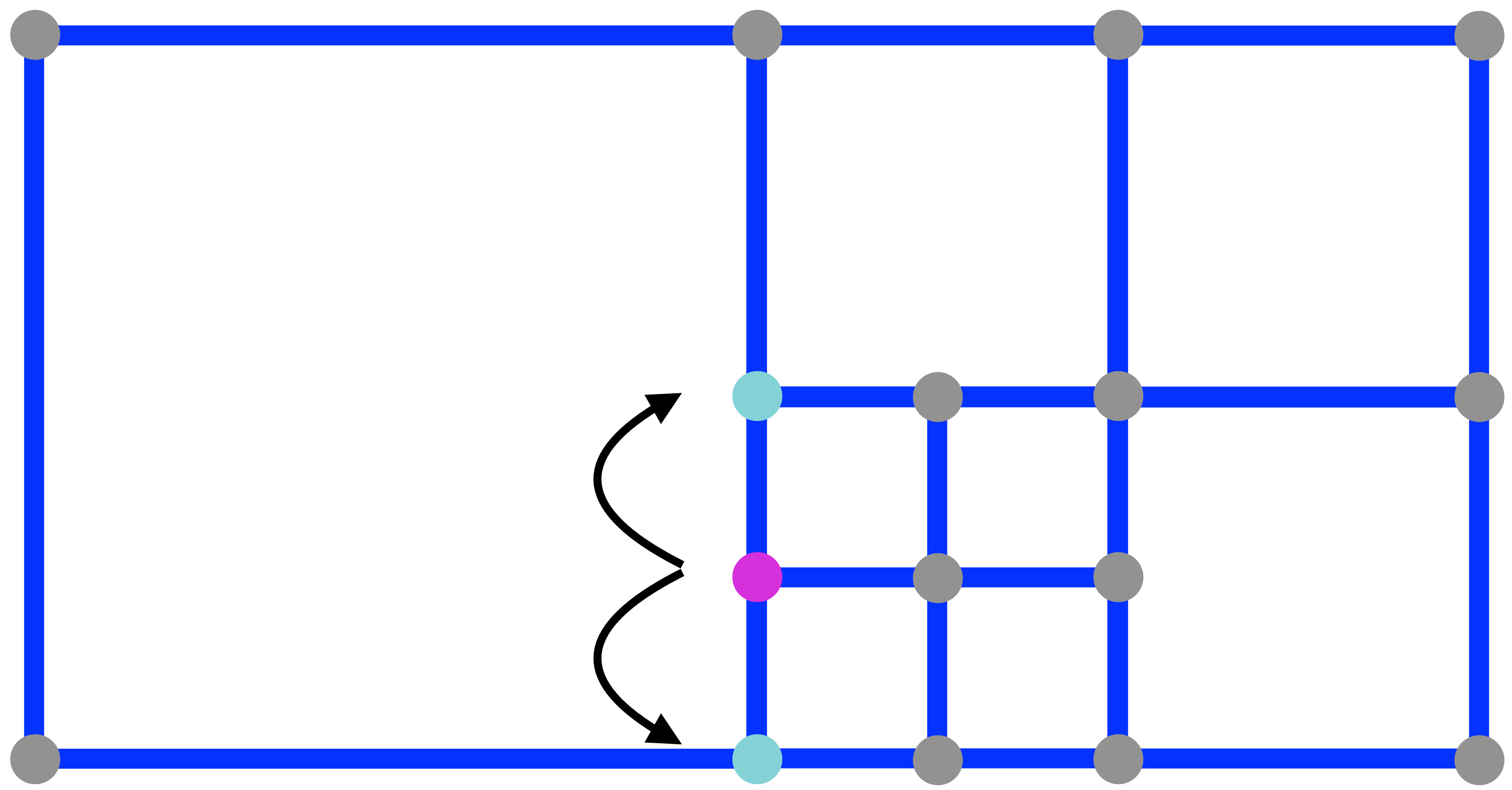
Step 2 - constrain T-junctions

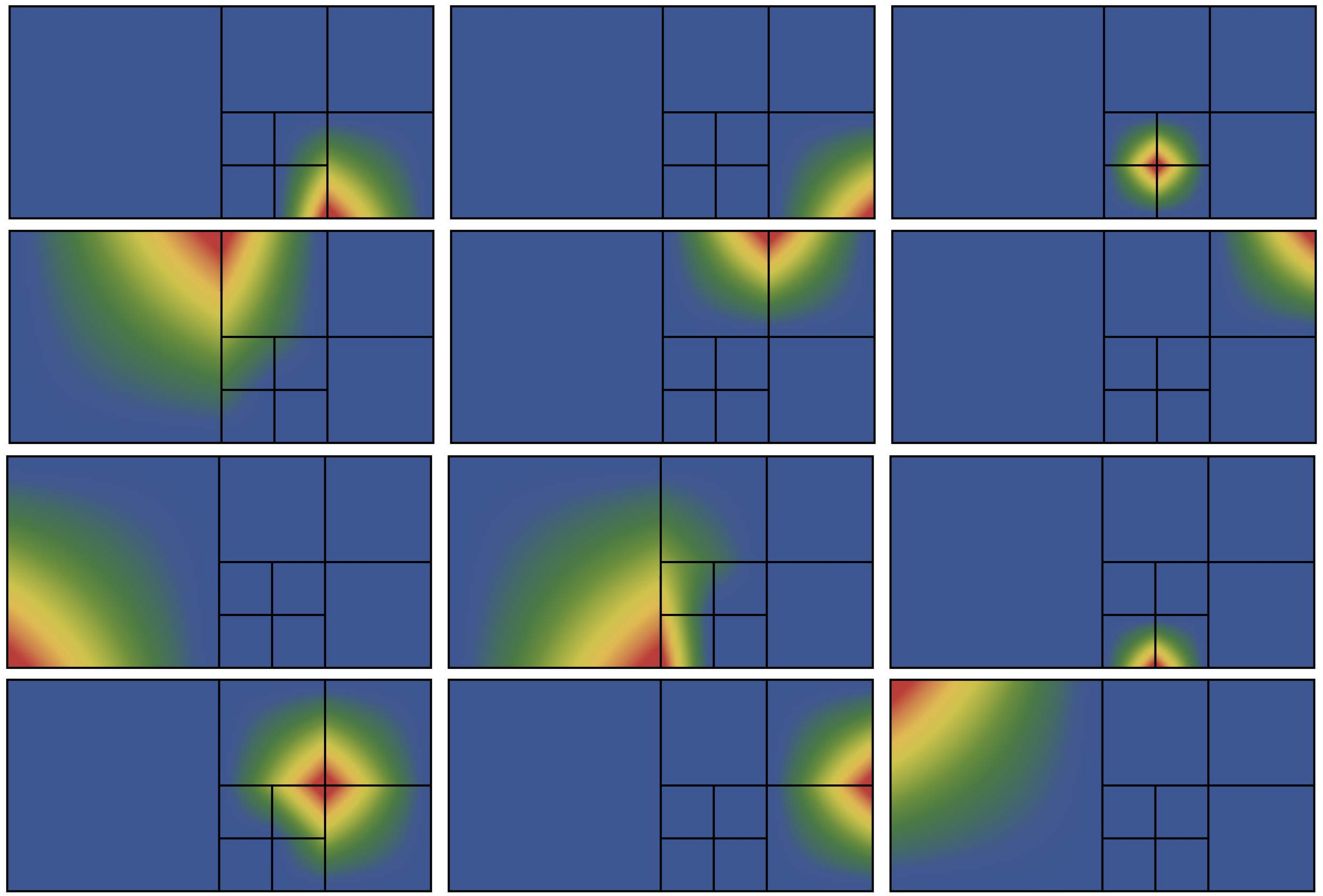
- Free node
- T-junction node
- Parent node



