

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/312943739>

A physics-motivated Centroidal Voronoi Particle domain decomposition method

Article in Journal of Computational Physics · January 2017

DOI: 10.1016/j.jcp.2017.01.051

CITATIONS

8

READS

469

3 authors:



Lin Fu

Stanford University

78 PUBLICATIONS 302 CITATIONS

[SEE PROFILE](#)



Xiangyu Hu

Technische Universität München

176 PUBLICATIONS 3,563 CITATIONS

[SEE PROFILE](#)



Nikolaus Adams

Technische Universität München

438 PUBLICATIONS 9,457 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Travel supports to all international participates: a conference in Beijing promoting young Aerospace Engineering scientist [View project](#)



Diffusion map for nonlinear systems [View project](#)

Accepted Manuscript

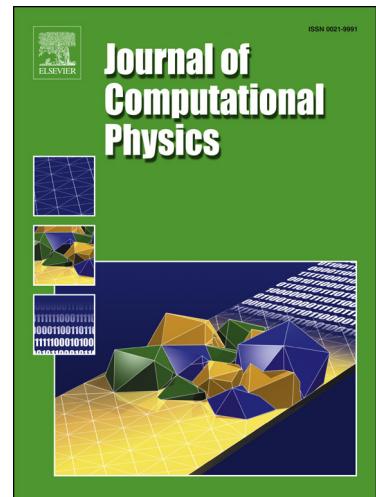
A physics-motivated Centroidal Voronoi Particle domain decomposition method

Lin Fu, Xiangyu Y. Hu, Nikolaus A. Adams

PII: S0021-9991(17)30067-0
DOI: <http://dx.doi.org/10.1016/j.jcp.2017.01.051>
Reference: YJCPH 7113

To appear in: *Journal of Computational Physics*

Received date: 27 April 2016
Revised date: 22 December 2016
Accepted date: 24 January 2017



Please cite this article in press as: L. Fu et al., A physics-motivated Centroidal Voronoi Particle domain decomposition method, *J. Comput. Phys.* (2017), <http://dx.doi.org/10.1016/j.jcp.2017.01.051>

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

A physics-motivated Centroidal Voronoi Particle domain decomposition method

Lin Fu, Xiangyu Y. Hu, Nikolaus A. Adams

Institute of Aerodynamics and Fluid Mechanics, Technische Universität München, 85748 Garching, Germany

Abstract

In this paper, we propose a novel domain decomposition method for large-scale simulations in continuum mechanics by merging the concepts of Centroidal Voronoi Tessellation (CVT) and Voronoi Particle dynamics (VP). The CVT is introduced to achieve a high-level compactness of the partitioning subdomains by the Lloyd algorithm which monotonically decreases the CVT energy. The number of computational elements between neighboring partitioning subdomains, which scales the communication effort for parallel simulations, is optimized implicitly as the generated partitioning subdomains are convex and simply connected with small aspect-ratios. Moreover, Voronoi Particle dynamics employing physical analogy with a tailored equation of state is developed, which relaxes the particle system towards the target partition with good load balance. Since the equilibrium is computed by an iterative approach, the partitioning subdomains exhibit locality and the incremental property. Numerical experiments reveal that the proposed Centroidal Voronoi Particle (CVP) based algorithm produces high-quality partitioning

Email addresses: lin.fu@tum.de (Lin Fu), xiangyu.hu@tum.de (Xiangyu Y. Hu), nikolaus.adams@tum.de (Nikolaus A. Adams)

with high efficiency, independently of computational-element types. Thus it can be used for a wide range of applications in computational science and engineering.

Keywords: Centroidal Voronoi Tessellation, Voronoi Particle, Centroidal Voronoi Particle, Partitioning, High-performance parallel computing

1. Introduction

The need for domain decomposition is encountered in many scientific applications, such as numerical continuum mechanics, parallel computing, data processing, mesh generation, and shape optimization [1][2]. Domain decomposition methods follow the divide and conquer principle, i.e. to solve the global problem by a sequence of local problems. Classical domain decomposition methods can be classified roughly as geometry-based [3] and graph-based approaches [4]. Several open-source codes, such as Metis [5] and PT-Scotch [6], are available to the community. Centroidal Voronoi Tessellation [7] is a typical domain decomposition method and will be the main subject of this paper.

In the following, we denote by "tessellation" a domain decomposition without overlapping subdomains [8]. Given a set of sites $p = \{p_1, \dots, p_n\}$ (Voronoi generators), a Voronoi Tessellation is a subdivision of a domain into n cells, with the property that a point q lies in the cell corresponding to a site p_i if $d(p_i, q) < d(p_j, q)$ for j distinct from i [9]. Centroidal Voronoi Tessellation (CVT) is a specific Voronoi tessellation of a domain characterized by a given density function, for which each Voronoi generator coincides with the mass centroid of the corresponding Voronoi cell. CVT is a partitioning

with an optimized generator distribution, and thus is used in a wide range of applications, e.g. optimal mesh generation, data compression, signal processing and cellular biology [9] [10]. For example, Du et al. [11] recently explored the CVT concept for unstructured isotropic mesh generation by utilizing the dual relation between Delaunay triangulation and Voronoi diagram. Moreover, it has been further extended to anisotropic meshing and geometric surface meshing [12][13].

A CVT can be computed by many methods, such as the deterministic Lloyd's iteration [14] and the probabilistic MacQueen's random algorithm [15]. Lloyd's method is popular for its numerical robustness and simplicity while decreasing the CVT energy function monotonically. A major drawback of Lloyd's method is that it is only linearly convergent, rendering it inefficient for large numbers of generators. Despite the fact that the MacQueen's random algorithm is intuitive to implement, it is not widely used owing to its weak convergence property. For improved efficiency, a parallel MacQueen random algorithm [16] and Lloyd's method in a multi-grid framework [17] have been developed. By recognizing that the CVT energy function has 2nd-order smoothness for convex domains with smooth density, Liu et al. [18] propose a quasi-Newton method to compute CVT and achieve a faster convergence. Nevertheless, to find the global minimum of the CVT energy function is numerically challenging due to its nonlinearity and non-convexity.

For CVT, Gersho's conjecture [19] states that "asymptotically speaking, all cells of the optimal CVT, while forming a tessellation, are congruent to a basic cell which depends on the dimension." Based on Gersho's conjecture, when the generator number tends to infinity and the density function is

smooth enough, the following relation applies to any two Voronoi cell i and j ,

$$\begin{aligned}\rho_i h_i^{d+2} &\approx \rho_j h_j^{d+2}, \\ \rho_i |\Omega_i|^{2/d+1} &\approx \rho_j |\Omega_j|^{2/d+1},\end{aligned}\tag{1}$$

where h , d and $|\Omega|$ denote the local length-scale of a Voronoi cell, spatial dimension and Voronoi cell area ($2D$) or volume ($3D$). In the limit of large generator number, a uniform mass distribution is achieved for $\rho = \rho_t^2$ in two dimensions [20], i.e.

$$\rho_{t,i} |\Omega_i| \approx \rho_{t,j} |\Omega_j|, \quad m_i \approx m_j,\tag{2}$$

where ρ_t denotes the target density function and $m_i = \rho_{t,i} |\Omega_i|$ denotes the mass. The error in satisfying the uniform mass distribution decays linearly with increasing generator number. Although CVT has been successfully used in many applications, it is unsuitable for domain decomposition problems with a strict mass-distribution constraint to be satisfied regardless of the smoothness of the density function and of generator number. In this paper, we first develop a Voronoi Particle method to achieve a target mass distribution, followed by a novel domain decomposition method based on Centroidal Voronoi Tessellation and Voronoi Particle concepts. Specifics of the numerical algorithms, such as boundary conditions, are addressed in detail. In the following, we mainly refer to domain decomposition for parallel computing in continuum mechanics although it should be emphasized that the proposed method has much wider applications. Numerical experiments demonstrate that the new Centroidal Voronoi Particle partitioning method achieves the optimization targets, and is highly robust, efficient and independent of specific mesh-element types, such as adaptive structured meshes,

unstructured meshes and particles.

2. Domain decomposition method

2.1. Centroidal Voronoi Tessellation (CVT)

Given a set of points (generators) $\{\mathbf{x}_i\}_{i=0}^{k-1}$, in an open, simply-connected and convex domain $\Omega \subset \mathbb{R}^d$ without curved boundaries, the Voronoi region Ω_i corresponding to the generator x_i is defined as [10][9]

$$\Omega_i = \{\mathbf{x} \in \Omega \mid \|\mathbf{x} - \mathbf{x}_i\| \leq \|\mathbf{x} - \mathbf{x}_j\|, \forall j \neq i\}, \quad (3)$$

where $\|\cdot\|$ denotes the Euclidean distance in \mathbb{R}^d . For $i \neq j$, $\Omega_i \cap \Omega_j = \emptyset$ and $\cup_{i=0}^{k-1} \Omega_i = \Omega$. Such a set of polyhedra $\{\Omega_i\}_{i=0}^{k-1}$ is called the **Voronoi tessellation (Voronoi diagram)** of Ω .