Step 2 - constrain T-junctions

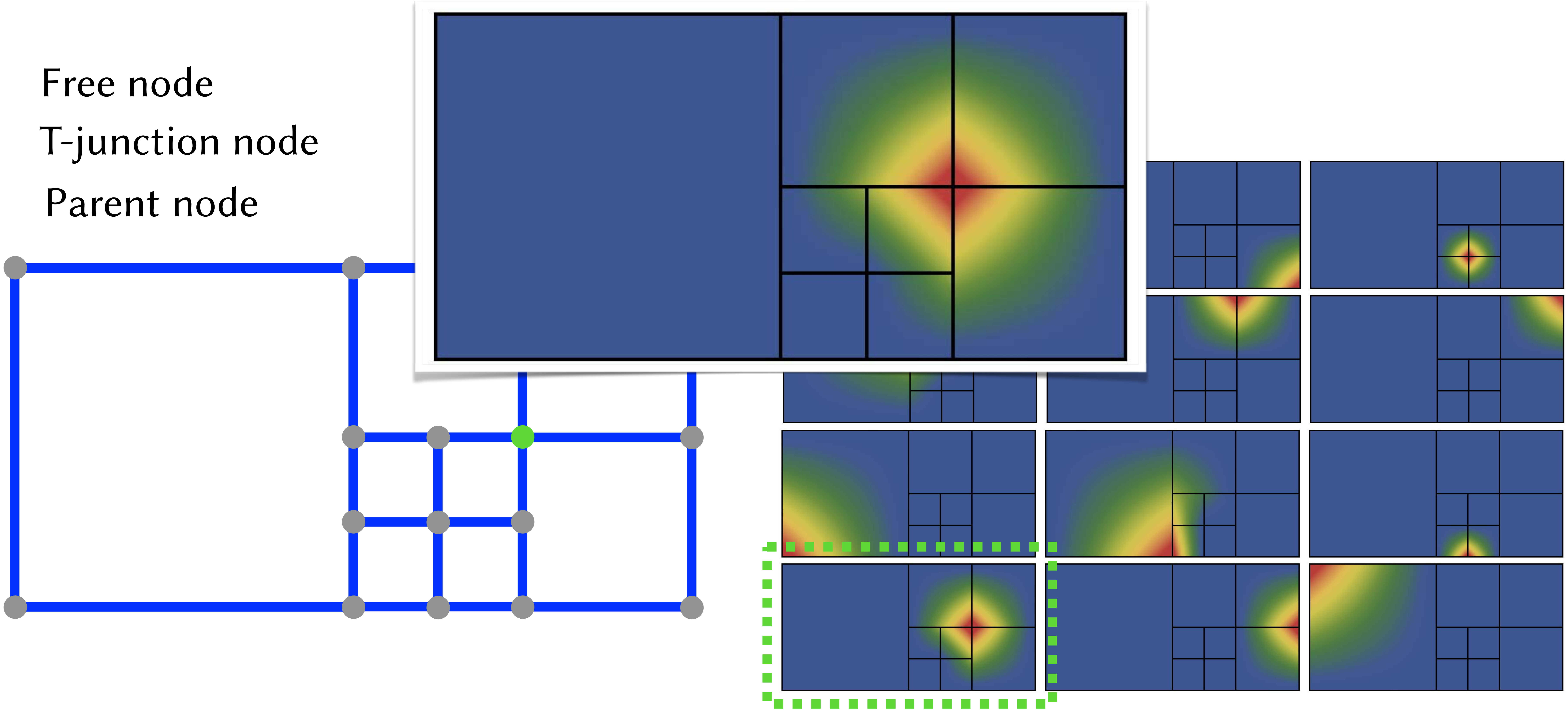
- Free node
- T-junction node
- Parent node