Considering that we assign a density function $\rho(\mathbf{x}) > 0$ to Ω , the mass centroid of a Voronoi element Ω_i is defined by

$$\mathbf{z}_i = \frac{\int_{\Omega_i} \rho(\mathbf{x}) \mathbf{x} d\sigma}{\int_{\Omega_i} \rho(\mathbf{x}) d\sigma}, \quad (4)$$

where $d\sigma$ denotes the volume differential.

A Voronoi tessellation $\{\Omega_i\}_{i=0}^{k-1}$, for which the generators $\{\mathbf{x}_i\}_{i=0}^{k-1}$ coincide exactly with the mass centroids, i.e.

$$\mathbf{x}_i = \mathbf{z}_i, \quad (5)$$

is called Centroidal Voronoi Tessellation (CVT). With given density field and number of generators, a CVT always exists but may be not unique [21]. However, without the generator-position optimization, the Voronoi Tessellation

of generally-placed generators does not happen to satisfy the requirement of CVT.

From the variational point of view [10], an energy functional can be defined as

$$F(\mathbf{x}) = \sum_{i=0}^{k-1} F_i(\mathbf{x}) = \sum_{i=0}^{k-1} \int_{\Omega_i} \rho(\mathbf{x}) \|\mathbf{x} - \mathbf{x}_i\|^2 d\sigma, \quad (6)$$

where $F_i(\mathbf{x})$ describes the compactness or inertia momentum [18] of Voronoi element Ω_i . The gradient of $F(\mathbf{x})$ is

$$\frac{\partial F}{\partial \mathbf{x}_i} = 2m_i(\mathbf{x}_i - \mathbf{z}_i), \quad (7)$$

with

$$m_i = \int_{\mathbf{x} \in \Omega_i} \rho(\mathbf{x}) d\sigma. \quad (8)$$

For continuous density functions, numerical integration over polyhedral domains is employed [22].

For CVT, we have, for all i ,

$$\mathbf{x}_i = \mathbf{z}_i = \frac{\int_{\Omega_i} \rho(\mathbf{x}) \mathbf{x} d\sigma}{\int_{\Omega_i} \rho(\mathbf{x}) d\sigma}, \quad (9)$$

thus $\frac{\partial F}{\partial \mathbf{x}_i}$ is zero, and the CVT configuration is a critical point of the energy functional. The energy functional Eq. 6 can be minimized by a CVT, but not every CVT configuration satisfying $\frac{\partial F}{\partial \mathbf{x}_i} = 0$ minimizes the energy functional due to the saddle point problem [10].

Since each Voronoi element depends on all generators $\{\mathbf{x}_i\}_{i=0}^{k-1}$, Eq. 9 constitutes a system of nonlinear equations. Through iteratively moving the generators to the mass centroid of Voronoi cells, the CVT can be computed by Lloyd's method, as shown in Algorithm. 1. Du and et al. [23] prove the

convergence of the Lloyd's method and state that Lloyd's method decreases the energy function $F(\mathbf{x})$ without step-size constraint until a local minimum is reached. Due to its superior robustness and simplicity, Lloyd's method is widely used for the discrete K-Means clustering algorithm despite linear convergence [24].

Algorithm 1 Lloyd's method

- 1: Given an initial set of points $\{\mathbf{x}_i\}_{i=0}^{k-1}$ in the computational domain Ω .
 - 2: Construct Voronoi tessellation $\{\Omega_i\}_{i=0}^{k-1}$ corresponding to the generators $\{\mathbf{x}_i\}_{i=0}^{k-1}$.
 - 3: Compute the mass centroids $\{\mathbf{z}_i\}_{i=0}^{k-1}$ of Voronoi elements; move each generator to the corresponding mass centroid, i.e. $\mathbf{x}_i = \mathbf{z}_i$.
 - 4: Check whether the new generators satisfy a convergence criterion, e.g. **the maximum generator displacement is less than a threshold value**; if true, terminate; otherwise, return to step 2.
-

2.2. Voronoi Particle (VP) dynamics

Particle hydrodynamics with Voronoi techniques has been introduced before, see e.g. [25][26]. We adopt this concept for domain decomposition in the following. Hereafter, the generators in the Voronoi diagram are represented by partitioning particles.

We consider the differential form of the inviscid momentum equations for a fluid in Lagrangian form

$$\frac{d\mathbf{v}}{dt} = -\frac{\nabla p}{\rho}, \quad (10)$$

where \mathbf{v} denotes the velocity vector, p denotes the pressure and ρ denotes the density.

Eq. 10 can be discretized based on the Voronoi mesh. For each partitioning particle i , the acceleration due to pressure gradient is computed as

$$\mathbf{a}_i = \frac{d\mathbf{v}_i}{dt} = -\frac{\int_{\Omega_i} \nabla p d\sigma}{\int_{\Omega_i} \rho d\sigma} = -\frac{\int_{\partial\Omega_i} p d\mathbf{S}}{m_i}, \quad (11)$$

where $\partial\Omega_i$ denotes the Voronoi-cell surface.

The partitioning particle pressure is defined as

$$p_i = f\left(\frac{m_i}{m_{tg,i}}\right), \quad (12)$$

where $f(x)$ is chosen as monotonically increasing, and $m_{tg,i}$ denotes a target mass of each Voronoi cell (the definition of $m_{tg,i}$ is application dependent). We set $f(x) = x$, i.e. $p_i = \frac{m_i}{m_{tg,i}}$ which is a suitable choice for our purpose. The target mass $m_{tg,i}$ should satisfy

$$\sum_{i=0}^{k-1} m_{tg,i} = \int_{\Omega} \rho d\sigma, \quad (13)$$

e.g. for equal-sized partitioning, $m_{tg,i} = \frac{\int_{\Omega} \rho d\sigma}{k}$.

The pressure p_{ij} at the Voronoi interface between cell i and j is approximated as

$$p_{ij} = \frac{p_i + p_j}{2}. \quad (14)$$

With the computation of acceleration, the partitioning-particle coordinates \mathbf{x}_i can be evolved as

$$\mathbf{x}_i^{n+1} = \mathbf{x}_i^n + \frac{1}{2}\mathbf{a}_i^n \Delta t^2, \quad (15)$$

where the timestep is constrained globally, i.e.

$$\Delta t = \min\left(0.25\sqrt{\frac{h_i^n}{|\mathbf{a}_i^n|}}\right), \quad (16)$$

for numerical stability. As shown on the left of Fig 1, the length-scale h_i for partitioning particle i is calculated by averaging the distances between neighboring particles from

$$h_i = \frac{1}{2N} \sum_{j=0}^{N-1} h_{ij}, \quad (17)$$

where N denotes the number of neighboring Voronoi cells sharing an edge with cell i .

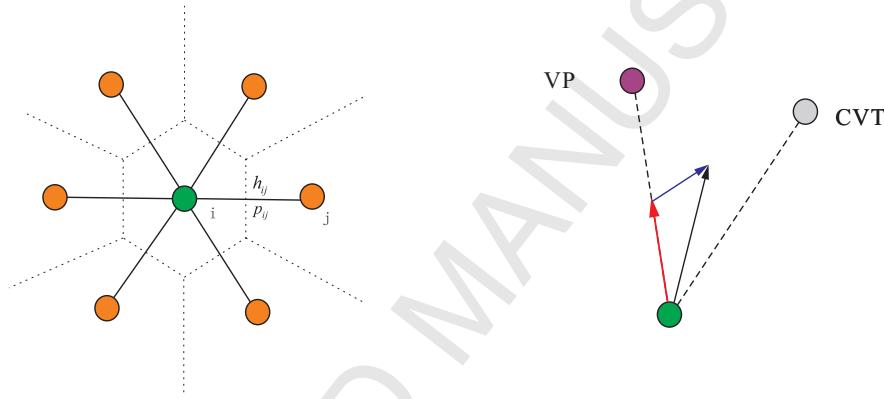


Figure 1: Sketch of Voronoi cell and particle update. For the left panel, the dashed lines denote the Voronoi cell edges, h_{ij} denotes the distance between particle i and j , p_{ij} denotes the pressure on the Voronoi cell edge between particle i and j . The orange circles denote the nearest neighbors of particle i represented by the green circle. For the right panel, the purple circle represents the new particle position by the pure VP method while the grey circle denotes that by pure CVT requirement. The red line with arrow denotes the particle evolution path by the VP part of CVP method while the blue line denotes that by the CVT part. The grey line with arrow denotes the real evolution path.

Provided that the target mass distribution has not been reached, p_i is not a constant, $\frac{\nabla p}{\rho} \neq 0$, and the particle will be accelerated by the pressure force. A stable relaxed configuration is achieved when $m_i = m_{tg,i}$ for all i .