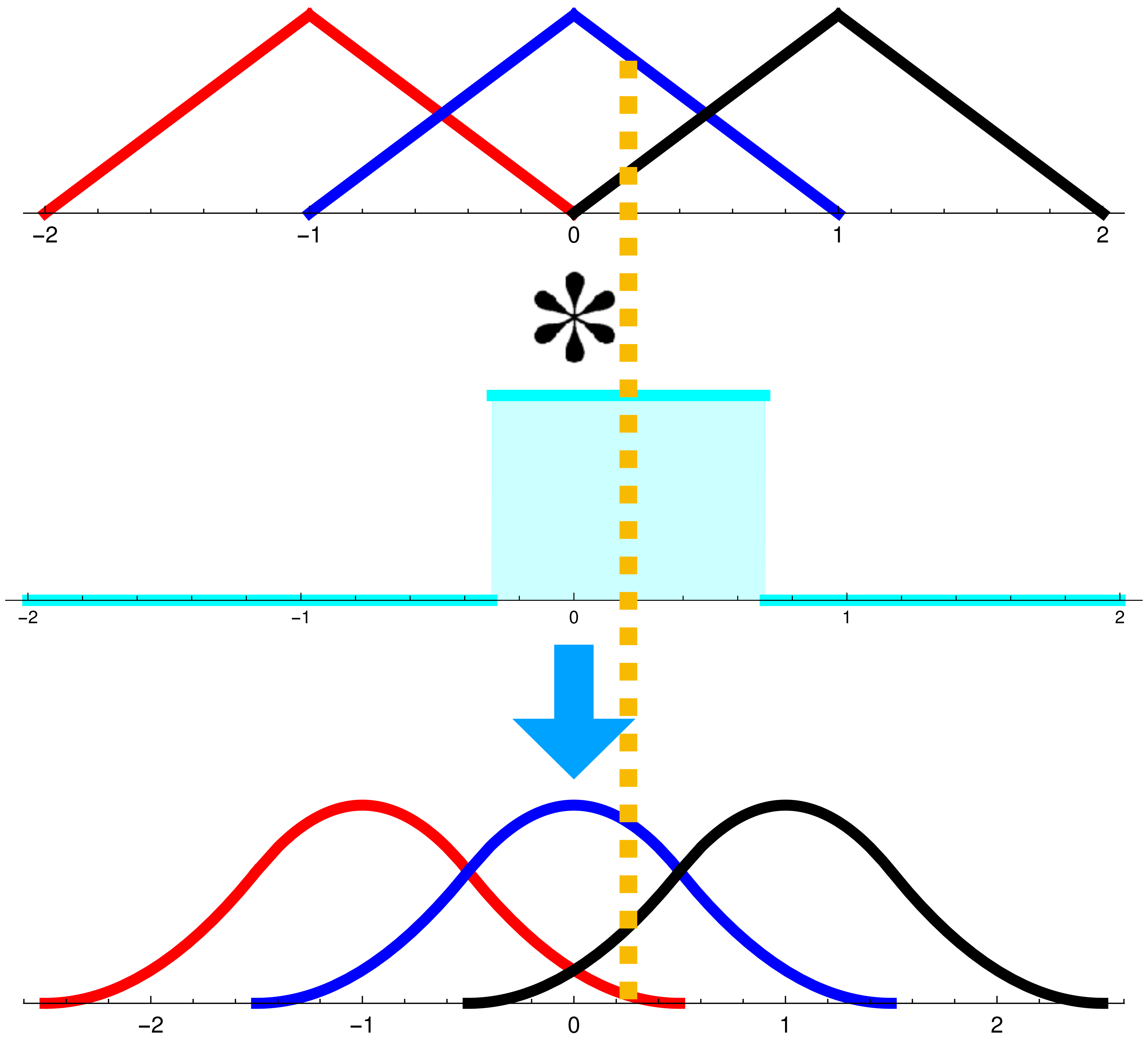


Step 2 - constrain T-junctions

- Free node
- T-junction node
- Parent node



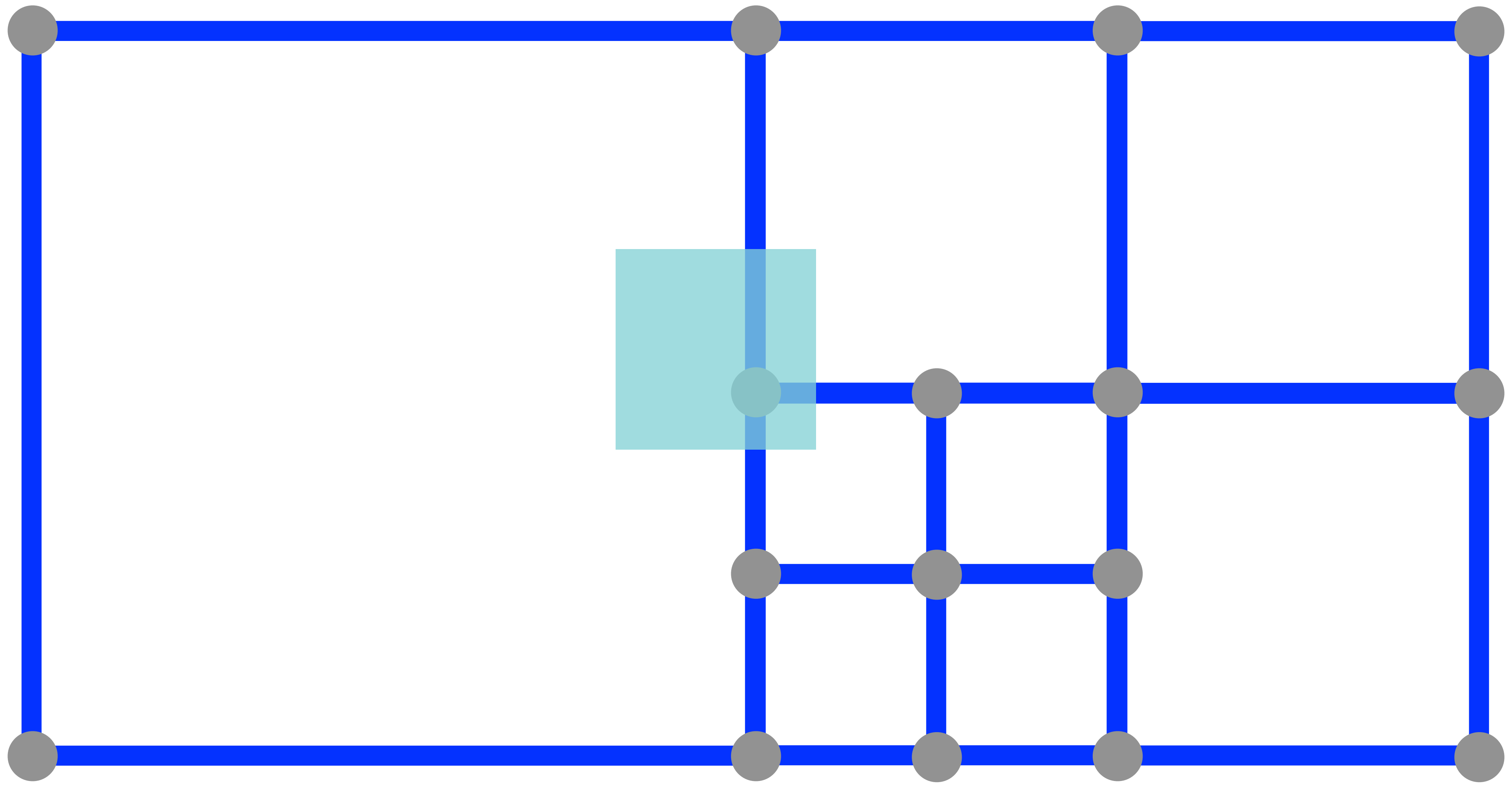
Step 3 - upgrade to C^1 continuity



GIMP

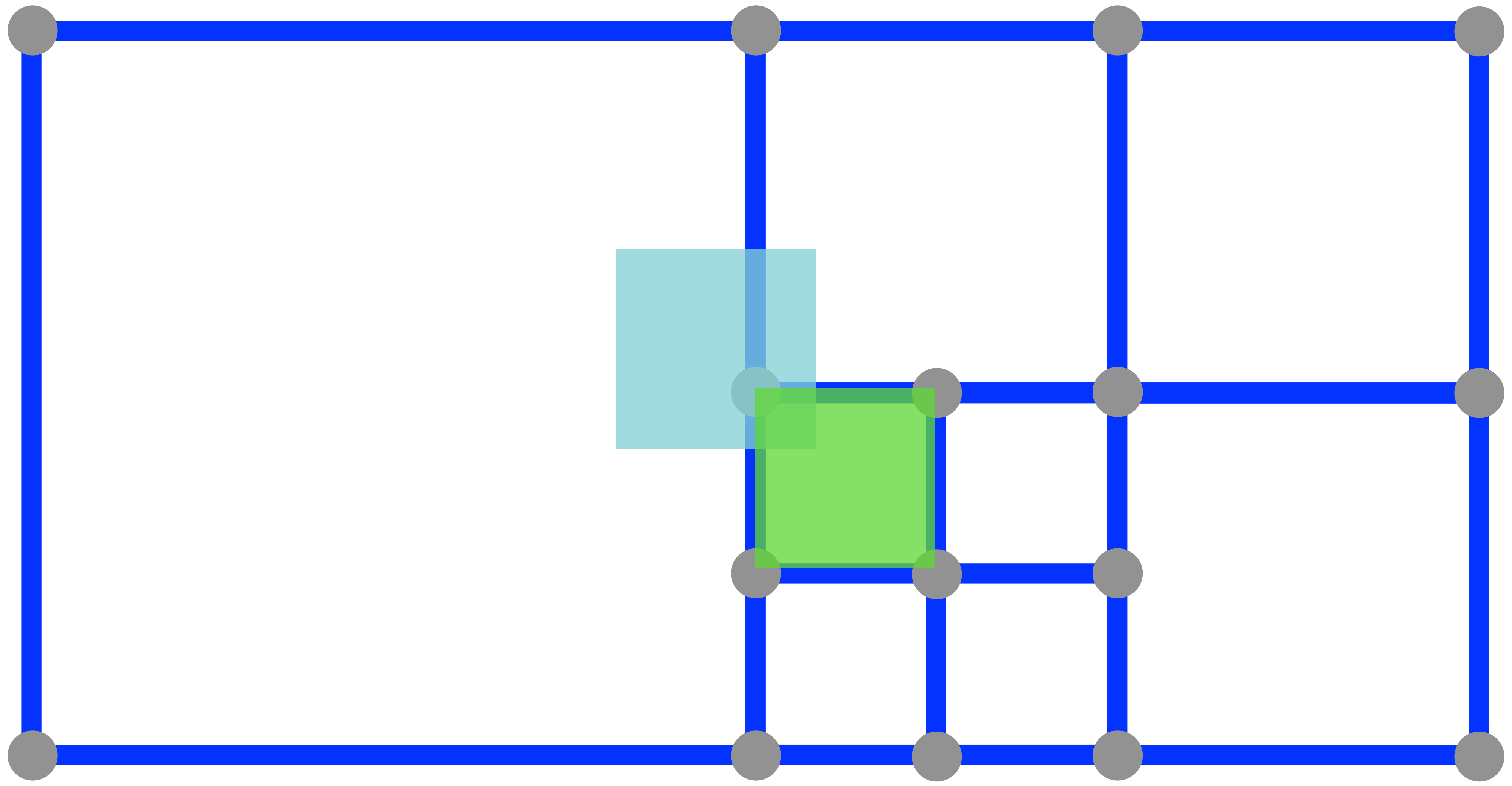
Step 3 - upgrade to C^1 continuity

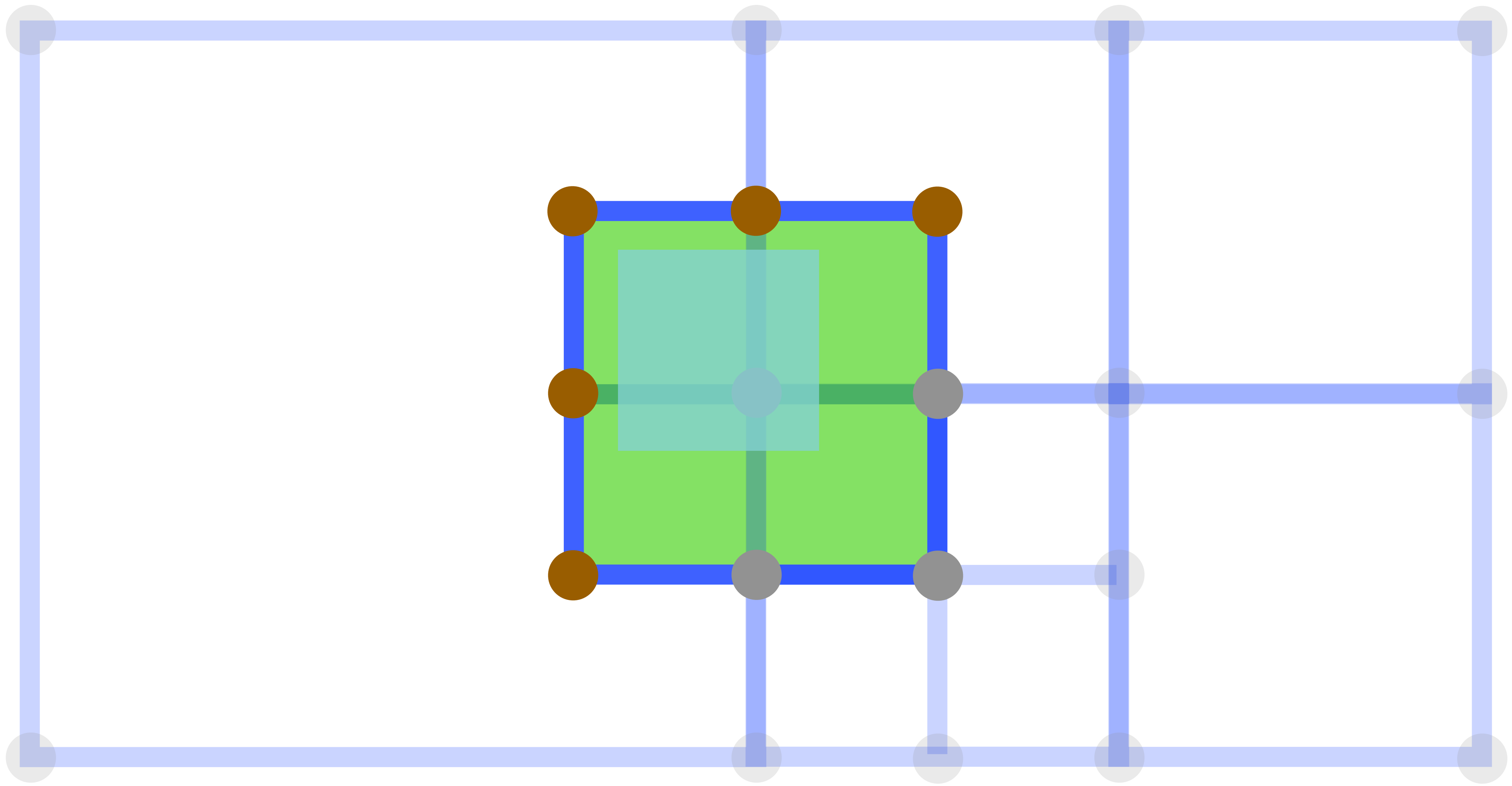
- Free node



Parallelism optimization