2.3. Centroidal Voronoi Particle (CVP) based domain decomposition method

A typical partitioning problem for parallel large-scale continuum-mechanics simulations imposes several objectives [27]: (a) equal-sized partitioning or an unbalanced partitioning tailored to a specific computing system; (b) minimum communication; (c) connected partitioning and strong locality for data management and communication reduction; (d) incremental property to reduce data migration between processors after repartitioning; (e) efficiency.

In order to allow application of the proposed partitioning method for all computational-element types, the concept of an interaction particle, which is a basic representation of a computational element and is generated at the computational-element center or vertex \mathbf{x} , is introduced. The interaction particle possesses as physical property a mass, which is proportional to the computing cost assigned to the respective computational element. **Assuming that all mesh elements require certain predefined amount of computational cost, the mass is set accordingly everywhere. With the mass distribution of interaction particles, the density function $\rho(\mathbf{x})$ can be obtained immediately.** In practice, a density function does not need to be explicitly defined, and the mass and mass centroid can be computed by

$$\begin{cases} m_i = \int_{\Omega_i} \rho(\mathbf{x}) d\sigma = \sum_{j=0}^{N_i-1} m_{j,i}, \\ \mathbf{z}_i = \frac{\int_{\Omega_i} \rho(\mathbf{x}) \mathbf{x} d\sigma}{\int_{\Omega_i} \rho(\mathbf{x}) d\sigma} = \frac{\sum_{j=0}^{N_i-1} m_{j,i} \mathbf{x}_{j,i}}{m_i}, \end{cases} \quad (18)$$

where N_i represents the number of interaction particles in Voronoi cell i , $m_{j,i}$ and $\mathbf{x}_{j,i}$ denote mass and coordinates of interaction particle j in Voronoi cell i .

Based on the density function $\rho(\mathbf{x})$, we develop the Centroidal Voronoi Particle (CVP) partitioning method by merging the CVT and VP concepts. A set of partitioning particles are initialized in the computational domain and evolved according to Lloyd's method and particle dynamics. While Lloyd's method optimizes the energy function with respect to the global compactness (CVT), the particle dynamics relaxes the partitioning towards the target mass distribution.

With CVP based partitioning method, objectives (a) and (c) are achieved implicitly. Objective (d) is also satisfied as the solution of CVP is computed iteratively. As the number of partitioning particles corresponds to the number of partitioning subdomains and thus typically is very small, the proposed CVP method is highly efficient. Thus objective (e) is also achieved. Although the requirement of minimum communication cannot be proved strictly, previous research [28] implies a direct relation between communication minimization and compactness. The prominent feature of compactness from CVP method results in strictly convex partitioning subdomains with small aspect-ratio, which leads to well communication optimization. The advantage of partition-shape optimization over traditional edge-cut optimization method with respect to communication reduction is demonstrated in [28][29].

Considering that Lloyd's method is not subject to a step-size constraint, unlike the VP method, a straightforward combination leads to poor convergence.

Upon defining a global step-size condition for Lloyd's method as

$$\Delta\tau = \min(\min(\frac{h_i^n}{32}, |\mathbf{z}_i^n - \mathbf{x}_i^n|)/|\mathbf{z}_i^n - \mathbf{x}_i^n|), \quad (19)$$

to satisfy the requirement that one particle moves less than $\frac{1}{32}h$ in

one explicit time integration step, the partitioning-particle location is updated by

$$\mathbf{x}_i^{n+1} = \mathbf{x}_i^n + \Delta\tau(\mathbf{z}_i^n - \mathbf{x}_i^n). \quad (20)$$

In order to improve convergence, an explicit two-step strategy for particle evolution is proposed. As shown on the right of Fig 1, the partitioning particle first is updated by the VP method as

$$\mathbf{x}_i^* = \mathbf{x}_i^n + \alpha \frac{1}{2} \mathbf{a}_i^n \Delta t^2, \quad (21)$$

i.e. the red path. In a second step, the CVT shift is imposed as

$$\mathbf{x}_i^{n+1} = \mathbf{x}_i^* + (1 - \alpha) \Delta\tau(\mathbf{z}_i^n - \mathbf{x}_i^*), \quad (22)$$

i.e. the blue path. α is a relaxation parameter, which affects convergence. We find that $\alpha > 0.5$ ensures that essentially VP dynamics dominates over CVT in relaxing the system. As VP dynamics is analogous to a physical pressure relaxation process, a stable equilibrium state of the system can be obtained. The relaxation process of VP dynamics ensures convergence and the CVT method optimizes compactness simultaneously. Extensive numerical experimentation shows that $\alpha = 0.8$ gives the best performance.

3. Numerical algorithms

3.1. Initial and boundary condition

The initial particle distribution should be chosen such that none of the particle has zero mass, i.e. $m_i > 0$ for all i , within the associated Voronoi tessellation. Concerning efficiency, sampling the initial partitioning particles by the Monte Carlo method according to the given density function, e.g.

rejection sampling, can accelerate the convergence for large number of particles [16]. For demonstrating the robustness and convergence of CVP, we randomly sample particles in the computational domain.

The ghost particle method is employed to enforce the symmetry boundary condition, **where the particle locations are pairwise symmetric about the domain boundary, scalar variables are the same and vector velocities are mirroring**. The boundary condition is enforced in two steps. In the first step, the domain-interior particles for mirroring, i.e. boundary particles, are identified. As shown on the left of Fig. 2, particles in all Voronoi cells that contain Voronoi vertices outside the computational domain are marked as boundary particles. **By mirroring these particles, a set of ghost particles is constructed, and all physical properties, such as mass and pressure, are copied from the corresponding particles.** In the second step, a new Voronoi diagram is constructed including these ghost particles, as shown on the right of Fig. 2. The Voronoi cells cutting the boundary are now closed by the domain-boundary segments. The ghost particles are reconstructed for each iteration. Consequently, all numerical algorithms discussed in the last section now can be applied to all particles within the computational domain.

3.2. Convergence criteria

In order to check convergence, we define the maximum partitioning error as

$$E_{max} = \max(E_0, \dots, E_{k-1}), \quad (23)$$

where $E_i = \frac{|m_i - m_{tg,i}|}{m_{tg,i}}$, $i = 0, \dots, k-1$.

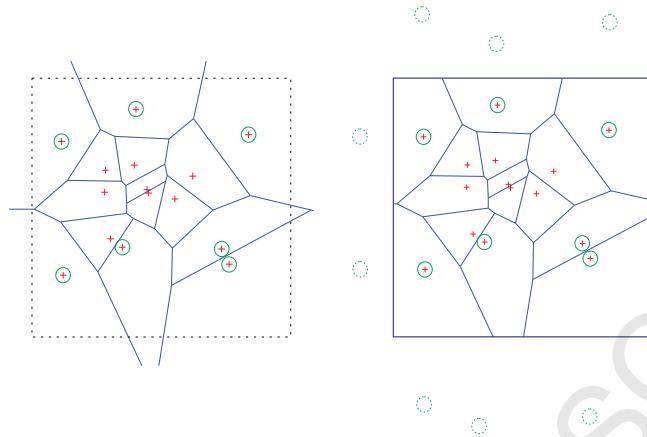


Figure 2: Sketch of boundary condition. The general Voronoi diagram (left panel) and the Voronoi diagram with ghost particles (right panel). The blue lines represent the Voronoi cell edges while the dashed black lines represent the computational domain boundaries. The red crosses denote partitioning particles, the green circles denote boundary particles while the dashed circles denote the ghost particles.

The convergence criterion is defined as follows : within a limited number of iterations, e.g. 100, both the averaged E_{max} and the E_{max} of the current iteration are smaller than a threshold, e.g. 5%.

The overall numerical algorithms for the CVP-based partitioning method are given as Algorithm. 2.

Algorithm 2 CVP-based domain decomposition method for general given density function $\rho(\mathbf{x})$

- 1: Randomly initialize the partitioning particle distribution in the computational domain.
 - 2: Initialize the density function $\rho(\mathbf{x})$ in the computational domain.
 - 3: **while** convergence criteria are not satisfied **do**
 - 4: Construct the Voronoi diagram, compute the partitioning particle mass (Eq. 8) and mass centroid (Eq. 4), define the partitioning particle pressure (Eq. 12).
 - 5: Construct the ghost particles enforcing the symmetric boundary condition (Section. 3.1).
 - 6: Compute the partitioning particle scale (Eq. 17), force and acceleration (Eq. 14, Eq. 11).
 - 7: Calculate the timestep (Eq. 16) and update the partitioning particles according to VP method (Eq. 21).
 - 8: Calculate the step-size condition (Eq. 19) and update the partitioning particles according to CVT method (Eq. 22).
 - 9: Check convergence (Section. 3.2); if true, terminate.
 - 10: **end while**
-

4. Numerical validations

Since the application research of proposed CVP method concentrates on the parallel computing and considering the fact that modern parallel computing system utilizes the hybrid parallel framework (multi-thread parallelization inside one CPU node and MPI parallelization between CPU nodes),

the number of the partitioning subdomains typically is moderately large. E.g. for the HPC system of Leibniz-Rechenzentrum, München (LRZ), 10000 CPU cores are available at 250 CPU nodes and 250 partitioning particles are needed correspondingly.

In order to measure and compare the computational efficiency, all simulations are carried out on the same desktop workstation equipped with 12 Intel(R) Xeon(R) CPU E5-2620 V2 cores (2.1 GHz and 32 G memory) and Scientific Linux 6.7 system.

4.1. Uniform mesh partitioning

In this case, a uniform mesh of resolution 3200×3200 in a computational domain $[0, 1] \times [0, 1]$ is partitioned into 400 subdomains. Thus, 400 partitioning particles are involved. In order to demonstrate the convergence, the iteration number for each algorithm is fixed as 4000.

Fig. 3 gives the convergence histories of maximum load-imbalance error E_{max} , energy function and interfacial area. All three methods roughly converge within a few hundred iterations. For uniform meshes, the interfacial area represents the communication between distinct partitioning subdomains. It is observed that the CVT method produces large load-imbalance errors although the energy function and the interfacial area are minimized. While the load-imbalance error is minimized by the VP method, it fails to optimize the energy function and the interfacial area. Indeed, the total interfacial area increases slightly indicating larger communication overhead. The proposed CVP method optimizes all three objective functions simultaneously. Moreover, the total interfacial area asymptotically converges to the minimum value given by CVT.

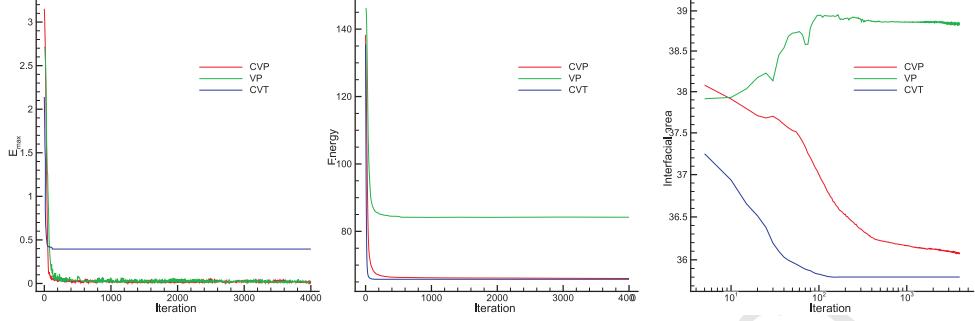


Figure 3: The convergence histories of maximum load-imbalance error E_{max} , energy function and interfacial area for three different partitioning methods.

From Fig. 4, we can see that most of partitioning subdomains from CVP and CVT methods converge to regular hexagons while that from the VP method are quite irregular. This can also be inferred from Gersho's conjecture, which states that "asymptotically speaking, all cells of the optimal CVT, while forming a tessellation, are congruent to a basic cell which depends on the dimension.". For two dimensions, the basic cell for the optimal CVT is a regular hexagon while for three dimensions, although not theoretically verified, body-centered-cubic (BCC) lattice based CVT, i.e. the truncated octahedron, is numerically demonstrated to be the congruent cell [19]. H. Meyerhenke and et al. [30] point out that such convex subdomain shapes, e.g. hexagon, are the regular shapes which cover a two-dimensional domain with smallest interface areas and thus the total interfacial energy is minimum. This is also confirmed by the convergence history of interfacial area in Fig. 3. The current partitioning result is highly similar to that from representative diffusion BUBBLE-FOS/C graph (Re)partitioning heuristic (see Fig.1 and Fig.7 in [30]) and is better than the solutions delivered by well-established softwares, e.g. Metis [5].

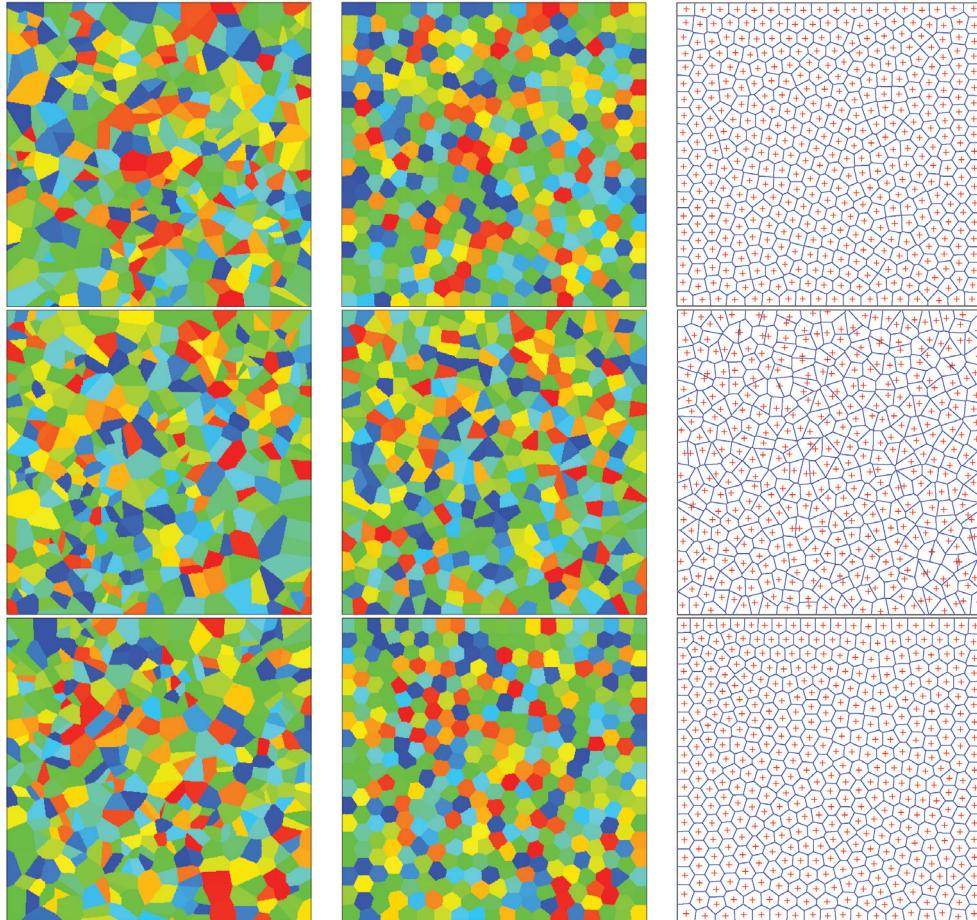


Figure 4: Sketch of the partitionings and Voronoi diagrams. From left to right: initial partitioning, converged partitioning and Voronoi diagram. From top to bottom: results from CVP, VP and CVT methods. The cross symbols in the right panel represent the partitioning particles.

Since in practice the computational capabilities of distinct processor cores may differ significantly, the target load-imbalance should be tailored to fit the computational capability of specific processor core. Assuming that 200 out of the 400 processor cores have a 50% higher computational capability than the others, an unbalanced partitioning simulation is performed. As shown in Fig. 5, three objective functions of E_{max} , energy function and interfacial area are well optimized.

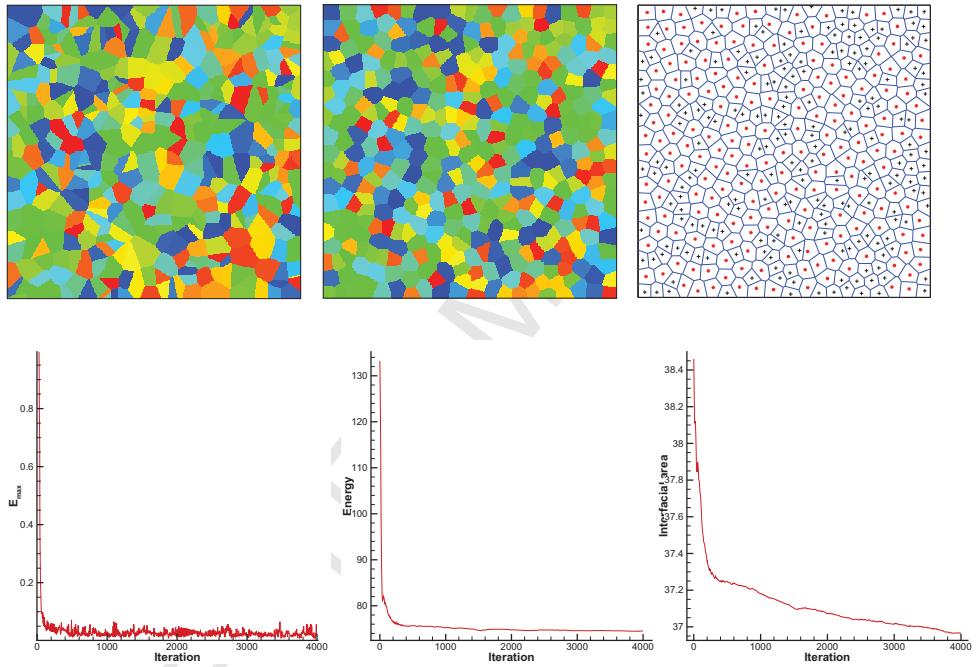


Figure 5: Sketch of the partitioning results and convergence history. Top: initial partitioning, convergent partitioning and convergent Voronoi diagram. The black cross symbols denote partitioning particles of small target mass while the red star symbols denote those of large target mass. Bottom: convergence history of maximum load-imbalance error E_{max} , energy function and interfacial area.