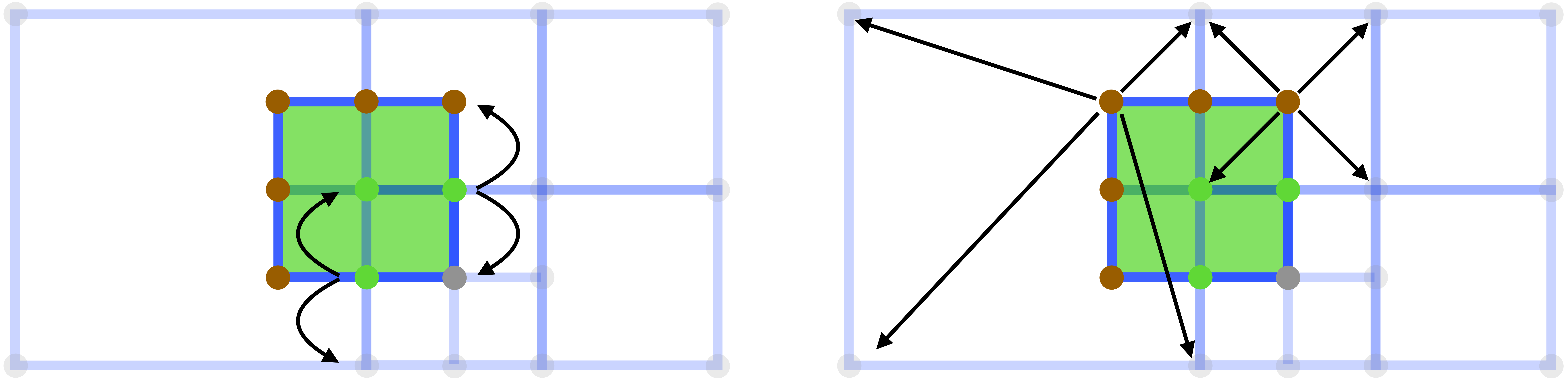
- Free node
- Ghost node



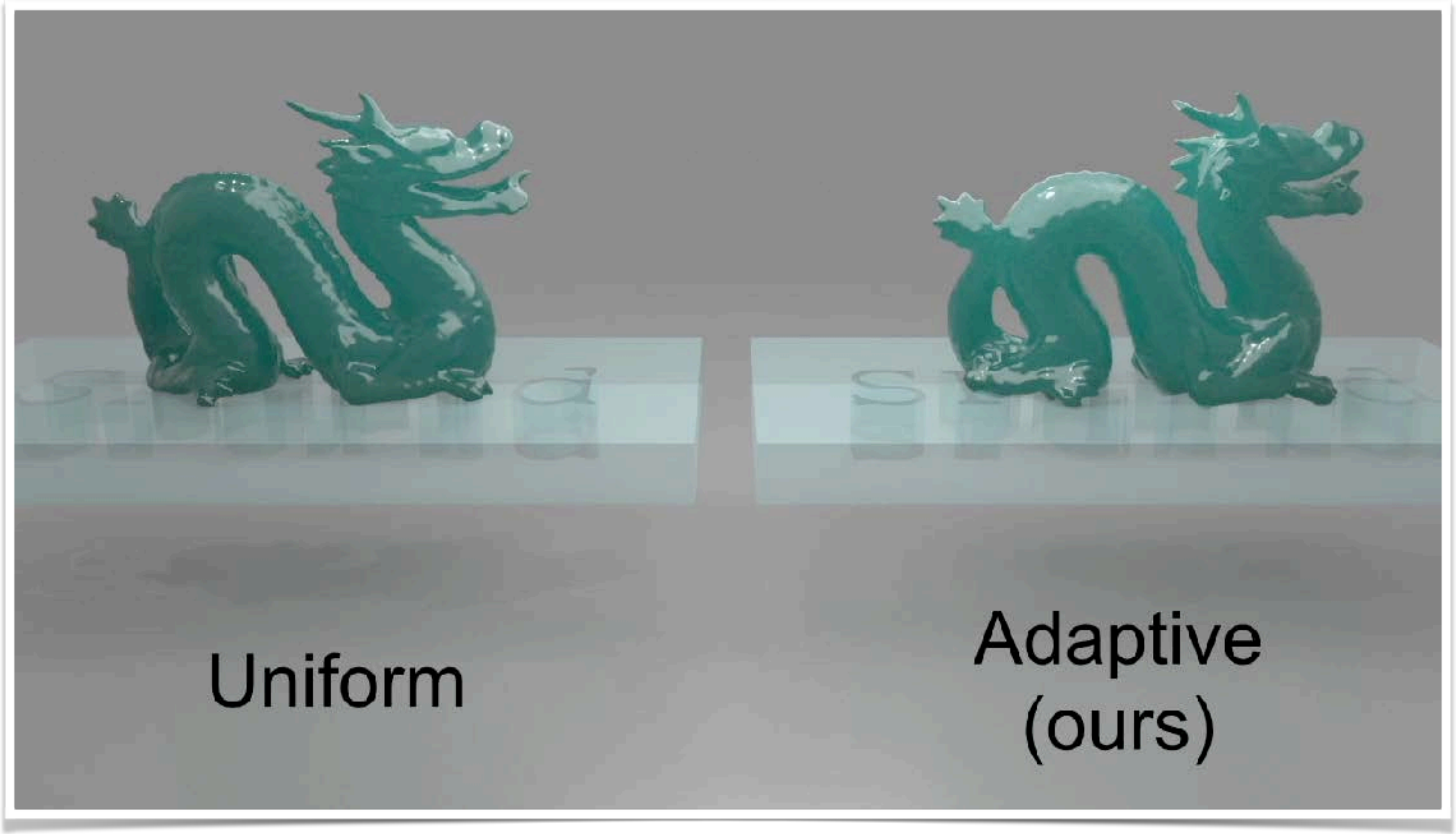


Parallelism optimization

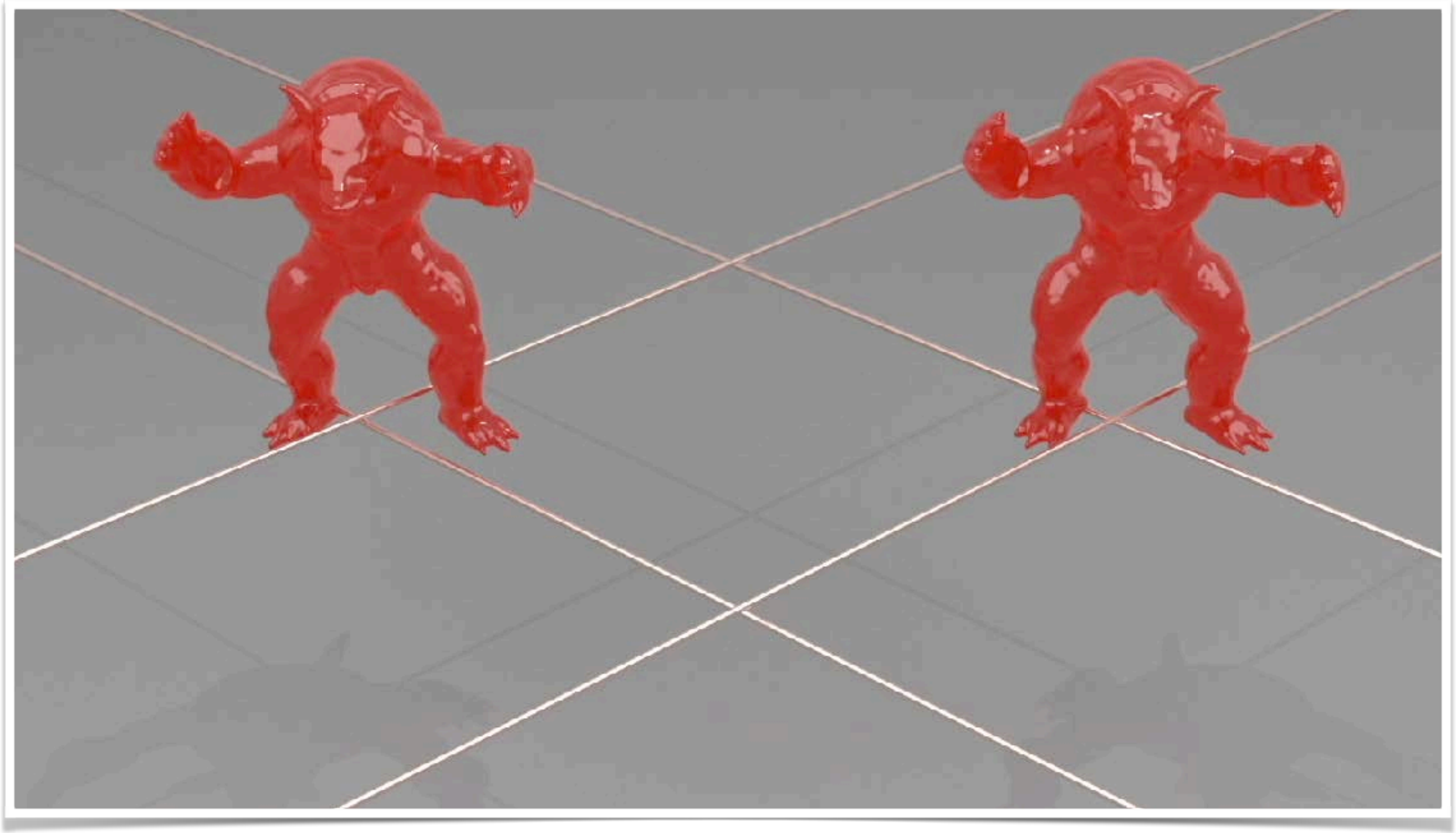
- Free node
- Ghost node
- T-junction node



Results



Results



GPU Optimization of Material Point Method

GPU Optimization of Material Point Methods

Paper ID: 250

ANONYMOUS AUTHOR(S)