Fig. 6 shows the partitioning topologies for uniform meshes with small partitioning numbers.

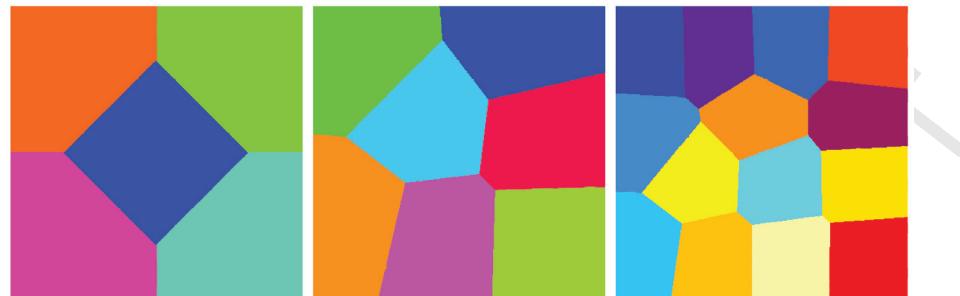


Figure 6: Sketch of the partitioning results for uniform mesh with few generators. The partitioning number is 5, 7, and 14.

4.2. Partitioning of adaptive particle distribution

In this case, an adaptive particle distribution consisting of 6736 particles, as shown in Fig. 7, is employed as the target partitioning mesh. The partitioning number ranges from 7 to 35. Only one processor core is adopted.

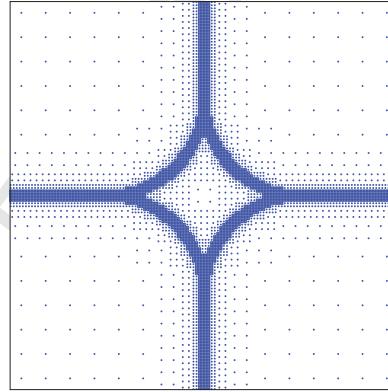


Figure 7: Sketch of adaptive particle distribution.

Fig. 8 gives the convergence histories. It can be noticed that all five partitioning simulations converge rapidly in less than 500 iterations while load-imbalance error and energy function are well optimized. Furthermore,

less than 2 seconds are consumed even for partitioning number 35, indicating high efficiency of the CVP method.

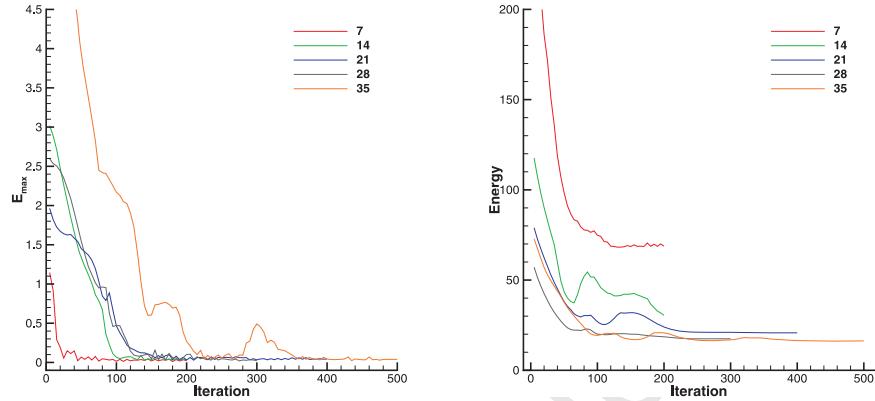


Figure 8: Convergence histories of maximum load-imbalance error E_{max} and energy function for different partitioning number.

As shown in Fig. 9, all converged partitioning subdomains are compact with particles located around the Voronoi-cell mass-center. One interesting thing is that the convergent Voronoi diagram preserves a high-degree of symmetry as the background mesh is symmetric.

The performance comparisons between CVP and VP method are shown in Fig. 10. It can be concluded that the CVP method has significantly better convergence properties than the VP method for both load-imbalance error and energy functional. Furthermore, the converged level of the energy functional from the CVP method is much lower than that from the pure VP method. Another notable outcome is that the energy functions from the CVP method converge to the same value from different initial conditions. As indicated in Fig. 11, the resulting partitioning subdomains for the VP method are much less compact than that for the CVP method. This is also

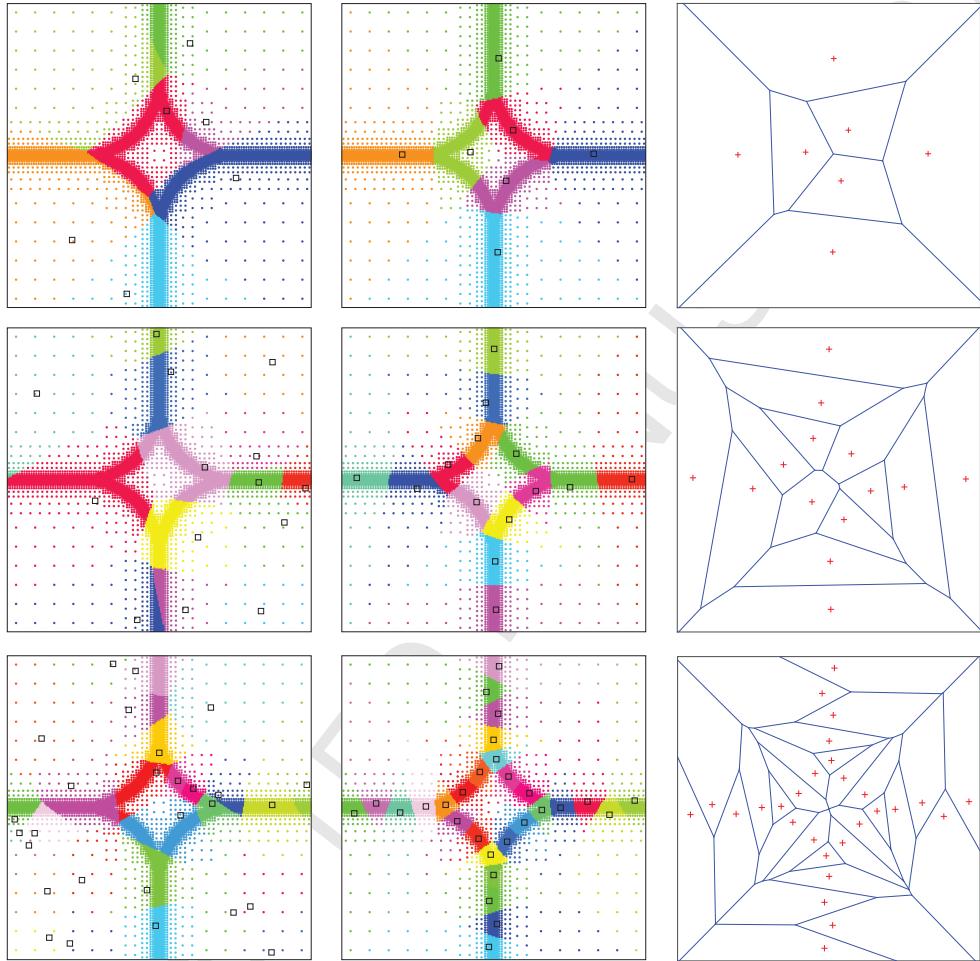


Figure 9: Sketch of partitionings and Voronoi diagrams. From the left to right: initial partitioning, convergent partitioning and convergent Voronoi diagram. From the top to bottom: partitioning number 7, 14, 28. The square and cross symbols denote the partitioning particles.

shown in Fig. 12, where compared with the pure VP method, the proposed CVP algorithm optimizes better the boundary element number, which is a proper measure of communication in parallel computing, during the entire iteration.

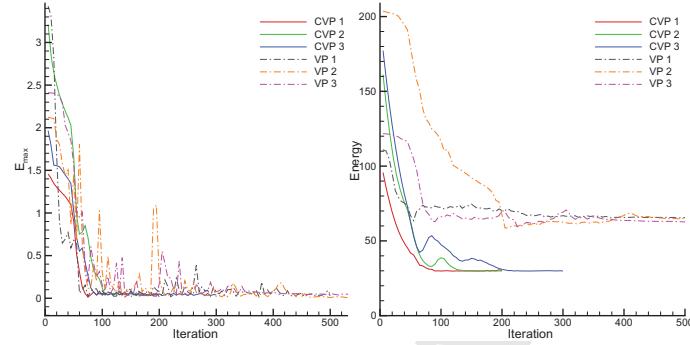


Figure 10: Convergence history comparisons between CVP and VP method. Three sets of simulations are carried out with random initial conditions. The partitioning number is 14.

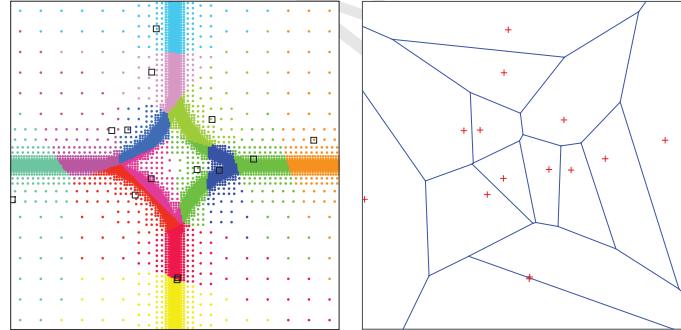


Figure 11: Sketch of partitioning result from VP method. Left panel: partitioning subdomains; right panel: Voronoi diagram. The partitioning number is 14. The square and cross symbols denote the partitioning particles.

As shown in Fig. 13, it is observed that the relaxation parameter $\alpha = 0.8$ gives the best convergence in terms of both load-imbalance error and the energy function.

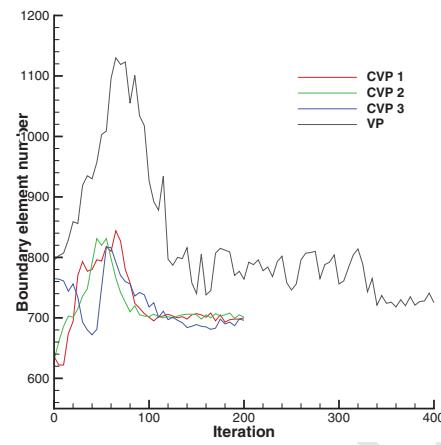


Figure 12: Comparison of the boundary element number. The mesh element is defined as boundary element under the condition that among its nearest element neighbours there is at least one element located in different subdomains.

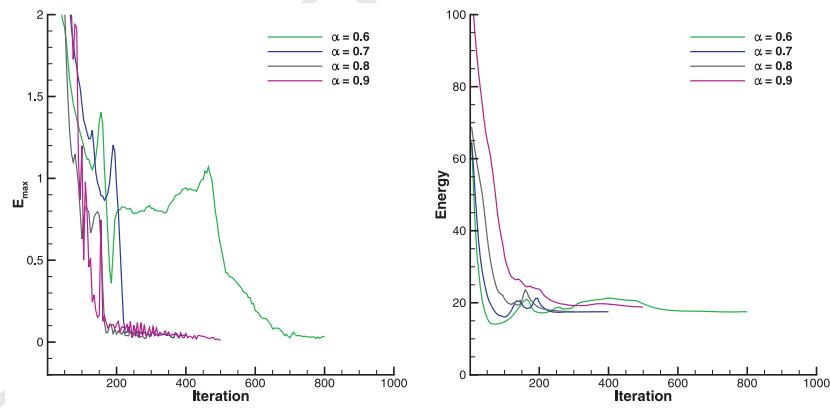


Figure 13: Convergence history comparisons with different choice of α in CVP method. The partitioning number is 28.

As shown in Fig. 14, the partitioning results from the CVT method are significantly different from those computed by the present CVP method. Since the load-imbalance error has magnitude $\mathcal{O}(1)$, the classical CVT domain decomposition method is unsuitable for parallel computing.

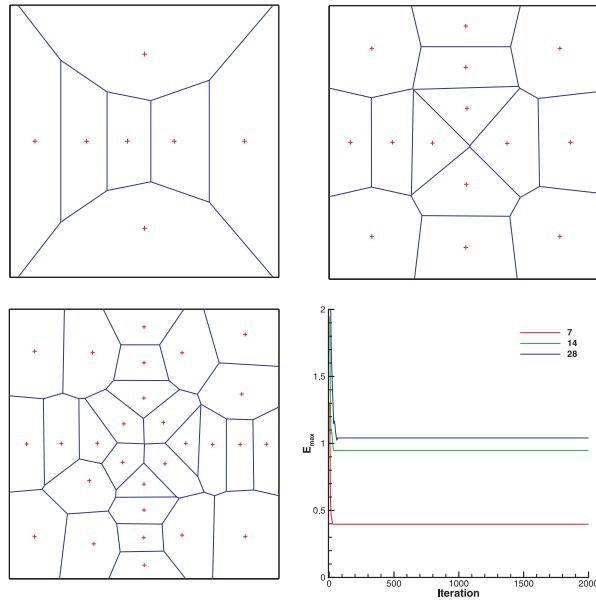


Figure 14: Sketch of partitionings and convergence history from CVT method. The partitioning numbers are 7 (top left), 14 (top right) and 28 (bottom left). Bottom right: convergence history of maximum load-imbalance error. The crosses denote the partitioning particles.

As shown in Fig. 15, by utilizing Gersho’s conjecture to redefine the density function for CVT method [19], the load-imbalance error is improved compared with that without conjecture, but still has magnitude $\mathcal{O}(1)$. The reason is that the two preconditions of the conjecture, i.e. smooth density function and infinite number of generators, are not satisfied in present applications.

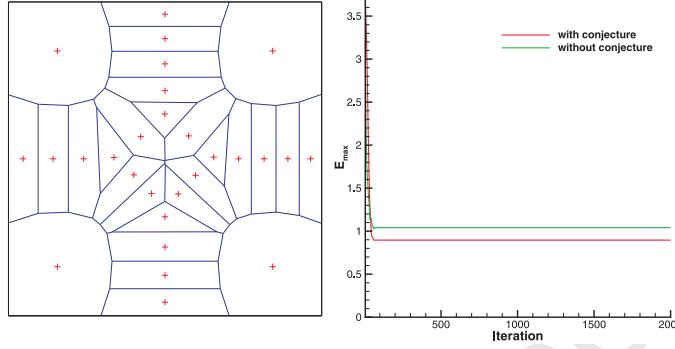


Figure 15: Sketch of partitionings (left) and convergence history (right) from CVT method based on the asymptotic conjecture. The partitioning number 28.

As shown in Fig. 16, for non-convex domain with curved boundaries, the proposed CVP method performs well and the convergence is good. Here, the domain boundaries of CVP algorithm cover the real mesh domain. It is observed that the convergence property including the slight oscillation of error function can be improved by decreasing the timestep.

4.3. Partitioning of adaptive structured and unstructured meshes

In this case, the partitioning performance for adaptive structured and unstructured meshes is demonstrated. As shown in Fig. 17, the block-structured mesh consists of 6097 blocks while the anisotropic unstructured mesh contains 14197 elements with resolution adaptation of 200.

As shown in Fig. 18, the partitioning subdomains are fairly compact. As can be seen from Fig. 19, for all three partitioning numbers, the simulation converges rapidly in less than 500 iterations within 1.8 seconds. Both the load-imbalance error and the energy function are well optimized. For partitioning number 9, the partitioning simulation reaches the equilibrium in

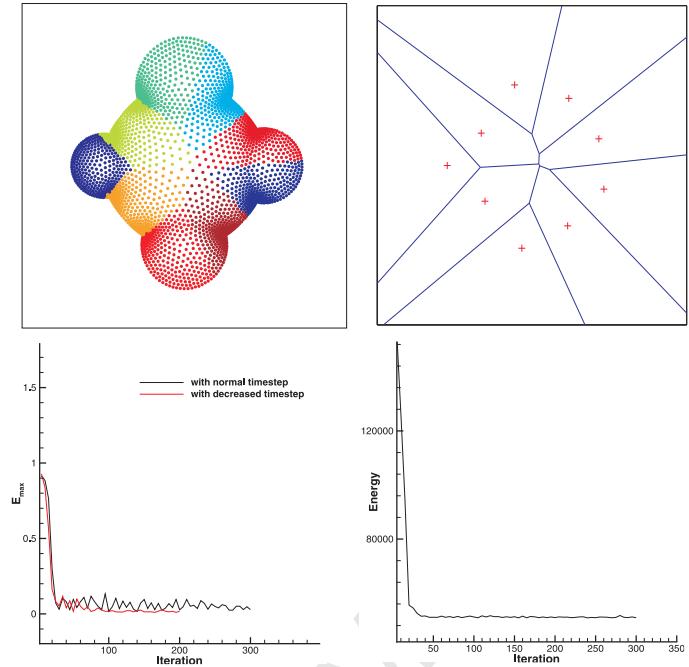


Figure 16: Sketch of partitionings and convergence history from CVP method. The partitioning number is 9. Top: particle partitioning configuration (left) and the Voronoi diagram (right). Bottom: convergence history of maximum load-imbalance error (left) and energy function (right). The crosses denote the partitioning particles.