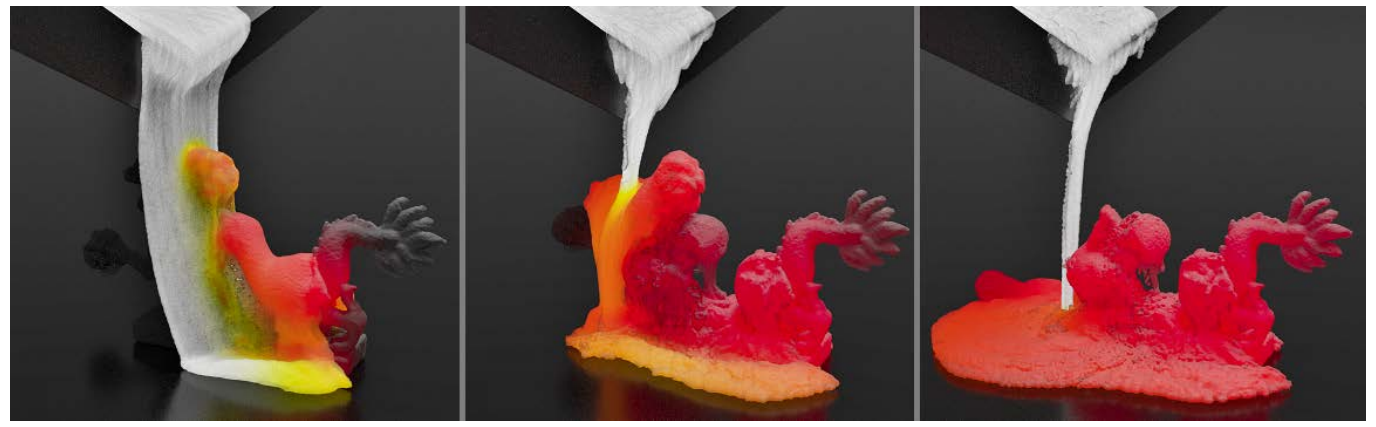


Fig. 1. **How to melt your dragon.** Melting an elastoplastic dragon with 4.2 million particles on a 256^3 grid using our GPU-optimized implicit MPM dynamics and heat solvers on a Nvidia *Quadro* P6000 GPU at an average 10.5 seconds per $48Hz$ frame.

The Material Point Method (MPM) has been shown to facilitate effective simulations of physically complex and topologically challenging materials, with a wealth of emerging applications in computational engineering and visual computing. Borne out of the extreme importance of regularity, MPM is given attractive parallelization opportunities on high-performance modern multiprocessors. Unlike the conceptually simple CPU parallelization, a GPU optimization of MPM that fully leverages computing resources presents challenges that require exploring an extensive design-space for favorable data structures and algorithms. In this paper we introduce methods for addressing the computational challenges of MPM and extending the capabilities of general simulation systems based on MPM, particularly concentrating on GPU optimization. In addition to our open-source high-performance framework, we also perform performance analyses and benchmark experiments to compare against alternative design choices which may superficially appear to be reasonable, but can suffer from suboptimal performance in practice. Our explicit and fully implicit GPU MPM solvers are further equipped with a Moving Least Squares MPM heat solver and a novel sand constitutive model to enable fast simulations of a wide range of materials. We demonstrate that more than an order of magnitude performance improvement can be achieved with our GPU solvers. Practical high-resolution examples with up to ten million particles run in less than one minute per frame.

CCS Concepts: • **Computing methodologies → Physical simulation**;

Additional Key Words and Phrases: Material Point Method (MPM), GPU, SPGrid, GVDB, Hybrid Particle/Grid

DOI: 10.1145/3197517.3201309

ACM Reference format:
Anonymous Author(s). 2018. GPU Optimization of Material Point Methods . *ACM Trans. Graph.* 1, 1, Article 1 (November 2018), 11 pages.
DOI: 10.1145/3197517.3201309

1 INTRODUCTION

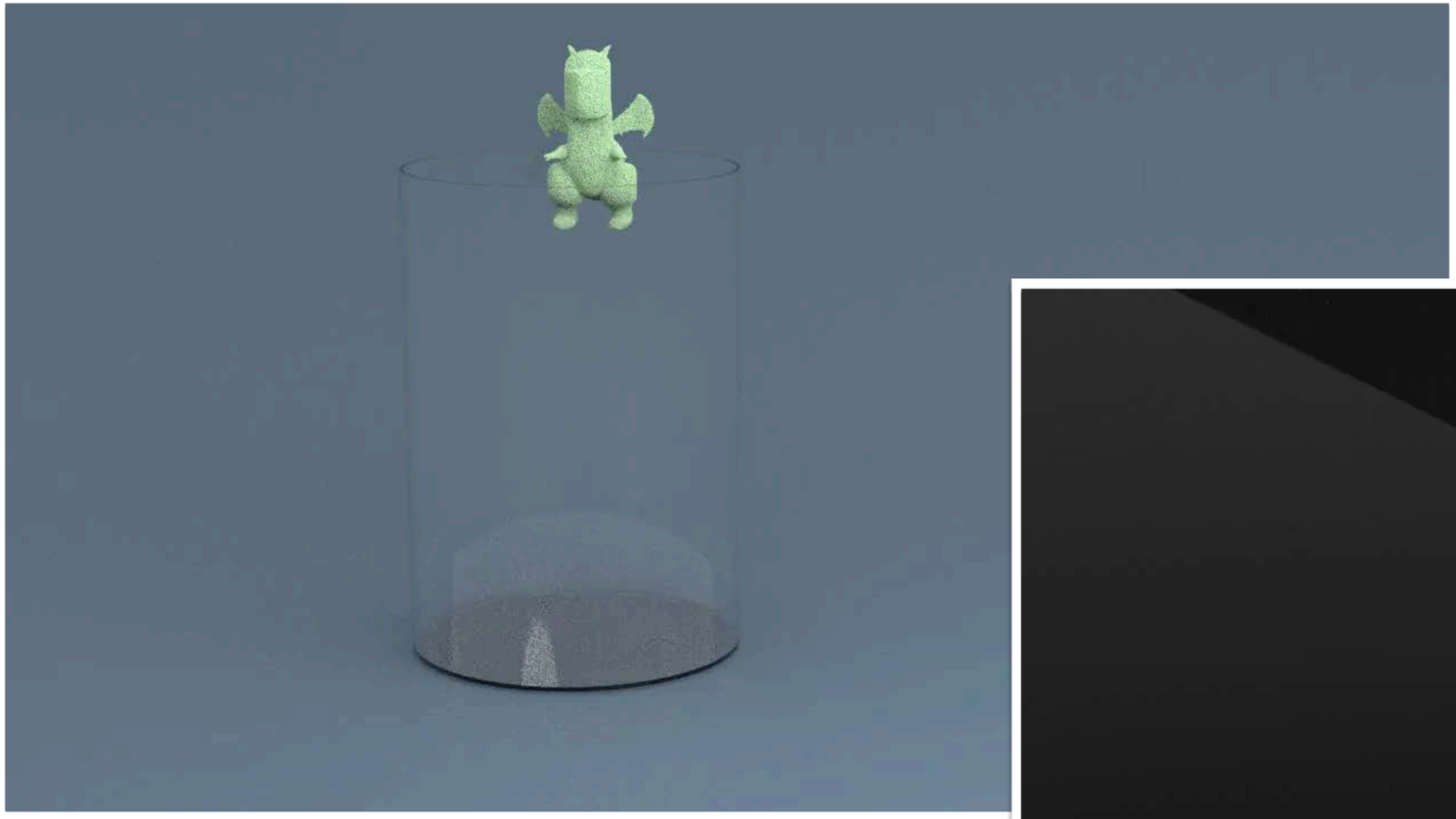
The Material Point Method (MPM) is a hybrid Lagrangian/Eulerian computational scheme that has been shown to simulate a large variety of traditionally-challenging materials with visually rich animations in computer graphics. Recent examples of MPM-based methods developed for such materials include simulations of snow [Stomakhin et al. 2013], granular solids [Klár et al. 2016], multi-phase mixtures [Gao et al. 2018; Stomakhin et al. 2014; Tampubolon et al. 2017], cloth [Jiang et al. 2017a] and many others. MPM has been shown to be particularly effective for simulations involving a large number of particles with complex interactions. However, the size and the complexity of these simulations lead to substantial demands on computational resources, thereby limiting the practical use cases of MPM in computer graphics applications.