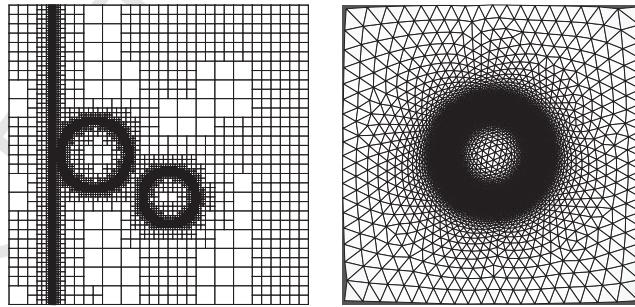


Figure 17: Sketch of the block-structured mesh (left panel) and anisotropic unstructured mesh (right panel).

about 0.3 seconds indicating high efficiency.

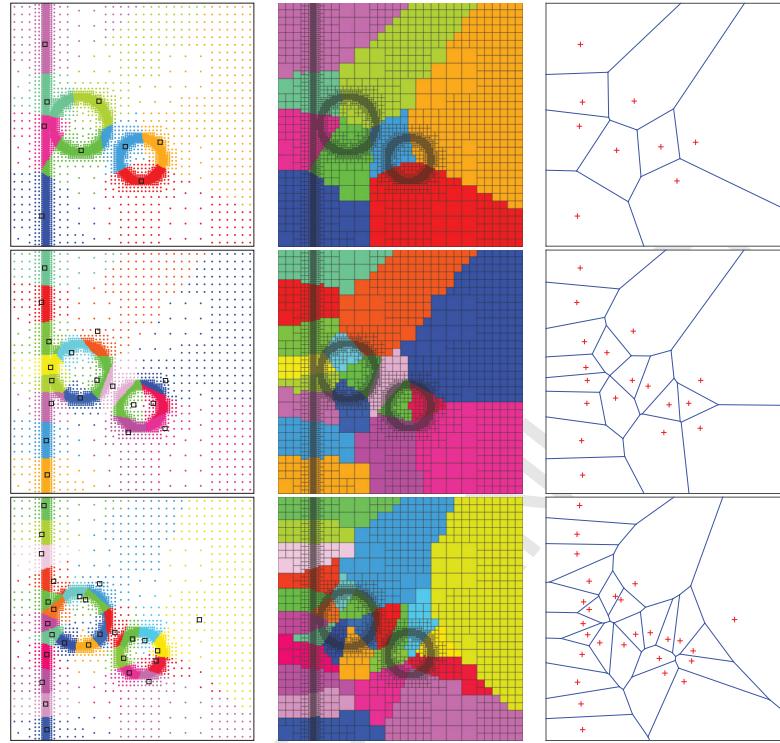


Figure 18: Sketch of partitioning for block-structured mesh for partitioning number 9, 18, 27. From the top to bottom: the partitioning number is 9, 18 and 27. From the left to right: the converged interaction particle partitioning, block-structured mesh partitioning and Voronoi diagram. The square and cross symbols denote the partitioning particles.

In Fig. 20, we show the domain decomposition with partitioning number as large as 250, e.g. each subdomain only possesses 57 mesh elements. It can be noticed that the convergence is good even for this extreme case.

4.4. 3D simulation

In this three-dimensional case, the total interaction particle number is 10000800 and the partitioning number is 64, 128 and 256 respectively. As

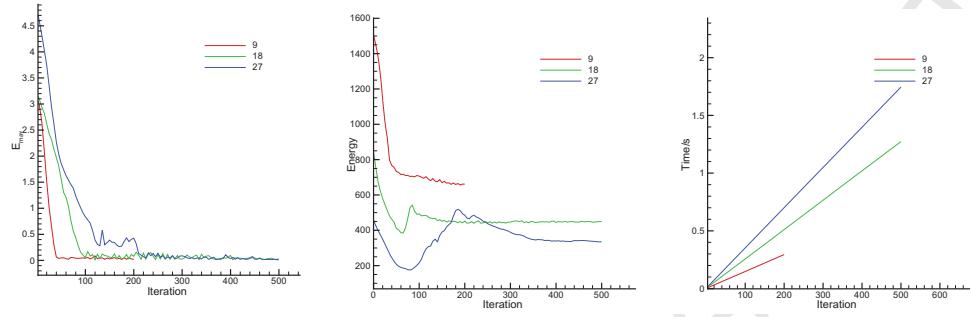


Figure 19: Convergence history of the partitioning for block-structured mesh. Maximum load-imbalance error E_{max} (left), energy function (middle) and elapsed time (right).

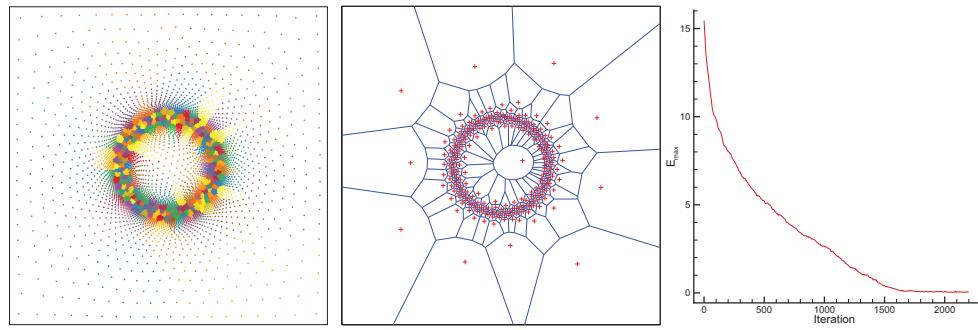


Figure 20: Sketch of the partitioning for adaptive unstructured mesh with partitioning number 250. Left: interaction particle partitioning; middle: Voronoi diagram; right: convergence history of maximum load-imbalance error E_{max} .

shown in Fig. 21, the load-imbalance error and energy function converge very well.

5. Concluding remarks

In this paper, a novel domain decomposition method, the Centroidal Voronoi Particle method, for high-performance parallel computing is proposed by combining the Centroidal Voronoi Tessellation and Voronoi Particle method. The performances of proposed domain decomposition method are summarized as follows.

- By generating the CVT diagram, Lloyd’s method optimizes the energy function monotonically, which enhances the compactness of the partitioning subdomains. The resulting partitioning subdomains are convex, **simply-connected** and have small aspect-ratio. These properties are beneficial for interface-element number reduction, i.e. communication reduction. For uniform mesh partitioning, the analytical solution can be obtained asymptotically while most of popular partitioning methods fail [30].
- With the tailored equation of state incorporating the target mass, an equal-sized partitioning can be achieved through solving the particle evolution model equation by the Voronoi Particle method. In practice, with adequate error tolerance, the particle system can be relaxed to equilibrium quickly. Furthermore, by predefining target mass proportional to the computing capability of corresponding processor, nonequal-sized partitioning can be achieved straightforwardly.

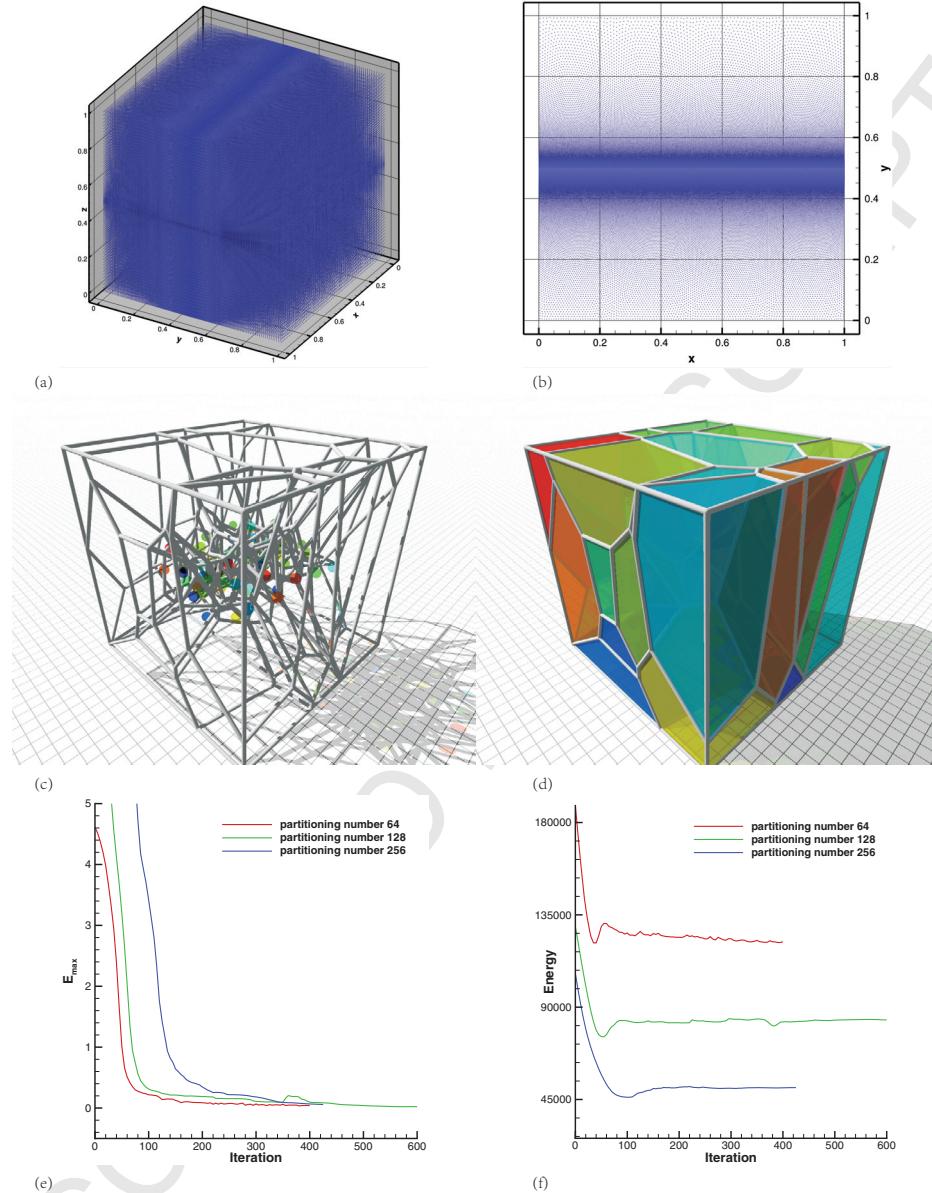


Figure 21: 3D partitioning simulation. (a) the 3D interaction particle distribution; (b) a zoom view in the x-y plane; (c) the partitioning topology of partitioning number 64; (d) the partitioning Voronoi diagram of partitioning number 64; (e) the convergence history of the load-imbalance error; (f) the convergence history of energy function.

- Since the partitioning is computed by solving a nonlinear PDE system, for which the solution continuously depends on the initial and boundary condition, the CVP partitioning method features good locality and the incremental property and thus is favourable with respect to data-migration reduction and data management.
- The proposed partitioning method is independent of mesh-element type and only mesh-element physical coordinates are necessary. The complex construction of mesh-element connectivity information is not necessary, making it generic for diverse applications. Generally speaking, the CVP method is highly efficient.
- Moreover, different from the CVT method, both the two concerned objectives, i.e. the arbitrary target mass distribution and compactness, can be optimized regardless of the smoothness of density function, the number of the generators and the boundary complexity.

Acknowledgements

The first author is partially supported by China Scholarship Council (NO. 201206290022). The authors acknowledge our colleague Zhe Ji, who extends the code from 2D to 3D.

- [1] D. E. Keyes, et al., Fifth International Symposium on Domain Decomposition Methods for Partial Differential Equations, no. 55, Siam, 1992.
- [2] S. Nepomnyaschikh, Domain decomposition methods, Radon Series Comp. Appl. Math 1 (2007) 89–160.
- [3] M. Berger, S. Bokhari, A partitioning strategy for nonuniform problems on multiprocessors, IEEE Trans. Computers 36 (1987) 570–580.

- [4] G. Karypis, V. Kumar, A fast and high quality multilevel scheme for partitioning irregular graphs, *SIAM Journal on Scientific Computing* 20 (1999) 359–392.
- [5] G. Karypis, K. V, A software package for partitioning unstructured graph, partitioning meshes, and computing fill-reducing orderings of sparse matrices. user's guide version 4.0 and version 5.0 pre2, <http://glaros.dtc.umn.edu/gkhome/metis/metis/overview> (1998) 1–44.
- [6] F. Pellegrini, Scotch and libscotch 5.1. user's guide, <http://www.labri.fr/perso/pelegrin/scotch/> (2008) 1–127.
- [7] Q. Du, X. Wang, Centroidal voronoi tessellation based algorithms for vector fields visualization and segmentation, in: Proceedings of the conference on Visualization'04, IEEE Computer Society, 2004, pp. 43–50.
- [8] A. Bowyer, Computing dirichlet tessellations, *The Computer Journal* 24 (2) (1981) 162–166.
- [9] A. Okabe, B. Boots, K. Sugihara, S. N. Chiu, *Spatial tessellations: concepts and applications of Voronoi diagrams*, Vol. 501, John Wiley & Sons, 2009.
- [10] Q. Du, V. Faber, M. Gunzburger, Centroidal voronoi tessellations: applications and algorithms, *SIAM review* 41 (4) (1999) 637–676.
- [11] Q. Du, M. Gunzburger, Grid generation and optimization based on centroidal voronoi tessellations, *Applied Mathematics and Computation* 133 (2) (2002) 591–607.
- [12] S. Valette, J.-M. Chassery, R. Prost, Generic remeshing of 3d triangular meshes with metric-dependent discrete voronoi diagrams, *Visualization and Computer Graphics, IEEE Transactions on* 14 (2) (2008) 369–381.
- [13] Q. Du, D. Wang, Anisotropic centroidal voronoi tessellations and their applications, *SIAM Journal on Scientific Computing* 26 (3) (2005) 737–761.
- [14] S. P. Lloyd, Least squares quantization in pcm, *Information Theory, IEEE Transactions on* 28 (2) (1982) 129–137.
- [15] J. MacQueen, et al., Some methods for classification and analysis of multivariate observations, in: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, Vol. 1, Oakland, CA, USA., 1967, pp. 281–297.
- [16] L. Ju, Q. Du, M. Gunzburger, Probabilistic methods for centroidal voronoi tessellations and their parallel implementations, *Parallel Computing* 28 (10) (2002) 1477–1500.
- [17] Q. Du, M. Emelianenko, Acceleration schemes for computing centroidal voronoi tessellations, *Numerical linear algebra with applications* 13 (2-3) (2006) 173–192.
- [18] Y. Liu, W. Wang, B. Lévy, F. Sun, D.-M. Yan, L. Lu, C. Yang, On centroidal voronoi tessellation—energy smoothness and fast computation, *ACM Transactions on Graphics (ToG)* 28 (4) (2009) 101.
- [19] Q. Du, D. Wang, The optimal centroidal voronoi tessellations and the gersho's conjecture in the three-dimensional space, *Computers & Mathematics with Applications* 49 (9) (2005) 1355–1373.
- [20] W. Bo, M. Shashkov, Adaptive reconnection-based arbitrary lagrangian eulerian method, *Journal of*

- Computational Physics 299 (2015) 902–939.
- [21] Q. Du, M. Gunzburger, L. Ju, Advances in studies and applications of centroidal voronoi tessellations, Numerical Mathematics: Theory, Methods and Applications 3 (2) (2010) 119–142.
 - [22] A. Genz, R. Cools, An adaptive numerical cubature algorithm for simplices, ACM Transactions on Mathematical Software (TOMS) 29 (3) (2003) 297–308.
 - [23] Q. Du, M. Emelianenko, L. Ju, Convergence of the lloyd algorithm for computing centroidal voronoi tessellations, SIAM journal on numerical analysis 44 (1) (2006) 102–119.
 - [24] R. Ostrovsky, Y. Rabani, L. J. Schulman, C. Swamy, The effectiveness of lloyd-type methods for the k-means problem, in: Foundations of Computer Science, 2006. FOCS'06. 47th Annual IEEE Symposium on, IEEE, 2006, pp. 165–176.
 - [25] S. Heß, V. Springel, Particle hydrodynamics with tessellation techniques, Monthly Notices of the Royal Astronomical Society 406 (4) (2010) 2289–2311.
 - [26] P. Espanol, Fluid particle model, Physical Review E 57 (3) (1998) 2930.
 - [27] B. Hendrickson, T. G. Kolda, Graph partitioning models for parallel computing, Parallel Computing 26 (12) (2000) 1519–1534.
 - [28] H. Meyerhenke, B. Monien, S. Schamberger, Graph partitioning and disturbed diffusion, Parallel Computing 35 (10) (2009) 544–569.
 - [29] W. Yu, X. Li, A geometry-aware data partitioning algorithm for parallel quad mesh generation on large-scale 2d regions, Procedia Engineering 124 (2015) 44–56.
 - [30] H. Meyerhenke, B. Monien, S. Schamberger, Graph partitioning and disturbed diffusion, Parallel Computing 35 (10-8C11) (2009) 544 – 569.