Using the parallel computation power of today's GPUs is an attractive direction for addressing computational requirements of simulations with MPM. However, the algorithmic composition of an MPM simulation pipeline can pose challenges in fully leveraging compute resources in a GPU implementation. Indeed, MPM simulations include multiple stages with different computational profiles, and the choice of data structures and algorithms used for handling some stages can have cascading effects on the performance of the remaining computation. Thus, discovering how to achieve a performant GPU implementation of MPM involves a software-level design-space exploration for determining the favorable combinations of data structures and algorithms for handling each stage.

M. Gao*, X. Wang*, K. Wu***(joint first authors)**,
A. Tampubolon, E. Sifakis, C. Yuksel, C. Jiang
SIGGRAPH Asia 2018 (under review)

SIGGRAPH ASIA 2018 TOKYO

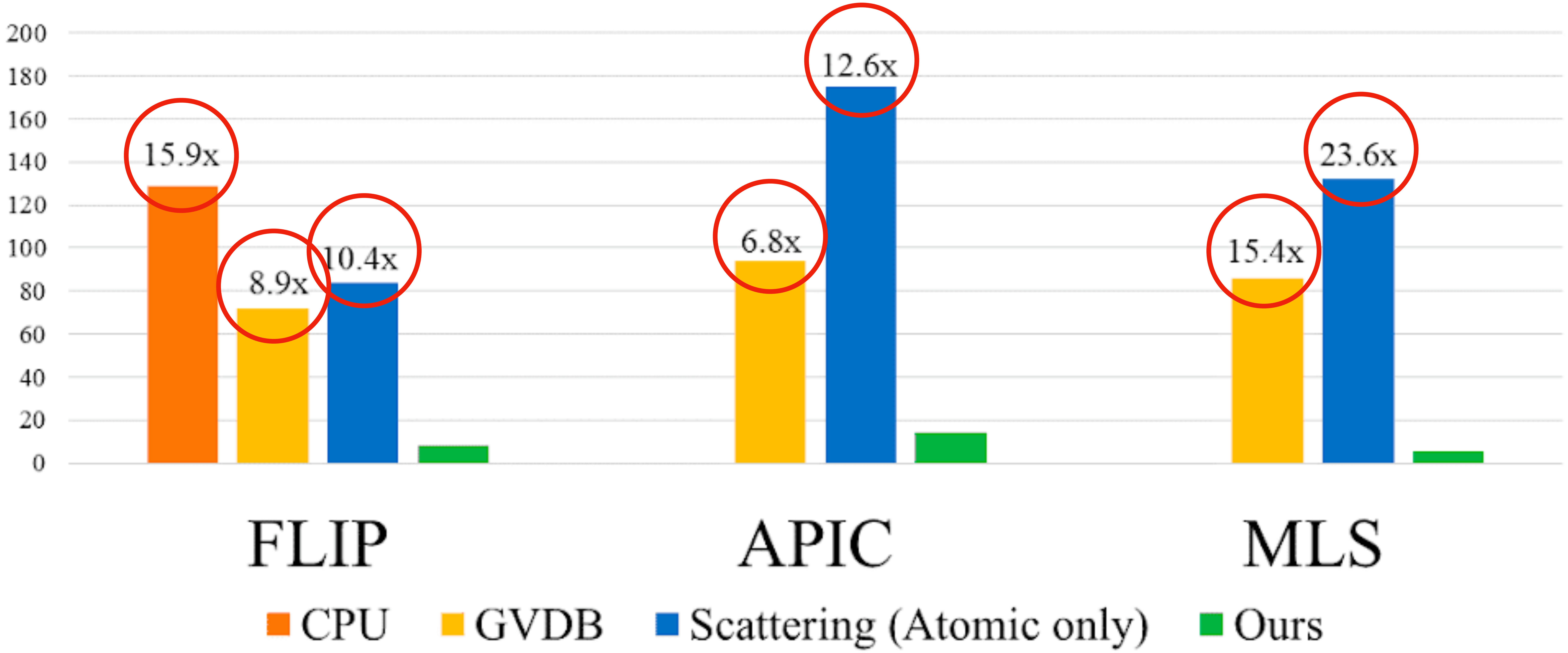
Two demos



Particles: 9.0 M
Grid resolution: 512^3
Simulation: 21.88 secs/frame

Particles: 4.2 M
Grid resolution: 256^3
Simulation: 10.48 secs/frame

Transfer Benchmark



Any question?